

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра технологій управління

ЗЮЗЮН В.І., КУБЯВКА Л.Б.

**ЗАСТОСУВАННЯ СУЧАСНИХ
ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ В УПРАВЛІННІ
ПРОЄКТАМИ**

Навчальний посібник

Електронне мережне навчальне видання

Київ – 2025

Рецензенти:

*д.т.н., професор кафедри інформаційних технологій проектування факультету інформаційних технологій і систем Черкаського державного технологічного університету (зовнішня рецензія) –
Тетяна ПРОКОПЕНКО*

*д.т.н., доцент кафедри інформаційних технологій факультету автоматизації і інформаційних технологій Київського національного університету будівництва і архітектури –
Тетяна ГОНЧАРЕНКО*

Рекомендовано кафедрою технологій управління (протокол № 9/1 від 27 лютого 2025 р.).

Рекомендовано науково-методичною комісією факультету інформаційних технологій (протокол № 8 від 9 квітня 2025 р.).

Рекомендовано Вченою радою факультету інформаційних технологій (протокол № 10 від 14 квітня 2025 р.).

**Зюзюн Вадим Ігорович
Кубявка Любов Богданівна**

Застосування сучасних інформаційних технологій в управлінні проєктами. [Електронний ресурс]: навчальний посібник для здобувачів третього (освітньо-наукового) рівня вищої освіти «Доктор філософії» за спеціальністю F3(122) «Комп'ютерні науки» / В.І. Зюзюн, Л.Б. Кубявка; Київський національний університет імені Тараса Шевченка. – Київ: КНУ ім. Тараса Шевченка, 2025. – 212 с.

Видається в авторській редакції.

© В.І. Зюзюн, Л.Б. Кубявка, 2025 рік

ЗМІСТ

ПЕРЕДМОВА.....	6
РОЗДІЛ 1. ВАЖЛИВІСТЬ ЗАСТОСУВАННЯ СУЧАСНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ В УПРАВЛІННІ ПРОЄКТАМИ.....	8
1.1. Інформаційні технології в процесах.....	8
1.2. Інформаційні програмні технології, що змінюють управління проєктами.....	10
Контрольні запитання до розділу	13
РОЗДІЛ 2. ХАРАКТЕРИСТИКА JIRA ЯК ЕФЕКТИВНОГО ІНСТРУМЕНТУ УПРАВЛІННЯ ПРОЄКТАМИ.....	14
2.1. Основні можливості Jira для проєктів.....	14
2.1.1. Управління завданнями та беклогом	18
2.1.2. Конфігурація проєктів для гнучких методологій.....	21
2.2. Використання Jira для Agile, Scrum та Kanban	30
2.2.1. Scrum: управління спринтами	30
2.2.2. Kanban: управління потоками завдань	33
2.3. Управління командами та ролями в Jira	40
2.3.1. Призначення ролей та прав доступу	41
2.3.2. Управління робочими процесами	44
2.4. Інтеграції та автоматизація процесів в Jira	48
2.4.1. Інтеграція з інструментами для розробників (GitHub, Bitbucket).....	49
2.4.2. Автоматизація через правила Jira. Створення тригерів для автоматизації	51
2.4.3. Автоматичне оновлення статусів задач в Jira	55
2.5. Звіти та аналітика в Jira	59
2.5.1. Створення звітів для Scrum-команд.....	64
2.5.2. Використання метрик та дашбордів	67
Завдання для самостійного опрацювання.....	76
Контрольні запитання до розділу	79
РОЗДІЛ 3. ОРГАНІЗАЦІЯ ПРОЦЕСІВ УПРАВЛІННЯ ПРОЄКТАМИ ІНСТРУМЕНТОМ ВІД USPACU.....	80
3.1. Основні можливості Uspace для управління проєктами.....	80
3.1.1. Інтеграція управління завданнями та комунікації	83
3.1.2. Панелі інструментів для проєктів	86
3.2. Управління командами та завданнями.....	89
3.2.1. Створення команд та розподіл задач	89

3.2.2. Спостереження за навантаженням команди	92
3.2.3. Відстеження прогресу по завданнях	95
3.3. Комунікація та співпраця в реальному часі	97
Завдання для самостійного опрацювання.....	100
Контрольні запитання до розділу	102
РОЗДІЛ 4. ЕФЕКТИВНЕ УПРАВЛІННЯ ПРОЄКТАМИ ЗА ДОПОМОГОЮ ВЕБ-ПЛАТФОРМИ MONDAY.COM	103
4.1. Основні функціональні можливості monday.com	103
4.1.1. Візуалізація робочих процесів.....	106
4.1.2. Відображення прогресу в реальному часі	110
4.2. Налаштування робочих шаблонів для проєктів	113
4.3. Управління завданнями та термінами.....	115
4.3.1. Створення завдань та контроль за термінами.....	115
4.3.2. Відстеження виконання та результатів.....	118
4.4. Спільна робота та комунікація в команді	120
4.4.1. Використання коментарів та тегів	122
4.4.2. Функція обговорень у рамках завдань.....	125
4.5. Автоматизація процесів в monday.com	126
Завдання для самостійного опрацювання.....	129
Контрольні запитання до розділу	132
РОЗДІЛ 5. УПРАВЛІННЯ ПРОЄКТАМИ ЗА ДОПОМОГОЮ ВЕБ-ПЛАТФОРМИ SMARTSHEET.....	133
5.1. Основи Smartsheet для управління проєктами	133
5.1.1. Гнучке керування таблицями для проєктів	135
5.1.2. Шаблони проєктів.....	141
5.2. Створення та відстеження завдань	143
5.2.1. Розподіл завдань та їх пріоритизація.....	147
5.2.2. Моніторинг прогресу виконання завдань	150
5.3. Управління ресурсами та часовими рамками	154
5.3.1. Визначення і контроль ресурсів	157
5.3.2. Відстеження термінів виконання	159
5.4. Спільна робота та звітування. Автоматизація звітів про виконання проєкту	161
Завдання для самостійного опрацювання.....	164
Контрольні запитання до розділу	167
РОЗДІЛ 6. ВЕБ-ПЛАТФОРМА ONLINE GANTT ДЛЯ УПРАВЛІННЯ ПРОЄКТАМИ: ПЛАНУВАННЯ ТА ВІДСТЕЖЕННЯ	168
6.1. Основні функції Online Gantt для управління проєктами	168

6.2. Інструменти для управління завданнями.....	170
6.2.1. Додавання завдань	170
6.2.2. Спільне управління завданнями	171
6.2.2. Управління робочим навантаженням і ресурсами	171
6.3. Спільна робота в межах проєкту	172
6.3.1. Спільне редагування проєктного плану	172
6.3.2. Спільне управління завданнями	173
6.4. Звіти та аналітика	174
Завдання для самостійного опрацювання.....	176
Контрольні запитання до розділу	179
ТЕСТОВІ ЗАВДАННЯ ДЛЯ САМОКОНТРОЛЮ	180
ГЛОСАРІЙ.....	187
РЕКОМЕНДОВАНА ЛІТЕРАТУРА.....	204
ПРЕДМЕТНИЙ ПОКАЖЧИК.....	211

ПЕРЕДМОВА

У сучасному світі інформаційних технологій управління проектами стало однією з найважливіших складових успішної діяльності організацій. Швидкий розвиток технологій, постійні зміни у вимогах ринку та зростаюча конкуренція вимагають від управлінців не лише глибоких знань у своїй сфері, а й адаптивності та готовності до впровадження інновацій. У цьому контексті використання сучасних інформаційних технологій і програмних застосунків стало невід'ємною частиною управлінських процесів у сфері проектів.

Цей навчальний посібник створений, щоб надати читачам загальні уявлення та знання про сучасні інформаційні технології, які значно полегшують управління процесами виконання проектів. У ньому розглядатимуться різноманітні програмні рішення, які допомагають планувати, координувати, контролювати та оцінювати ефективність проектів. Читачі, зазвичай мають, детальну інформацію про такі інструменти автоматизації управлінських процесів, Jira, Trello, Asana, та Microsoft Project. Ці загальновідомі платформи забезпечують структурований підхід до виконання завдань, контроль термінів, управління ресурсами та моніторинг прогресу.

Проте існують й інші, не менш потужні та ефективні рішення, які заслуговують уваги.

До таких платформ належать, зокрема, Uspace, Smartsheet, monday.com, та Online Gantt. Кожен із цих інструментів пропонує унікальні можливості для організації командної роботи, відстеження прогресу та управління завданнями в проектах. Вони підтримують різні підходи до управління, включаючи Agile і Waterfall, та дозволяють інтегруватися з іншими сервісами, що робить їх надзвичайно гнучкими для різних потреб.

Однак, попри численні переваги, інтеграція нових інформаційних програмних технологій у процеси управління стикається з низкою викликів. Опір змінам з боку співробітників, питання навчання та витрат на впровадження нових технологій потребують серйозного розгляду. Цей посібник допоможе

читачам усвідомити важливість підготовки персоналу та формування організаційної культури, яка сприяє адаптації до нових умов.

Зокрема, буде надано практичні приклади впровадження цих інструментів в організаціях різного масштабу. Наприклад, Jira широко використовується в ІТ-компаніях для трекінгу завдань, управління проектами та автоматизації робочих процесів. Завдяки функціоналу, що підтримує методології Agile, Jira, Uspasy, Smartsheet, monday.com та Online Gantt надають можливість командам швидко адаптуватися до змін у вимогах проєкту, що в свою чергу підвищує продуктивність та якість виконання.

Даний посібник стане корисним інструментом для студентів, фахівців і всіх, хто прагне вдосконалити свої навички в управлінні проектами, використовуючи сучасні інформаційні технології. Знання, отримані з цього навчального матеріалу, сприятимуть формуванню нових ідей і підходів у сфері управління проектами, а також допоможуть стати більш конкурентоспроможними на ринку праці.

РОЗДІЛ 1. ВАЖЛИВІСТЬ ЗАСТОСУВАННЯ СУЧАСНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ В УПРАВЛІННІ ПРОЄКТАМИ

1.1. Інформаційні технології в процесах

У сучасному світі інформаційні технології стали невід'ємною частиною управлінських процесів. Вони не лише спростили рутинні завдання, а й значно підвищили ефективність роботи організацій. Завдяки швидкому розвитку ІТ, управлінці мають доступ до новітніх інструментів, які дозволяють оптимізувати внутрішні процеси, покращити комунікацію та забезпечити конкурентоспроможність компанії.

Однією з основних переваг впровадження інформаційних технологій є автоматизація управлінських процесів. Сучасні програмні рішення дозволяють автоматизувати рутинні завдання, такі як ведення обліку, планування ресурсів та моніторинг проєктів. Наприклад, системи управління проєктами, такі як *Jira* або *Trello*, забезпечують ефективне управління завданнями, контроль термінів та ресурсів. Це зменшує ймовірність помилок, сприяє більш точному виконанню проєктів і знижує навантаження на співробітників, які можуть зосередитися на більш стратегічних завданнях.

Інформаційні технології також відіграють ключову роль у покращенні комунікації всередині організацій. Впровадження електронної пошти, систем обміну повідомленнями та відеоконференцій дозволяє зменшити час на прийняття рішень і забезпечити швидкий обмін інформацією. У глобалізованому світі, де команди можуть працювати в різних частинах світу, важливість миттєвого обміну інформацією важко переоцінити. Це дозволяє не лише зекономити час, але й покращити співпрацю між співробітниками.

Однак інформаційні технології не лише покращують ефективність управлінських процесів, а й відкривають нові можливості для аналізу даних. Використання аналітичних інструментів, таких як Business Intelligence, дозволяє компаніям збирати, аналізувати та інтерпретувати великі обсяги даних, що надає можливість приймати обґрунтовані рішення на основі фактичних даних. Завдяки

цим технологіям управлінці можуть отримати глибокі сучасні знання та можливості про ринок, конкурентів та споживачів, що сприяє розробці ефективних стратегій розвитку.

Проте впровадження інформаційних технологій у процеси управління також пов'язане з низкою викликів. Однією з головних проблем є складнощі інтеграції нових систем в існуючу інфраструктуру. Багато організацій стикаються з опором змінам з боку співробітників, які можуть побоюватися, що нові технології загрожують їхній роботі. Це вимагає не лише технічних, а й культурних змін в організації. Управлінці повинні активно залучати співробітників до процесу впровадження нових технологій, пояснюючи їм переваги та можливості, які вони можуть отримати.

Іншою важливою проблемою є вартість впровадження інформаційних технологій. Хоча багато з цих технологій можуть суттєво підвищити продуктивність, початкові витрати на їх впровадження можуть бути значними. Підприємства повинні ретельно оцінювати ризики та можливі вигоди від інвестицій в інформаційні технології, щоб прийняти обґрунтоване рішення про їх впровадження.

Загалом, інформаційні технології стали важливим інструментом для управлінців, які прагнуть підвищити ефективність своїх організацій. Вони дозволяють автоматизувати процеси, покращувати комунікацію та проводити детальний аналіз даних. Незважаючи на численні виклики, пов'язані з їх впровадженням, вигоди від використання інформаційних технологій переважають ризики, і компанії, які інвестують у новітні технології, мають значні конкурентні переваги.

Інформаційні технології продовжують відігравати критично важливу роль у процесах управління. Для успішного впровадження ІТ у бізнес-процеси необхідно не лише забезпечити належну технологічну базу, а й створити культуру, що підтримує зміни. Це дозволить організаціям ефективно реагувати на виклики сучасного світу та досягати своїх цілей.

1.2. Інформаційні програмні технології, що змінюють управління проектами

У сучасному світі управління проектами стало складним і відповідальним завданням. Поява нових інформаційних програмних технологій відкриває нові горизонти для ефективного управління проектами, надаючи інструменти для планування, моніторингу та виконання завдань. Завдяки цим технологіям управлінці можуть підвищувати продуктивність, знижувати ризики і досягати вищих результатів.

Однією з ключових груп інформаційних програмних технологій, що змінили управління проектами, є платформи для автоматизації проектів. Такі інструменти, як *Jira*, *Asana* та *Trello*, дозволяють командам управляти завданнями, відстежувати прогрес і забезпечувати прозорість у виконанні проектів. Наприклад, у компанії *Atlassian*, яка розробила *Jira*, використовують цю платформу для управління своїми проектами та командами. Вона дозволяє візуалізувати етапи виконання завдань за допомогою *Kanban-дошок*, що сприяє ефективній координації дій і покращує командну комунікацію. Ці системи надають можливість інтегрувати різноманітні функції, такі як планування ресурсів, контроль термінів і візуалізація робочих процесів. Таким чином, автоматизація процесів не лише скорочує час на виконання рутинних завдань, а й підвищує ефективність комунікації між членами команди.

Впровадження хмарних технологій також значно змінило підходи до управління проектами. Хмарні рішення, такі як *Microsoft Azure*, *Amazon Web Services (AWS)* та *Google Cloud*, забезпечують організаціям доступ до необмежених ресурсів без необхідності в значних капітальних інвестиціях у власну інфраструктуру. Наприклад, компанія *Spotify* використовує *AWS* для зберігання та обробки величезних обсягів даних, що дозволяє їй аналізувати поведінку користувачів і оптимізувати свої сервіси. Це дозволяє командам працювати з будь-якої точки світу, забезпечуючи гнучкість у виконанні завдань. Хмарні технології також дозволяють компаніям зберігати дані, аналізувати їх і проводити моніторинг проектів у реальному часі, що є критично важливим для

успішного завершення проєктів.

Ще одним важливим аспектом є впровадження методології Agile та DevOps, які змінюють традиційні підходи до управління проєктами. Ці методології акцентують увагу на гнучкості, швидкості реакції на зміни та безперервній інтеграції. *Agile* методи, такі як *Scrum* і *Kanban*, дозволяють командам адаптуватися до змін у вимогах замовників і швидко реагувати на нові виклики. Наприклад, компанія *Spotify* реалізувала методологію *Scrum*, розділивши команди на мікросервіси, що дозволяє кожній команді працювати над окремими функціями платформи, забезпечуючи при цьому високу швидкість розробки. *DevOps*, у свою чергу, об'єднує команди розробників і експлуатаційників, забезпечуючи безперервну доставку та інтеграцію продукту, що призводить до зниження часу виходу на ринок та підвищення якості продукту.

Інформаційні програмні технології, засновані на штучному інтелекті та аналітиці даних, також мають значний вплив на управління проєктами. Використання штучного інтелекту дозволяє автоматизувати аналіз даних, що збираються під час виконання проєкту, і генерувати прогнози щодо потенційних ризиків або проблем. Наприклад, *IBM Watson* використовує алгоритми машинного навчання для аналізу даних з проєктів, надаючи рекомендації щодо управління ризиками. Системи на базі штучного інтелекту можуть виявляти патерни, які можуть призвести до затримок у виконанні проєкту, надаючи управлінцям можливість вжити заходів до виникнення проблем. Також, аналітичні інструменти можуть забезпечувати детальну інформацію про ефективність команди, дозволяючи краще планувати ресурси та оптимізувати робочі процеси.

Крім того, інформаційні технології значно покращили комунікацію всередині команд. Інструменти, такі як *Slack*, *Microsoft Teams* та *Zoom*, дозволяють членам команди спілкуватися в режимі реального часу, обмінюватися файлами та проводити відеоконференції. Компанія *Salesforce*, наприклад, використовує *Microsoft Teams* для спілкування між різними відділами, що дозволяє швидко вирішувати проблеми та обговорювати стратегії.

Це суттєво підвищує продуктивність, оскільки команди можуть швидко обговорювати питання, приймати рішення та вирішувати проблеми. Поліпшення комунікації сприяє зменшенню ризику непорозумінь і затримок у виконанні завдань.

Попри численні переваги, інтеграція нових інформаційних програмних технологій у процеси управління проектами також стикається з низкою викликів. По-перше, опір змінам з боку співробітників може стати серйозною перешкодою. Багато з них можуть не бути готовими до адаптації нових технологій через страх втрати роботи або недостатню кваліфікацію. Тому важливо забезпечити навчання та підтримувати культуру змін у команді, щоб сприяти успішному впровадженню нових інструментів.

По-друге, важливим є питання витрат на впровадження нових технологій. Хоча інвестиції в сучасні програмні рішення можуть суттєво підвищити продуктивність, початкові витрати можуть бути значними. Організації повинні ретельно аналізувати ризики та вигоди, щоб прийняти обґрунтоване рішення про впровадження нових технологій.

У підсумку, інформаційні програмні технології стали важливими інструментами для управлінців проектів, які прагнуть підвищити ефективність та досягти успіху в умовах сучасної конкуренції. Вони забезпечують автоматизацію процесів, покращують комунікацію та надають можливості для детального аналізу даних. Однак для успішного впровадження нових технологій необхідно враховувати людський фактор, проводити навчання та формувати культуру змін в організації. Це дозволить компаніям адаптуватися до нових викликів і досягати своїх цілей у стрімко змінюваному світі.

Контрольні запитання до розділу

1. Що таке інформаційні технології, і як вони використовуються в процесах управління?
2. Яким чином інформаційні технології підвищують ефективність прийняття рішень в управлінні?
3. Як автоматизація управлінських процесів за допомогою ІТ-технологій впливає на продуктивність команди?
4. Які приклади інформаційних технологій використовуються для покращення управлінських процесів в організаціях?
5. Які основні проблеми можуть виникнути при інтеграції інформаційних технологій в управлінські процеси?
6. Які програмні технології найбільше впливають на сучасне управління проектами?
7. Як хмарні технології змінюють спосіб управління проектами в ІТ?
8. Які переваги дає використання Agile-інструментів для управління проектами?
9. Як штучний інтелект і машинне навчання можуть впливати на управління проектами?
10. Які програмні платформи найчастіше використовуються для покращення співпраці в проектних командах?

РОЗДІЛ 2. ХАРАКТЕРИСТИКА JIRA ЯК ЕФЕКТИВНОГО ІНСТРУМЕНТУ УПРАВЛІННЯ ПРОЄКТАМИ

2.1. Основні можливості Jira для проєктів

Jira є одним із найпопулярніших і найпотужніших інструментів управління проєктами в ІТ-індустрії, який широко використовується командами для планування, відстеження та управління розробкою програмного забезпечення. Вона дозволяє ефективно контролювати робочі процеси, сприяти співпраці між членами команди та забезпечувати високу прозорість усього проєкту. Основні можливості *Jira*, які роблять її такою популярною серед проєктних команд, включають. Нижче розглянемо декілька ключових можливостей більш детально (рис. 2.1).

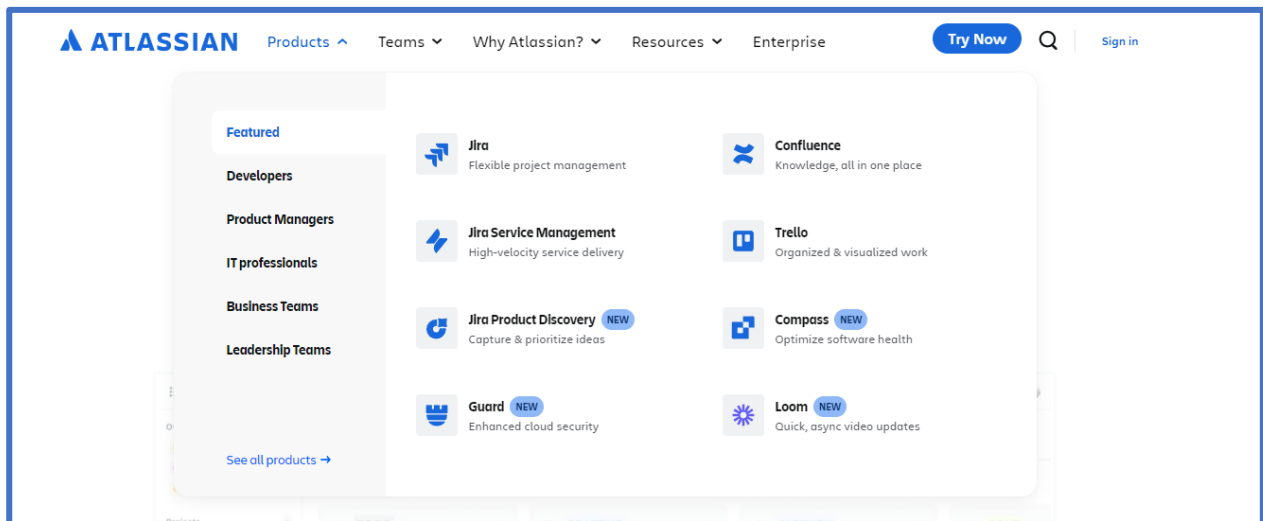


Рис. 2.1 – Початок роботи в Jira

Управління завданнями в проєктах. *Jira* дозволяє створювати, відслідковувати та управляти різними типами завдань: багами, новими функціями, технічними покращеннями, тестуванням та іншими. Кожне завдання має чітко визначені атрибути, такі як пріоритет, статус, відповідальна особа, терміни завершення та опис, що допомагає забезпечити прозорість і чіткість у процесі виконання. Завдяки гнучкості *Jira*, команди можуть легко масштабувати проєкти від невеликих стартапів до великих корпоративних систем (рис. 2.2).

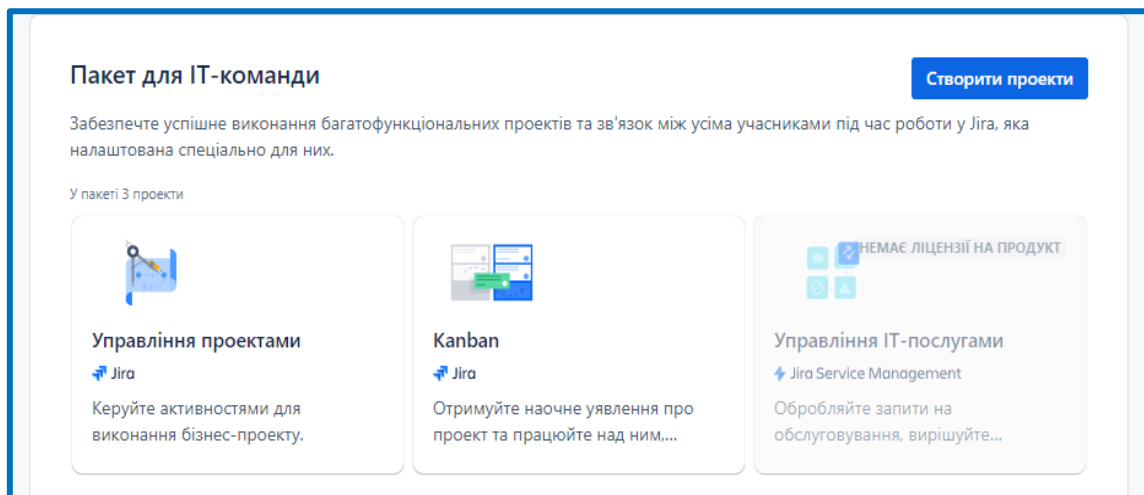


Рис. 2.2 – Теоретичні відомості щодо роботи з Jira та створення Kanban дошок

Адаптивні робочі процеси. **Jira** дозволяє налаштовувати робочі процеси під специфічні потреби кожної команди або проєкту, що є великою перевагою для проєктів різної складності та галузей. Команди можуть створювати різні статуси для завдань, додавати автоматизовані дії (тригери), а також створювати складні логіки переміщення завдань між етапами. Це дозволяє організувати роботу відповідно до внутрішніх процесів команди та забезпечити зручний контроль за виконанням завдань (рис. 2.3).

Підтримка гнучких методологій управління проектами. **Jira** ідеально підходить для команд, які використовують гнучкі (*Agile*) методології управління проектами, такі як *Scrum* і *Kanban*. Для *Scrum*-команд **Jira** пропонує зручні інструменти для управління беклогом, планування спринтів, моніторингу прогресу під час спринтів і огляду виконаної роботи на підсумкових зустрічах (ретроспективах). Для команд, що працюють за методом *Kanban*, є можливість створювати візуальні дошки, що дозволяють відслідковувати рух завдань по стовпцях у реальному часі. Це значно спрощує керування змінами в проєктах і дозволяє швидше адаптуватися до нових вимог.

Аналітика та звіти. **Jira** пропонує широкий набір інструментів для аналітики та генерації звітів, що дають можливість вивчати прогрес виконання проєкту, продуктивність команди та прогнозувати можливі затримки або вузькі

місця в процесах. Серед вбудованих звітів є burndown та velocity діаграми для Scrum-команд, діаграми контролю прогресу для Kanban-команд, а також різноманітні звіти для аналізу часу виконання завдань, виявлення багів та інших ключових показників продуктивності (KPI). Крім того, **Jira** дозволяє створювати користувацькі дашборди для відображення ключової інформації в реальному часі.

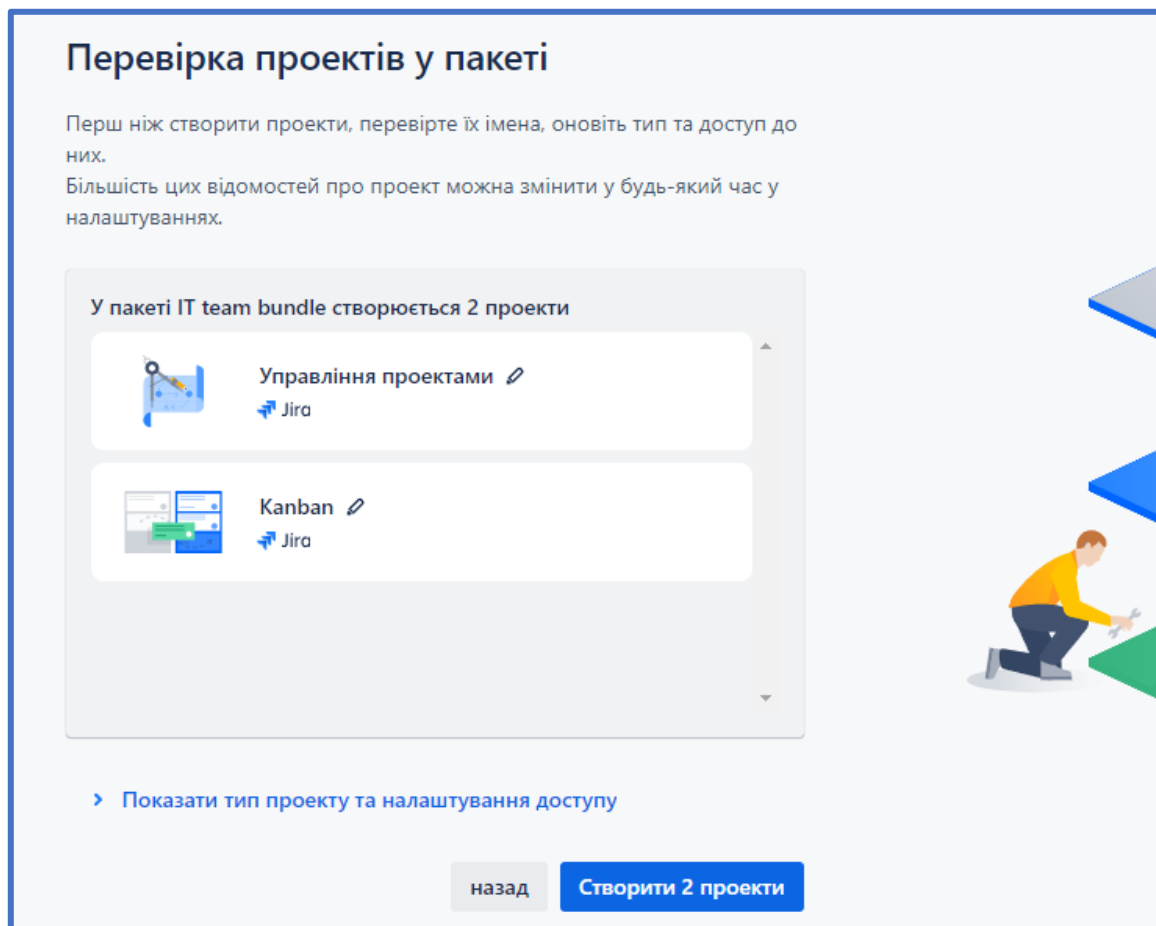


Рис. 2.3 – Теоретичні відомості щодо роботи з Jira та створення Kanban- дошок

Інтеграція з іншими інструментами. **Jira** легко інтегрується з іншими популярними інструментами для розробки та управління проектами, що робить її універсальним рішенням для *DevOps* та *Agile-команд*. Серед таких інструментів можна виділити *Bitbucket*, *GitHub*, *GitLab* для управління репозиторіями коду, *Bamboo* та *Jenkins* для безперервної інтеграції (CI/CD), а також *Slack*, *Microsoft Teams* та інші інструменти для комунікації всередині

команди. Така інтеграція дозволяє оптимізувати робочі процеси та забезпечити безперервність роботи над проєктом без переходів між різними платформами.

Кастомізація інтерфейсу та автоматизація. **Jira** дозволяє налаштовувати інтерфейс під специфічні потреби кожної команди або проєкту, що забезпечує гнучкість у роботі з великими обсягами даних. Користувачі можуть додавати власні поля, встановлювати фільтри для швидкого доступу до важливих завдань, автоматизувати процеси через додаткові плагіни або правила автоматизації. Це дозволяє зменшити ручну роботу та збільшити ефективність виконання повторюваних завдань. *Наприклад*, можна автоматизувати оновлення статусів завдань, відправку повідомлень або створення підзавдань на основі певних тригерів.

Безпека та контроль доступу. **Jira** надає широкий функціонал для керування доступом користувачів до даних проєкту. Вона дозволяє встановлювати різні рівні доступу для різних членів команди та забезпечувати належний контроль над конфіденційною інформацією. Наприклад, окремі групи користувачів можуть мати доступ лише до певних проєктів або завдань, в той час як інші мають повний доступ для адміністрування. Це забезпечує безпеку та контроль за виконанням завдань у великих командах, де є необхідність розподілу обов'язків та обмеження доступу до певних ресурсів.

Мобільний доступ та підтримка віддаленої роботи. **Jira** пропонує мобільні додатки для *iOS* та *Android*, що дозволяють командам залишатися на зв'язку та управляти проєктами навіть у віддаленому режимі. Це особливо важливо в умовах сучасних реалій, коли більшість проєктних команд працюють дистанційно або в гібридному форматі. Мобільні додатки забезпечують доступ до основних функцій **Jira** на ходу, дозволяють відслідковувати статус завдань, додавати нові коментарі та брати участь у командній співпраці з будь-якого місця.

Широка екосистема плагінів і розширень. **Jira** має величезну кількість доступних плагінів та розширень через *Atlassian Marketplace*, що дозволяє додатково покращувати функціональність платформи. Команди можуть

встановлювати плагіни для управління ризиками, бюджетами, документами, інтеграцій з *CRM* та *ERP-системами*. Це дозволяє кастомізувати *Jira* під специфічні вимоги проєкту або компанії, забезпечуючи додаткові можливості для гнучкого управління.

Ці можливості роблять *Jira* незамінним інструментом для команд, що працюють над великими, складними та довготривалими проєктами, допомагаючи зберігати організованість, прозорість і ефективність у всіх процесах.

2.1.1. Управління завданнями та беклогом

Jira – це потужний інструмент для управління проєктами, що надає командам можливість організовувати, відстежувати та контролювати завдання на всіх етапах їхнього виконання. Одна з основних функцій *Jira* – це управління задачами та беклогом, що дозволяє командам структурувати свою роботу, планувати її на майбутнє та ефективно розподіляти ресурси. Важливим аспектом цього процесу є розуміння англomовного інтерфейсу, який широко використовується у багатьох ІТ-компаніях по всьому світу.

Розглянемо основні елементи інтерфейсу *Jira*:

1. «*Create Issue*» (*Створити завдання*). Функція, яка дозволяє користувачам створювати нові завдання в системі. При натисканні цієї кнопки відкривається форма, де можна ввести основні деталі, такі як заголовок завдання, опис, відповідальна особа, пріоритет та дедлайн. Це ключовий етап для започаткування роботи над новими функціями, виправленнями багів або іншими задачами, які потребують уваги команди.
2. «*Backlog*» (*Беклог*). Це місце, де зберігаються всі невиконані завдання та ідеї, що очікують на свій час. Беклог є своєрідним резервуаром для завдань, які ще не були включені в активну роботу, але будуть розглянуті командою під час планування спринтів або етапів проєкту. Він допомагає структурувати майбутню роботу, впорядкувати її за пріоритетами та мати під рукою всі необхідні завдання для подальшої реалізації.

3. «Board» (Дошка). Дошка є центральним інструментом для відслідковування прогресу завдань. Вона візуально представляє процес виконання завдань, дозволяючи команді переміщувати завдання між статусами, такими як «To Do» (До виконання), «In Progress» (У процесі) та «Done» (Виконано). Це дає змогу наочно бачити поточний стан справ та реагувати на зміни в режимі реального часу.
4. «Sprint» (Спринт). Спринт – це короткий, зазвичай двотижневий період часу, протягом якого команда зосереджена на виконанні конкретних завдань з беклогу. Спринти допомагають розбити великий обсяг роботи на більш керовані частини, дозволяючи команді фокусуватися на найважливіших завданнях і поступово досягати поставлених цілей.
5. «Issues» (Завдання). Цей розділ містить загальний список усіх завдань у проєкті. Його можна використовувати для фільтрації завдань за різними параметрами, такими як пріоритет, статус або відповідальний, що полегшує управління великими проєктами та знижує ризик втратити контроль над важливими завданнями.
6. «Assign» (Призначити). Функція для вибору відповідальної особи за виконання завдання. Цей процес дозволяє чітко розподілити обов'язки серед членів команди та забезпечити виконання завдань у встановлені терміни.
7. «Epic» (Епік). Це велике завдання або історія, що об'єднує кілька менших завдань (наприклад, історії користувачів або підзавдання). Епіки допомагають структурувати складні проєкти, групуючи пов'язані між собою завдання у більш зрозумілі блоки, які легше планувати та виконувати.
8. «Sub-task» (Підзавдання). Це частина більшого завдання, яке розбивається на менші кроки для спрощення його виконання. Користувачі можуть створювати підзавдання для детальнішого розподілу роботи.
9. «Priority» (Пріоритет). Кожне завдання може бути відмічене пріоритетом, що визначає його важливість. Завдання можуть бути «High» (Високий),

- «*Medium*» (Середній) або «*Low*» (Низький) пріоритет, що допомагає команді зосереджуватися на найважливіших задачах у першу чергу.
- 10.«*Status*» (Статус). Статус показує поточний стан завдання – від його початку до завершення. Команди можуть налаштовувати власні статуси, які відповідають їхнім процесам роботи, наприклад, «*To Do*», «*In Progress*» або «*Done*».
 - 11.«*Comment*» (Коментар). Дозволяє учасникам команди залишати примітки, додавати уточнення чи обговорювати деталі завдання безпосередньо в картці завдання.
 - 12.«*Due Date*» (Термін виконання). Дата, до якої завдання має бути завершене. Це допомагає планувати роботу й уникати затримок у виконанні проєктних завдань.
 - 13.«*Log Work*» (Зареєструвати роботу). Функція для реєстрації часу, витраченого на виконання завдання, що дає можливість відслідковувати продуктивність та оцінювати тривалість виконання робіт.
 - 14.«*Resolution*» (Рішення). Відображає, як було вирішене завдання, наприклад, виправлено чи відхилено.
 - 15.«*Labels*» (Мітки). Ключові слова або теги, які дозволяють здійснити категоризацію завдань та полегшують пошук подібних завдань.

Розглянемо декілька прикладів управління завданнями. Уявімо, що ваша команда працює над розробкою нової функції для мобільного додатку. Керівник проєкту створює нове завдання, натискаючи кнопку «*Create Issue*» (Створити завдання) і заповнює необхідну інформацію: заголовок завдання, опис функції та термін виконання «*Due Date*». Потім він призначає відповідального, використовуючи функцію «*Assign*» (Призначити), і додає це завдання до «*Backlog*» для подальшого розгляду.

Під час планування наступного спринту команда бере це завдання з беклогу, додає його до «*Sprint*», і починає роботу. Завдання переміщується на дошці «*Board*» зі статусу «*To Do*» у статус «*In Progress*», коли команда починає працювати над його реалізацією (рис. 2.4).

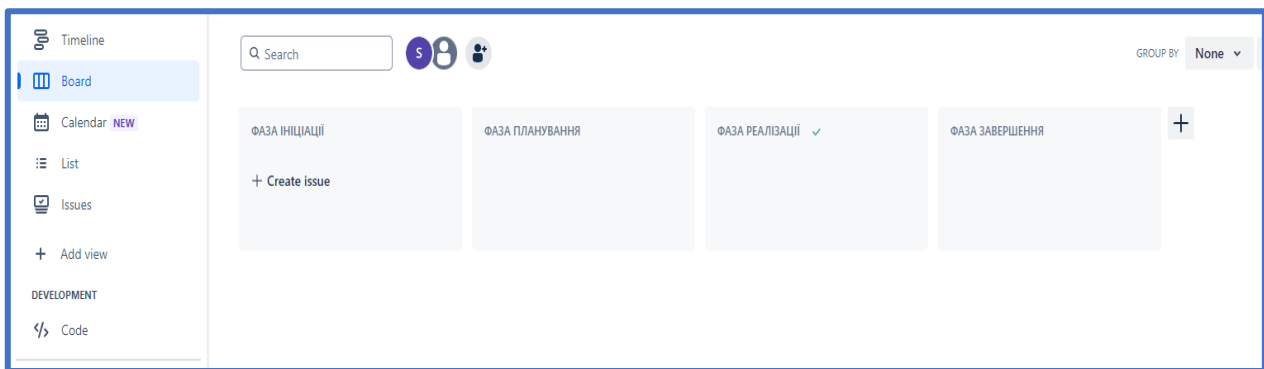


Рис. 2.4 – Фази дошки KAN

В процесі роботи відповідальна особа реєструє виконані години через функцію «*Log Work*» (*Зареєструвати роботу*) та додає коментарі для команди, щоб обговорити складнощі або нові вимоги. Коли завдання виконано, команда змінює статус на «*Done*» і переходить до наступних етапів.

Таким чином, **Jira** забезпечує структуроване та прозоре управління завданнями, спрощуючи роботу команди й допомагаючи дотримуватися встановлених термінів та пріоритетів.

2.1.2. Конфігурація проєктів для гнучких методологій

Для створення нового Scrum-проєкту в **Jira** необхідно:

- увійти в **Jira**;
- налаштувати мову інтерфейсу на українську чи англійську (рис. 2.5);
- натиснути «*Project*» у верхньому меню та обрати «*Create project*» (рис. 2.6);
- обрати шаблон *Scrum* (зазвичай знаходиться під категорією «*Software development*» (*Розробка програмного забезпечення*)) (рис. 2.7);
- ввести назву проєкту та інші необхідні дані;
- натиснути «*Create*» (*Створити*) для створення проєкту.

Параметри облікового запису

Керуйте налаштуваннями облікового запису.

Мова та регіон

Зміни параметрів мови й часових поясів відобразяться в Jira, Confluence, Trello, Bitbucket і каталозі. Параметри мови й часових поясів для інших продуктів слід змінювати в налаштуваннях цих продуктів.

Мова

Українська

Часовий пояс

Ваш часовий пояс

Видалення облікового запису

Якщо видалити обліковий запис, ви втратите доступ до сервісів облікового запису Atlassian, а ми назавжди видалимо ваші особисті дані. Скасувати видалення можна протягом 14 днів.

Видалити обліковий запис

[Дізнатися більше](#)

Рис. 2.5 – Налаштування мови інтерфейсу

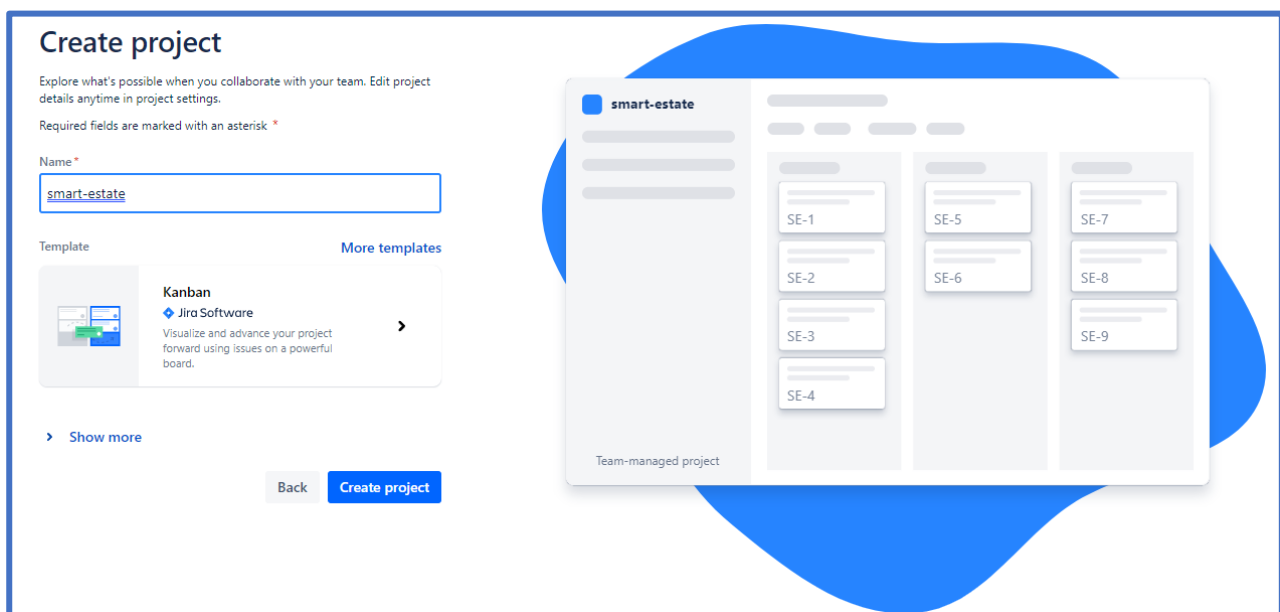


Рис. 2.6 – Створення проекту в Jira

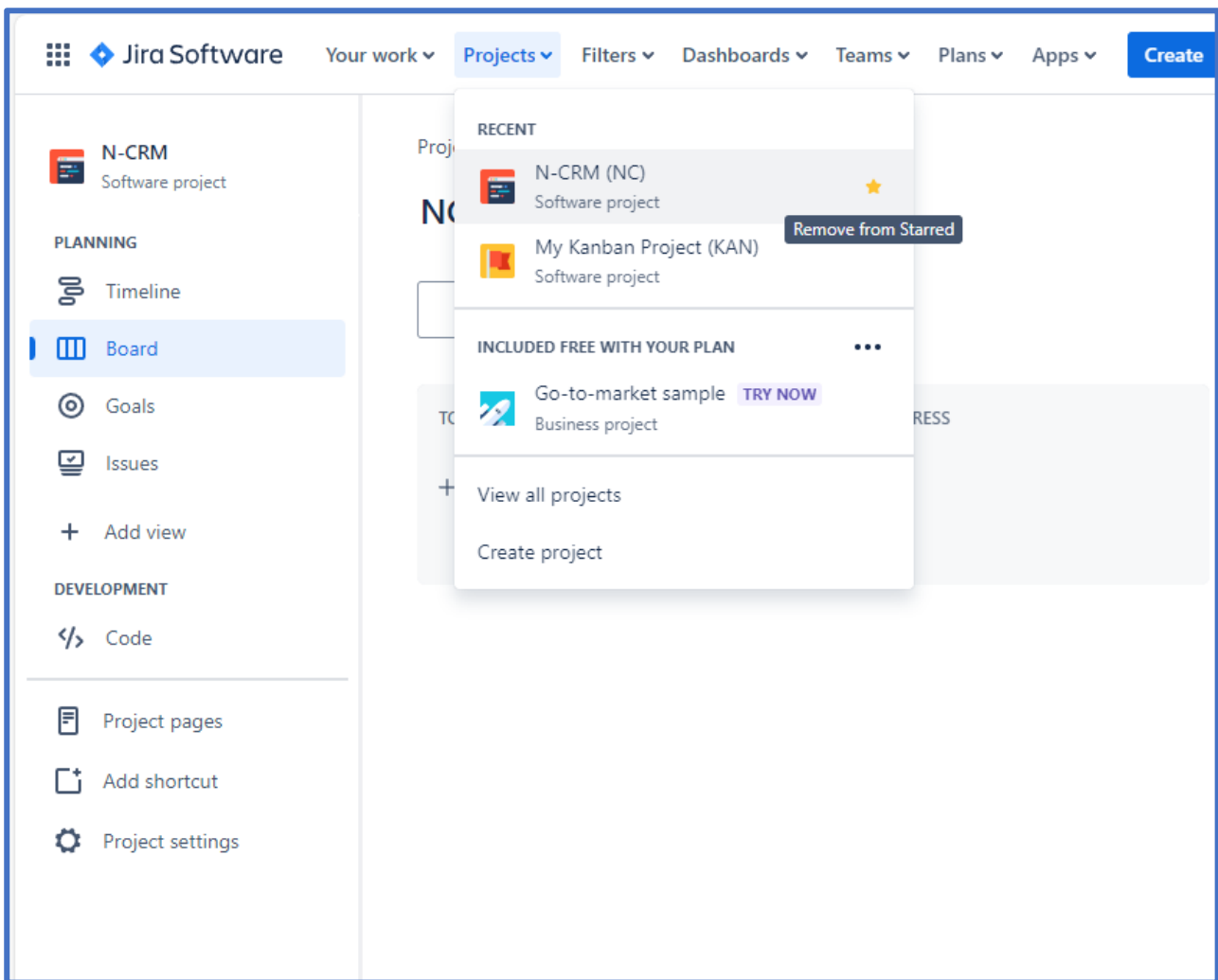


Рис. 2.7 – Налаштування Scrum-проєкту

Налаштування Scrum-проєкту

Налаштування Scrum-проєкту в **Jira** є критично важливим етапом, що допомагає командам ефективно організувати свою роботу та досягати поставлених цілей. Щоб розпочати, відкрийте вже створений *Scrum-проєкт*. У верхньому меню знайдіть вкладку «Boards» або перейдіть до «Backlog» (рис. 2.7). Саме тут ви зможете ознайомитися з двома основними компонентами бортів Scrum. Перший з них – це «Backlog», де плануються завдання для майбутніх спринтів, а другим – «Active sprints» (*Активні спринти*), де команда відслідковує активні задачі протягом поточного спринту.

Після цього перейдіть до налаштувань проєкту. Знайдіть розділ «Project settings» та виберіть «People». Тут ви можете додати учасників команди, визначивши їх ролі – розробники, тестувальники, адміністратори тощо. Це

дозволить чітко розподілити обов'язки та забезпечити ефективну комунікацію в команді.

Наступним кроком є налаштування спринтів. Для цього відкрийте «*Backlog*» і натисніть кнопку «*Create sprint*» (*Створити спринт*), щоб ініціювати новий спринт. У цьому процесі ви можете перетягувати задачі з беклогу до активного спринту, визначаючи пріоритети. Визначте тривалість спринту – зазвичай це 1-4 тижні, і, коли все буде готово, натисніть «*Start sprint*», щоб приступити до виконання задач.

Додайте нові задачі, переходячи до беклогу проєкту. Натисніть кнопку «*Create issue*» для додавання нової задачі. При створенні задачі виберіть її тип, наприклад, «*Epic*», «*Story*», «*Task*» або «*Bug*». Введіть назву, детальний опис та інші необхідні дані, призначте відповідальних осіб та вкажіть пріоритет задачі. Це допоможе вам краще структурувати роботу та розподілити обов'язки серед членів команди.

Важливим аспектом є налаштування статусів задач, або «*Workflow*». Перейдіть у «*Project settings* → *Workflows*» (*Налаштування проєкту* → *Робочі процеси*). Тут ви зможете редагувати або створювати нові етапи виконання задачі, такі як «*To Do*», «*In Progress*» і «*Done*». Щоб налаштувати переходи між цими етапами, просто натисніть на статус задачі та задайте відповідні правила (рис. 2.8).

Для оцінки задач у спринті використовуйте «*Story Points*» (*Бали за користувачьку історію*) або інші одиниці виміру. Відкрийте кожну задачу, введіть кількість балів у полі «*Story Points*». Це дозволить вам оцінити складність задач та відповідно розподілити роботу між членами команди, що сприятиме більш ефективному виконанню завдань.

Не забувайте про важливість звітності. Використовуйте звіти Scrum для відстеження прогресу команди. Звіти доступні у вкладці «*Reports*», де ви можете переглянути, наприклад, «*Burn Down Chart*» (*Діаграма згорання*), яка показує, скільки роботи залишилося до завершення спринту, або «*Velocity report*» (*Звіт про швидкість*), що відображає швидкість команди за кілька спринтів. Це

допоможе вам оцінити продуктивність команди та виявити можливі області для покращення.

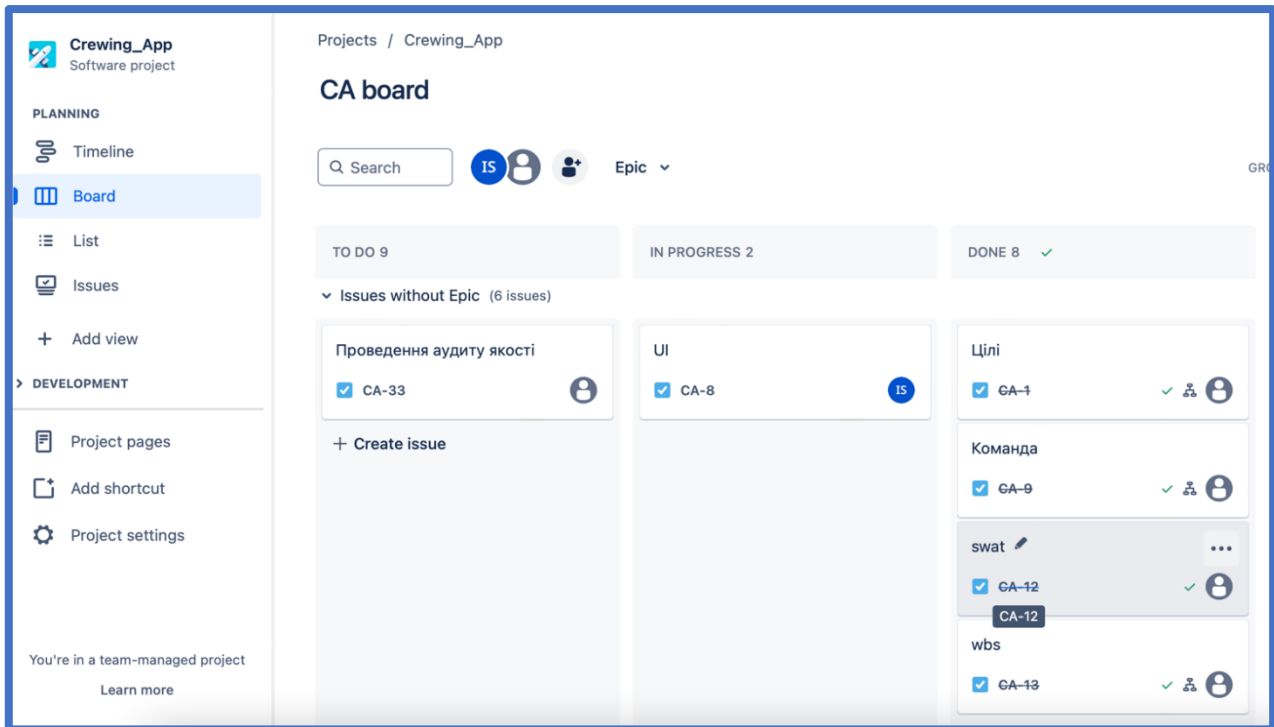


Рис. 2.8 – Налаштування статусів задач

По завершенню спринту перейдіть до вкладки «*Active sprints*» натисніть кнопку «*Complete sprint*». Це важливий етап, на якому команда може обговорити виконані задачі на ретроспективі. Під час цього обговорення ви повинні перенести незавершені задачі в наступний спринт або в беклог, щоб забезпечити їх подальше опрацювання.

Розглянемо **приклад** налаштувань *Scrum-проєкту*:

1. *Створення спринту*. Команда планує спринт тривалістю два тижні. Вони перетягують 12 задач з беклогу до активного спринту, включаючи 8 задач типу «*Task*» і 4 задачі типу «*Bug*».
2. *Додавання учасників команди*. Менеджер проєкту додає трьох розробників, одного спеціаліста з тестування та аналітика, визначаючи їх ролі для чіткого розподілу обов'язків.
3. *Оцінка задач*. Під час оцінювання команда встановлює 8 Story Points для задачі «Розробка інтерфейсу користувача» та 3 Story Points для задачі

«Тестування функціоналу».

4. *Аналіз звітів*. Після завершення спринту команда вивчає «*Velocity Report*» (*Звіт про швидкість*) і виявляє, що їхня швидкість за попередні два спринти зросла з 20 до 30 *Story Points*. Це спонукає їх обговорити, як вони можуть ще більше підвищити продуктивність у майбутніх спринтах.

Таким чином, налаштування *Scrum-проекту* в *Jira* – це комплексний процес, що включає створення структури, розподіл обов’язків, оцінювання задач і аналіз результатів, що в свою чергу дозволяє команді ефективно працювати над проектами та досягати високих результатів.

Налаштування Kanban-проекту

Налаштування *Kanban-проекту* в *Jira* – це процес, що допомагає командам ефективно організувати свою роботу, забезпечити прозорість та підвищити продуктивність. Почніть з входу в *Jira*. У верхньому меню натисніть «*Projects*» та оберіть «*Create project*». Далі оберіть шаблон «*Kanban*», який зазвичай знаходиться в категорії «*Software development*». Заповніть необхідні поля, такі як назва проекту, ключ проекту та опис, після чого натисніть кнопку «*Create*» для завершення процесу створення проекту (рис. 2.9).

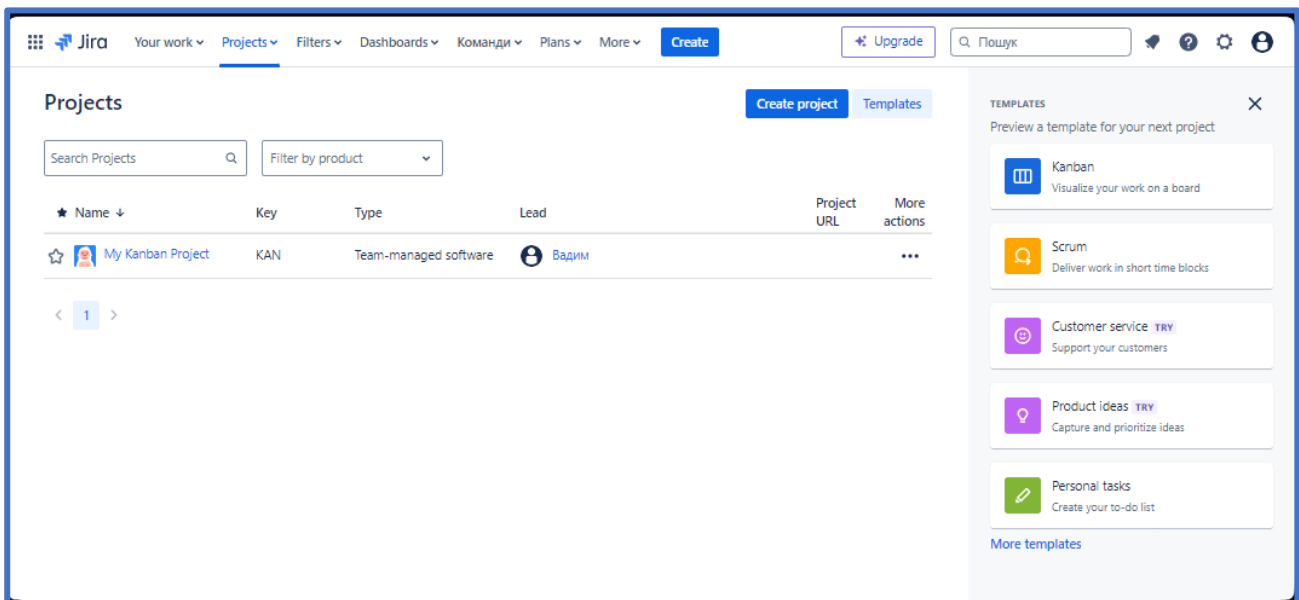


Рис. 2.9 – Створення *Kanban-проекту* в *Jira*

Після створення проєкту відкрийте його, і ви побачите Kanban-дошку, яка відображає етапи робочого процесу. Базові колонки на дошці включають «Backlog», «To Do», «In Progress» і «Done». Ці колонки допомагають візуалізувати стан задач і їх просування (рис. 2.10). Ви можете додавати нові колонки або редагувати існуючі для відображення специфічних етапів вашого робочого процесу, що дозволить адаптувати дошку до потреб команди.

Далі, для забезпечення чіткої структури команди, перейдіть до «Project settings» і виберіть «People». Додайте членів команди, вказавши їх ролі – це можуть бути розробники, тестувальники або адміністратори. Чітке визначення ролей допоможе створити злагоджену командну взаємодію.

Переходьте до додавання задач. Для цього ви можете перейти на Kanban-дошку або в беклог проєкту. Натисніть кнопку «Create issue» для створення нової задачі. Виберіть тип задачі – це може бути «Task», «Bug» або «Story». Введіть назву, опис задачі, визначте пріоритет та призначте відповідальних осіб. Також не забудьте додати будь-яку іншу необхідну інформацію, таку як терміни виконання або вкладення.

Наступним етапом є налаштування «Workflow» (Робочий процес). Перейдіть у «Project settings → Workflows». Ви можете редагувати наявні стани задач або додавати нові. Важливо налаштувати правила переходів між станами задач – це можуть бути умови, тригери та перевірки, що дозволяють автоматизувати робочі процеси та уникати затримок.

Для контролю робочого навантаження використовуйте *WIP limits* (обмеження кількості задач у колонках). На Kanban-дошці у верхньому правому куті натисніть «Board settings» (Налаштування дошки), виберіть розділ «Columns» (Колонки). Для кожної колонки ви можете встановити обмеження на кількість задач, які можуть перебувати в ній одночасно. Це допоможе контролювати обсяг роботи та уникати перевантаження команди.

Ви також можете налаштувати фільтри для швидкого доступу до задач. У налаштуваннях борту редагуйте «Quick Filters» (Швидкі фільтри), що дозволяє швидко фільтрувати задачі за різними критеріями, такими як пріоритет,

виконавець або статус. Це значно полегшить перегляд актуальних задач для окремих членів команди.

Автоматизація – ще один важливий аспект налаштування Kanban-проекту. Перейдіть до «*Project settings* → *Automation*» (*Налаштування проекту* → *Автоматизація*) і додайте правила автоматизації, які можуть включати автоматичне оновлення статусів задач, відправлення сповіщень або зміни пріоритетів у залежності від певних подій (рис. 2.10). Це допоможе зменшити рутинні завдання та зосередитися на важливіших аспектах проекту.

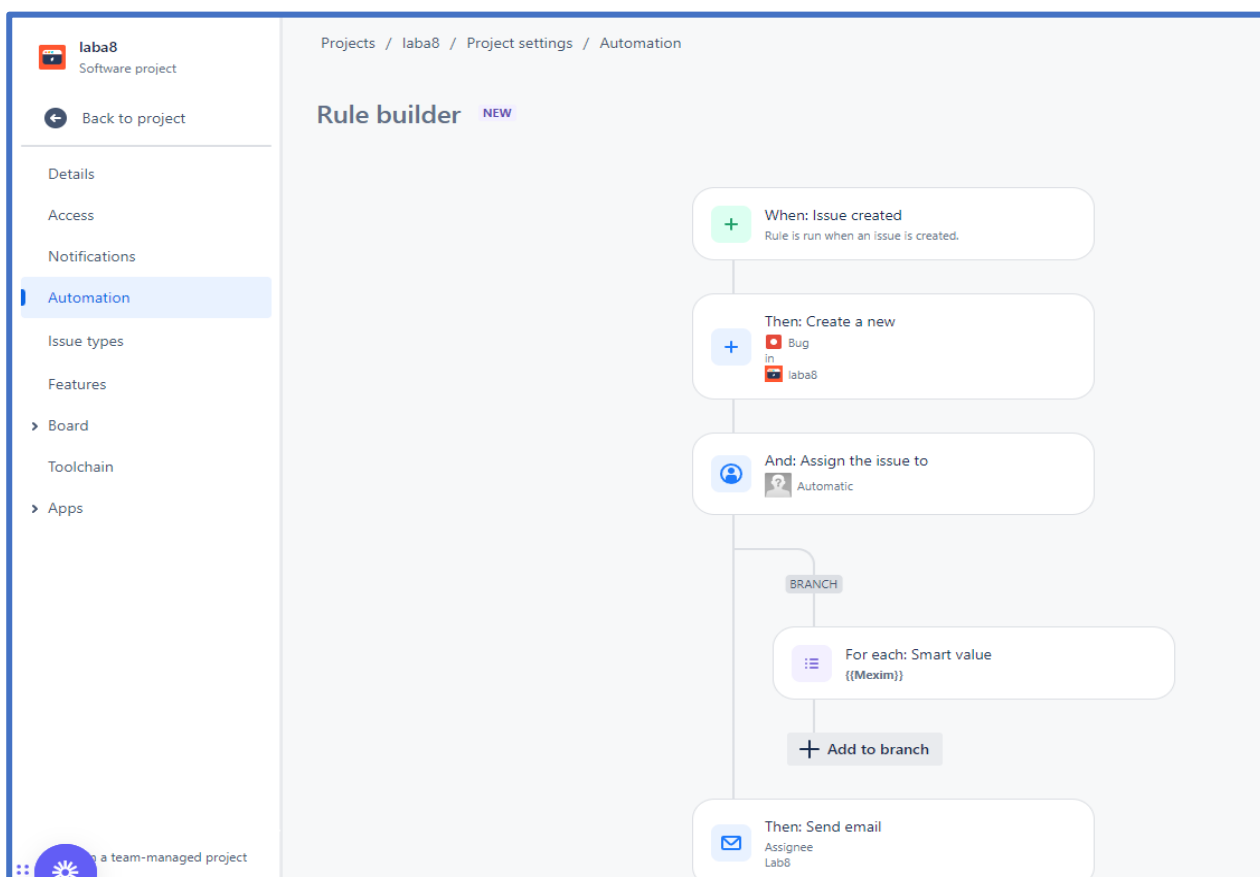


Рис. 2.10 – Створення автоматизації проекту

Також використовуйте звіти та метрики для аналізу прогресу. Kanban надає можливість переглядати різні звіти, такі як «*Cumulative Flow Diagram*» (*Кумулятивна діаграма потоку*), яка показує кількість задач на кожному етапі процесу з плином часу, або «*Control Chart*» (*Контрольна діаграма*), що дозволяє аналізувати швидкість виконання задач. Для перегляду звітів перейдіть до вкладки «*Reports*» на панелі інструментів проекту.

Нарешті, важливо регулярно перевіряти стан задач на Kanban-дошці, щоб уникнути затримок у виконанні. Переглядайте налаштування *Workflow*, *WIP limits* та звіти, щоб оптимізувати робочий процес і підвищити ефективність команди. Це дозволить вам вчасно реагувати на проблеми та покращувати продуктивність.

Розглянемо декілька прикладів налаштувань Kanban-проєкту:

1. Додавання нової колонки. Команда вирішує додати колонку «*In Review*» (*На перевірці*) між «*In Progress*» та «*Done*», щоб покращити процес перевірки виконаних задач.
2. Встановлення *WIP limits*. Команда встановлює обмеження в 3 задачі для колонки «*In Progress*», щоб уникнути перевантаження розробників і покращити фокус на виконанні поточних завдань (рис.2.11).
3. Автоматизація статусів. Налаштовано автоматичне оновлення статусу задачі з «*In Progress*» на «*Done*» після того, як задача була підтверджена тестувальником, що зменшує рутинні роботи.
4. Використання швидких фільтрів. Команда створює швидкий фільтр для задач з високим пріоритетом, що дозволяє членам команди швидко зосередитися на найважливіших завданнях у поточний момент.

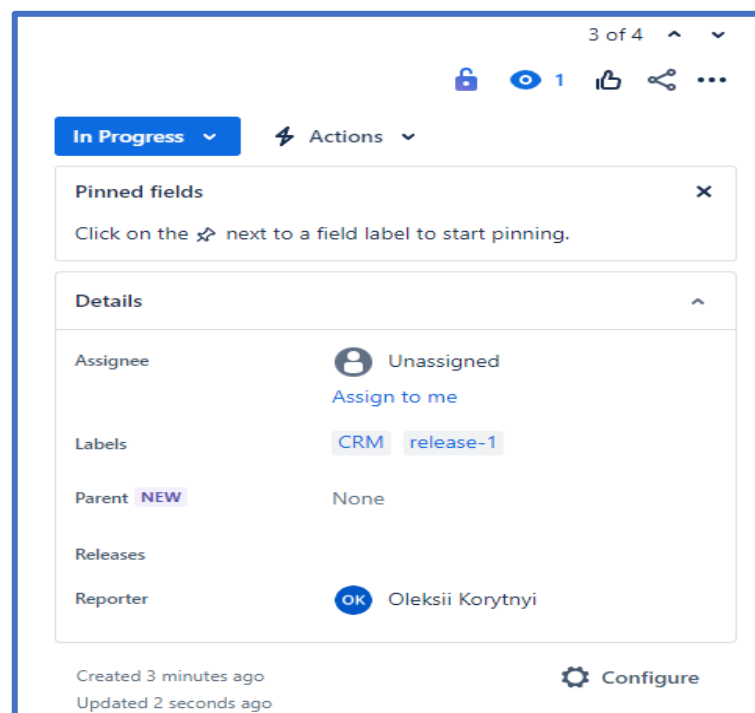


Рис. 2.11 – Створення лейблів

Таким чином, налаштування *Kanban-проекту* в *Jira* є ключовим етапом, що дозволяє командам організувати свою роботу, покращити комунікацію та забезпечити ефективність виконання задач.

2.2. Використання Jira для Agile, Scrum та Kanban

2.2.1. Scrum: управління спринтами

Для того щоб створити спринт у *Jira*, вам спочатку необхідно увійти до системи, ввівши свої облікові дані. Після успішного входу, у верхньому меню виберіть потрібний проєкт зі списку «*Projects*» або на головній сторінці *Jira*. Щоб відкрити беклог проєкту, натисніть на «*Backlog*» на лівій бічній панелі. Це дозволить вам переглянути список усіх задач, які ще не прив'язані до активних спринтів.

Для створення нового спринту натисніть кнопку «*Create sprint*», яка зазвичай розташована у верхній частині вікна беклогу. У списку з'явиться новий спринт, за замовчуванням підписаний як «*Sprint 1*», і ви зможете редагувати його назву відповідно до потреб вашої команди. Щоб додати задачі до спринту, перетягніть їх з «*Backlog*» у новостворений спринт (рис. 2.12). Альтернативно, за допомогою правої кнопки миші ви можете вибрати опцію «*Move to sprint*» (*Перемістити у спринт*) для кожної задачі окремо. Переконайтеся, що задачі мають оцінки, такі як «*Story Points*», щоб правильно планувати обсяг роботи.

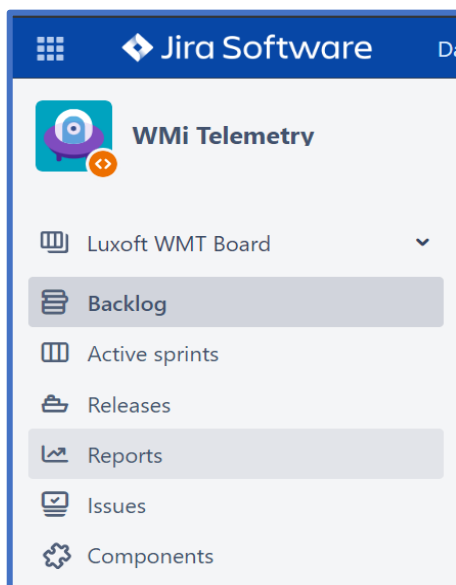


Рис. 2.12 – Навігаційна панель Jira

Щоб визначити тривалість спринту, натисніть на спринт у списку та виберіть «*Edit sprint*» (*Редагувати спринт*). У полі «*Sprint duration*» (*Тривалість спринту*) оберіть потрібний термін спринту, який зазвичай варіюється від 1 до 4 тижнів. Після додавання всіх необхідних задач натисніть кнопку «*Start sprint*» (*Запустити спринт*) у верхній частині екрана. Тут ви зможете вказати дату початку та завершення спринту, і натиснути «*Start*», щоб офіційно розпочати процес.

Протягом активного спринту ви маєте можливість відстежувати прогрес команди, користуючись вкладкою «*Active sprints*», що знаходиться на лівій бічній панелі. У цій секції відображаються всі активні задачі спринту. Борд (дошка задач) дозволяє вам відслідковувати задачі, які поділені на колонки відповідно до їх статусів, таких як «*To Do*», «*In Progress*» і «*Done*». Коли член команди починає роботу над задачею, він може перетягнути її з колонки «*To Do*» у «*In Progress*», а після завершення – у «*Done*» (рис. 2.13).

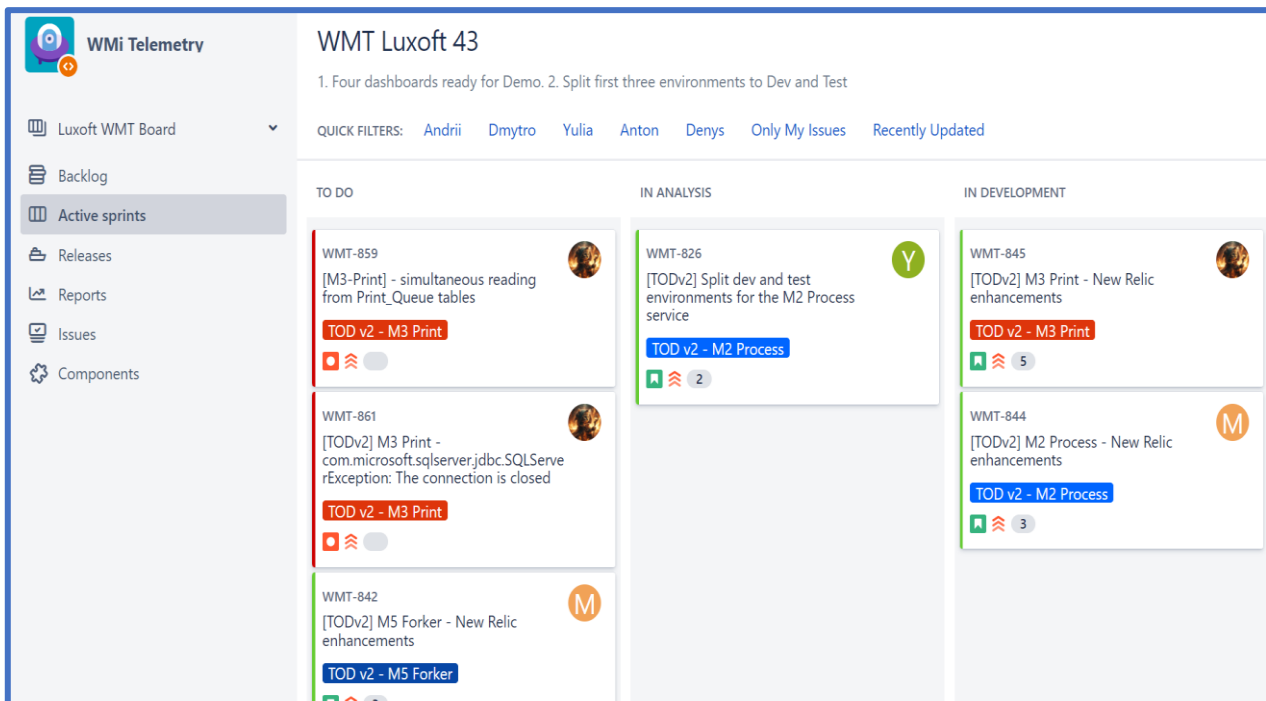


Рис. 2.13 – Активний спринт

В активному спринті ви також можете переглядати звіти, які допомагають аналізувати прогрес команди. Для цього натисніть на вкладку «*Reports*» (Звіти) на лівій панелі. Серед доступних звітів, що можуть бути корисними для вашої

команди, є:

- «*Burndown Chart*» (Діаграма згорання) – показує, скільки роботи залишилося до завершення спринту, надаючи змогу візуалізувати прогрес.
- «*Sprint Report*» (Звіт по спринту) – містить інформацію про всі задачі в спринті, їх статус та прогрес, що дозволяє оцінити виконання задач.
- «*Velocity Report*» (Звіт про швидкість) – відображає швидкість виконання задач командою за кілька попередніх спринтів, що допомагає у плануванні майбутніх робіт.

Кожну задачу можна оновити або за допомогою дошки, або натиснувши кнопку «*Edit*» (Редагувати). Це дозволяє змінити статус задачі, додати коментарі або оновити оцінки, забезпечуючи актуальність інформації.

Daily Standups (Щоденні збори) також грають важливу роль у підтриманні комунікації між членами команди та моніторингу прогресу. Команда може використовувати Kanban-дошку або вкладку «*Active sprints*», щоб бачити, які задачі наразі в роботі, і оновлювати їхній статус перед зборами.

Коли спринт підходить до завершення або всі задачі виконані, натисніть «*Complete sprint*» у верхньому правому куті на вкладці «*Active sprints*». *Jira* запропонує перенести незавершені задачі в наступний спринт або в беклог, що забезпечить чіткий план для подальшої роботи.

Після завершення спринту важливо провести ретроспективу, обговорюючи, що було зроблено добре, а що можна покращити. Ви можете використовувати «*Sprint Report*» для аналізу продуктивності команди та виявлення проблем, які могли завадити виконанню задач.

Для планування наступного спринту можна повторити описаний вище процес, слідуючи тим самим крокам. Регулярне використання звітів, таких як «*Burndown Chart*», допоможе виявляти потенційні затримки, тоді як «*Velocity Report*» дозволить коригувати планування майбутніх спринтів на основі продуктивності команди.

Розглянемо декілька **прикладів** використання спринтів у *Jira*:

1. Спринт на поліпшення продукту. Команда планує спринт тривалістю два

тижні, присвячений виправленню помилок у програмному забезпеченні. Вони включають до спринту завдання, пов'язані з критичними помилками, і використовують «*Burndown Chart*» для контролю прогресу.

2. Спринт для розробки нової функції. Команда вирішує створити нову функцію в продукті за три тижні. Вони формують спринт, додавши задачі, що описують етапи розробки, тестування та впровадження функції, а потім відстежують прогрес за допомогою «*Sprint Report*».
3. Планування спринту на основі швидкості. Після аналізу даних з «*Velocity Report*» команда вирішує, що може виконати більше задач у наступному спринті. Вони додають до спринту додаткові задачі, які раніше не були включені, щоб максимізувати продуктивність.

Завершення спринту з перенесенням задач. Коли спринт підходить до завершення, команда має кілька незавершених задач. Вони вирішують перенести їх у наступний спринт, щоб зберегти зосередженість на виконанні нових завдань, тим самим покращуючи планування майбутніх робіт.

Таким чином, створення та управління спринтами в *Jira* – це ефективний спосіб організації командної роботи, що сприяє досягненню поставлених цілей і підвищенню продуктивності команди.

Поради:

1. Варто регулярно оновлювати статуси задач для точного відображення прогресу.
2. Контролювати обсяг задач у спринті, щоб уникнути перевантаження команди.
3. Використовувати звіти для аналізу продуктивності та пошуку можливих покращень.

2.2.2. Kanban: управління потоками завдань

Визначення робочих процесів Kanban-проекту. Щоб розпочати роботу, увійдіть до *Jira*, використовуючи ваші облікові дані. Після входу зверніться до верхнього меню, де необхідно натиснути на «*Projects*» (рис. 2.14). У спадному

меню виберіть ваш *Kanban-проект* або, за потреби, створіть новий за допомогою кнопки «Create project».

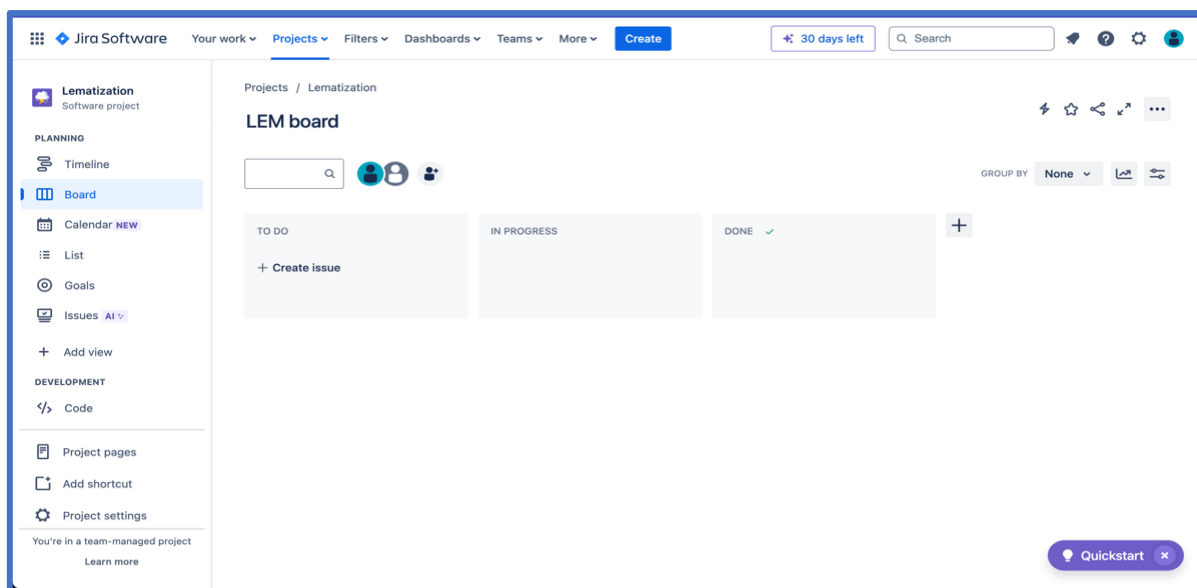


Рис. 2.14 – Створення Kanban-дошки в Jira

Перехід до налаштувань робочого процесу *Workflow*. Щоб додати зміни до робочого процесу, перейдіть до налаштувань вашого проекту. Для цього на лівій бічній панелі опустіться вниз і натисніть «Project settings». У меню налаштувань оберіть вкладку «Workflows». Тут можна переглянути поточний робочий процес або перелік процесів, прив’язаних до різних типів задач (рис. 2.15).



Рис. 2.15 – Огляд існуючого робочого процесу *Workflow*

Огляд існуючого робочого процесу. Після відкриття розділу «Workflows», знайдіть потрібний робочий процес, що використовується для конкретних задач, таких як «Task», «Bug» або «Story», і натисніть «Edit» (Редагувати) біля обраного процесу. Ви побачите робочий процес у вигляді схеми або списку статусів, що описують поточний стан задач, наприклад, «To Do», «In Progress», «In Review» та «Done». Переходи (Transitions) визначають, як задачі можуть змінювати свій статус (рис. 2.16).

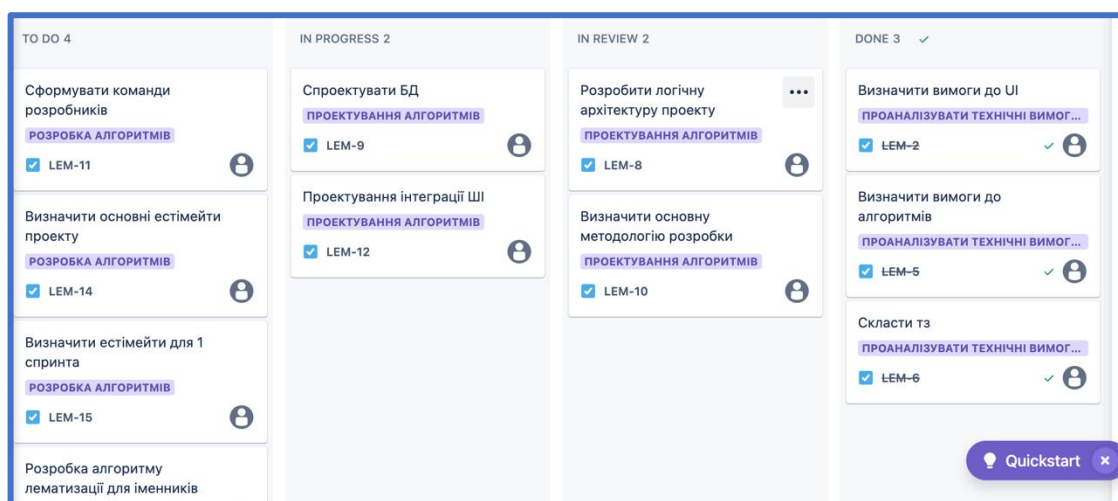


Рис. 2.16 – Основні 4 групи задач «To Do», «In Progress», «In Review», «Done»

Редагування робочого процесу. Для додавання нових етапів робочого процесу натисніть «Add Status» у верхньому правому куті діаграми або списку. Введіть назву нового статусу, наприклад, «In QA» або «Ready for Deployment» (Готово до розгортання), та оберіть його колір для візуальної зручності. Натисніть «Add» для збереження змін.

Щоб налаштувати переходи між статусами, натисніть на стрілку, що з'єднує етапи, або створіть новий перехід, натиснувши на статус, з якого ви хочете зробити перехід. У меню оберіть «Add transition» (Додати перехід), вкажіть кінцевий статус для цього переходу, задайте ім'я, наприклад, «Start Progress» або «Send for Review» (Відправити на перевірку), та натисніть «Add» для збереження.

Для детальної настройки переходів можна додати правила і умови. Натисніть на потрібний перехід і в меню оберіть «Conditions» (Умови),

«Validators» (Перевірки) або «Post functions» (Дії після переходу).

Наприклад, умови дозволяють визначити, хто може виконувати перехід (лише відповідальні за задачу).

Перевірки гарантують, що всі необхідні поля заповнені перед переходом, а дії після переходу можуть автоматично оновлювати поля або відправляти сповіщення.

Якщо необхідно видалити статус або перехід, просто оберіть відповідний елемент і натисніть «Delete». Зважайте на те, що це може вплинути на задачі, що вже використовують цей робочий процес.

Публікація змін у робочому процесі. Після завершення редагування натисніть «Publish Draft» (Опублікувати чернетку) у верхньому правому куті. **Jira** також запропонує створити резервну копію робочого процесу перед публікацією, що буде корисним на випадок непередбачених помилок. Якщо ви хочете застосувати цей процес до різних типів задач, оберіть опцію «Associate» (Прив'язати) у розділі «Workflow schemes» (Схеми робочих процесів), визначте, для яких типів задач цей процес має застосовуватись, і натисніть «Save».

Налаштування колонок на Kanban-дошці. Поверніться до сторінки проекту та відкрийте «Kanban board» на лівій панелі. Для налаштування дошки натисніть на три крапки у правому верхньому куті та виберіть «Board settings». У меню налаштувань оберіть вкладку «Columns», де можна редагувати колонки, перетягуючи статуси між ними, додавати нові за допомогою кнопки «Add column» або видаляти непотрібні за допомогою кнопки «Delete column».

Також ви можете встановити обмеження на кількість задач, що можуть одночасно перебувати в певній колонці. Це допоможе контролювати робоче навантаження. Натисніть на колонку, для якої хочете встановити WIP-ліміт, і введіть відповідне значення.

Тестування та використання робочого процесу. Створіть нову задачу на Kanban-дошці, використовуючи кнопку «Create» у верхньому меню або через бічну панель «Backlog». Для зміни статусу задачі просто перетягніть її між колонками на Kanban-дошці відповідно до налаштованих переходів. Усі дії

відбуватимуться згідно з правилами, які ви налаштували раніше.

Моніторинг та оптимізація робочого процесу. Щоб стежити за прогресом, перейдіть до вкладки «*Reports*» на лівій панелі. Використовуйте звіти «*Control Chart*» або «*Cumulative Flow Diagram*» для аналізу часу виконання задач та контролю за їх накопиченням на різних етапах. Після кількох ітерацій ви зможете оцінити ефективність свого робочого процесу та, за потреби, додати зміни до його налаштувань, повернувшись до розділу «*Workflows*».

Наприклад, після кількох тижнів роботи з новим робочим процесом, команда помітила, що статус «*In Review*» систематично перевантажується, оскільки задачі часто затримуються на цьому етапі. За допомогою «*Cumulative Flow Diagram*» було виявлено, що затримка пов'язана з недостатньою кількістю ресурсів на перевірку. Щоб покращити ситуацію, керівник проекту вирішує додати зміни до робочого процесу, додавши проміжний статус «*Ready for Review*», що допоможе чіткіше розподілити задачі та покращити загальну ефективність команди.

Покроковий алгоритм для визначення лімітів WIP (Work in Progress) у Kanban-дошці в Jira

Спочатку вам потрібно увійти до системи **Jira**. Використайте ваші облікові дані для входу, щоб потрапити до робочого середовища. Після входу у верхньому меню знайдіть пункт «*Projects*», відкрийте випадний список і оберіть потрібний вам *Kanban-проект*, над яким працюєте.

Перехід до налаштувань Kanban-дошки. Для налаштування дошки, в лівій бічній панелі вашого проекту знайдіть і натисніть на «*Kanban board*» (*Kanban-дошка*). Потім у правому верхньому куті дошки натисніть на іконку з трьома крапками «*⋮*», щоб відкрити контекстне меню. У спадному меню оберіть «*Board settings*». Це дозволить вам перейти до конфігурації всіх параметрів вашої *Kanban-дошки* (рис. 2.17).

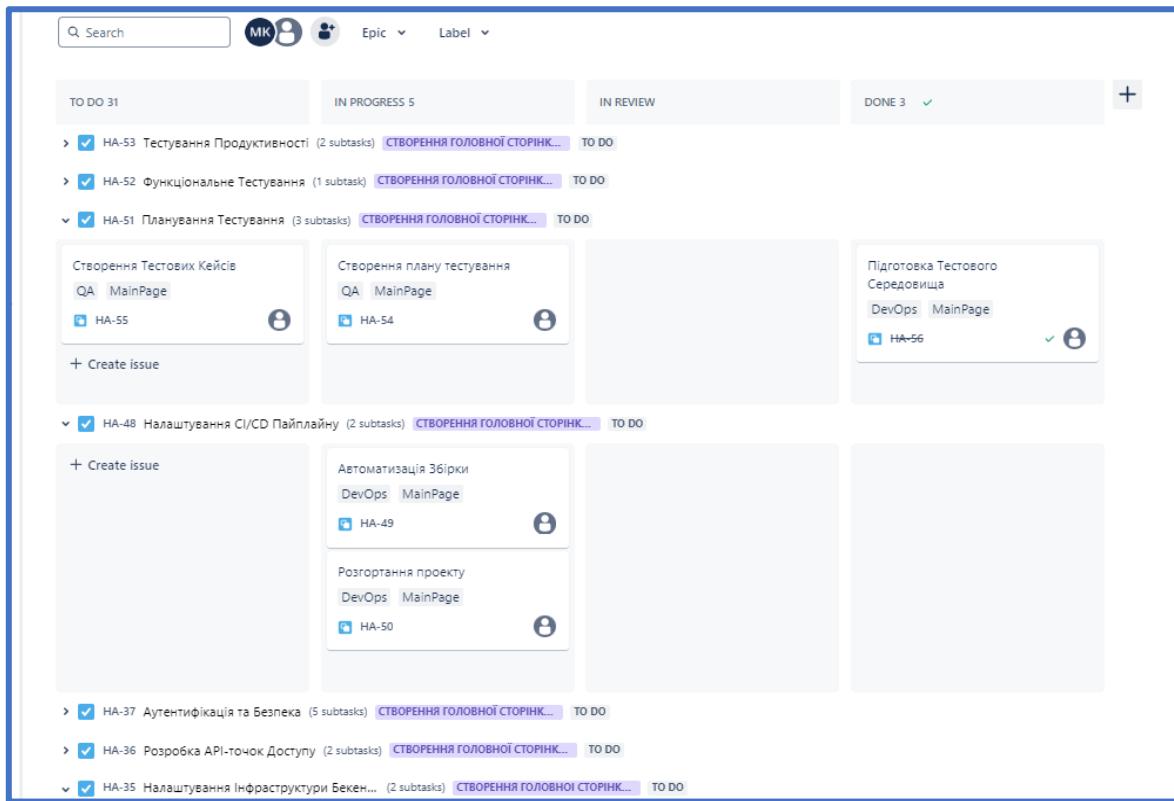


Рис. 2.17 – Практична реалізація вигляду Kanban-дошки

Налаштування колонок для визначення WIP-лімітів. Щоб визначити робоче навантаження на вашу команду, важливо правильно налаштувати ліміти задач на різних етапах. Після переходу в меню налаштувань дошки зліва виберіть вкладку «Columns». Тут ви побачите існуючі колонки, які відповідають статусам на вашій Kanban-дошці, як-от «To Do», «In Progress» та «Done» (рис. 2.18).

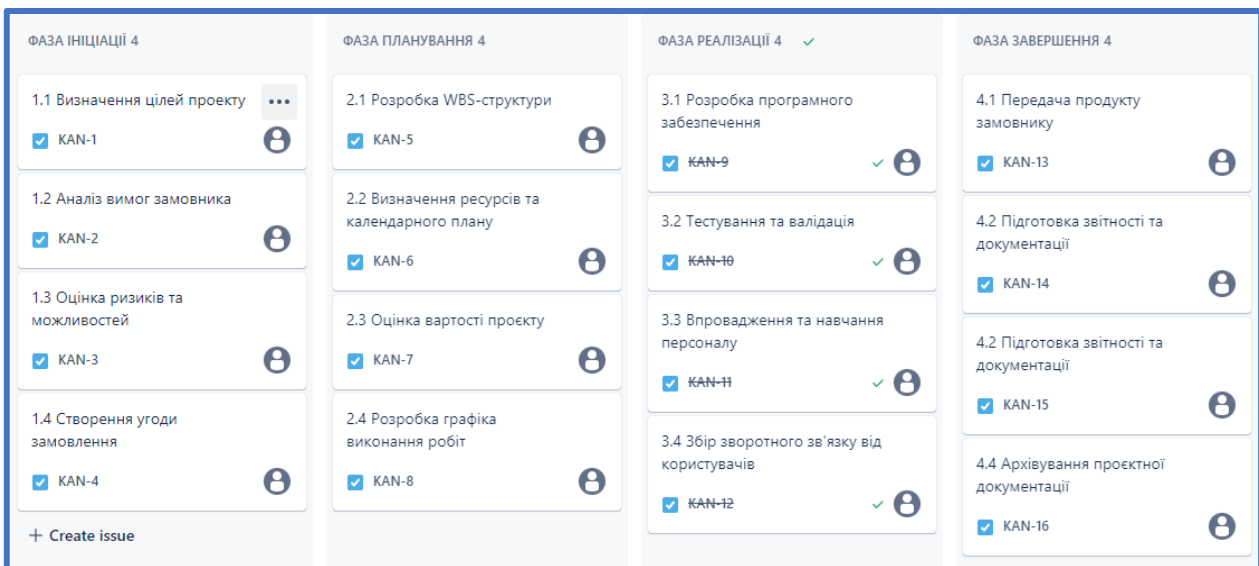


Рис. 2.18 – Заповнені колонки фазами життєвого циклу

Для того, щоб встановити ліміт на кількість задач у колонці, знайдіть потрібну колонку, і у правій частині цього блоку натисніть на поле «*Max issues*» (Макс. кількість задач) або «*WIP limit*» (Ліміт WIP). Введіть максимально допустиму кількість задач, які можуть перебувати на цьому етапі одночасно, наприклад, 5. Усі зміни зберігаються автоматично, і вам не потрібно натискати додаткові кнопки.

Тестування встановлених WIP-лімітів. Після налаштувань поверніться на свою Kanban-дошку через бічне меню або головне меню. Спробуйте додати або перемістити задачі до колонки з встановленим WIP-лімітом. Якщо кількість задач у колонці досягає або перевищує встановлений ліміт, **Jira** автоматично повідомить вас про це або підсвітить відповідну колонку, вказуючи на перевищення ліміту задач.

Моніторинг і управління WIP-лімітами. Під час роботи з Kanban-дошкою варто постійно слідкувати за кількістю задач у кожній колонці. Якщо одна з колонок систематично перевантажена, це може бути ознакою перенавантаження команди. Переглядайте такі ситуації, щоб своєчасно змінити пріоритети або розподілити ресурси.

Рекомендується також регулярно переглядати встановлені WIP-ліміти після кількох ітерацій роботи з проектом. Якщо певні колонки постійно перевантажені або недовантажені, вам може знадобитися змінити ці ліміти. Це допоможе краще збалансувати робоче навантаження команди.

Оптимізація процесу на основі WIP-лімітів. Щоб оцінити ефективність виконання задач і використання WIP-лімітів, скористайтесь звітами у меню «*Reports*», такими як «*Cumulative Flow Diagram*» або «*Control Chart*». Ці звіти дозволяють аналізувати, як швидко виконуються задачі та чи не накопичуються вони на певних етапах. На основі цього аналізу ви зможете визначити, чи потрібно змінювати WIP-ліміти для окремих колонок. Якщо з'ясується, що команда не встигає виконувати задачі у певному статусі, поверніться до налаштувань «*Board settings* → *Columns*» і скоригуйте ліміти WIP.

Наприклад, якщо ваша команда постійно перевантажена на етапі «*In Progress*» і не встигає завершити всі задачі, ви можете знизити ліміт *WIP* для цього статусу з 7 до 5. Це дозволить команді зосередитись на меншій кількості задач і швидше їх завершувати, перш ніж братися за нові (рис. 2.19).

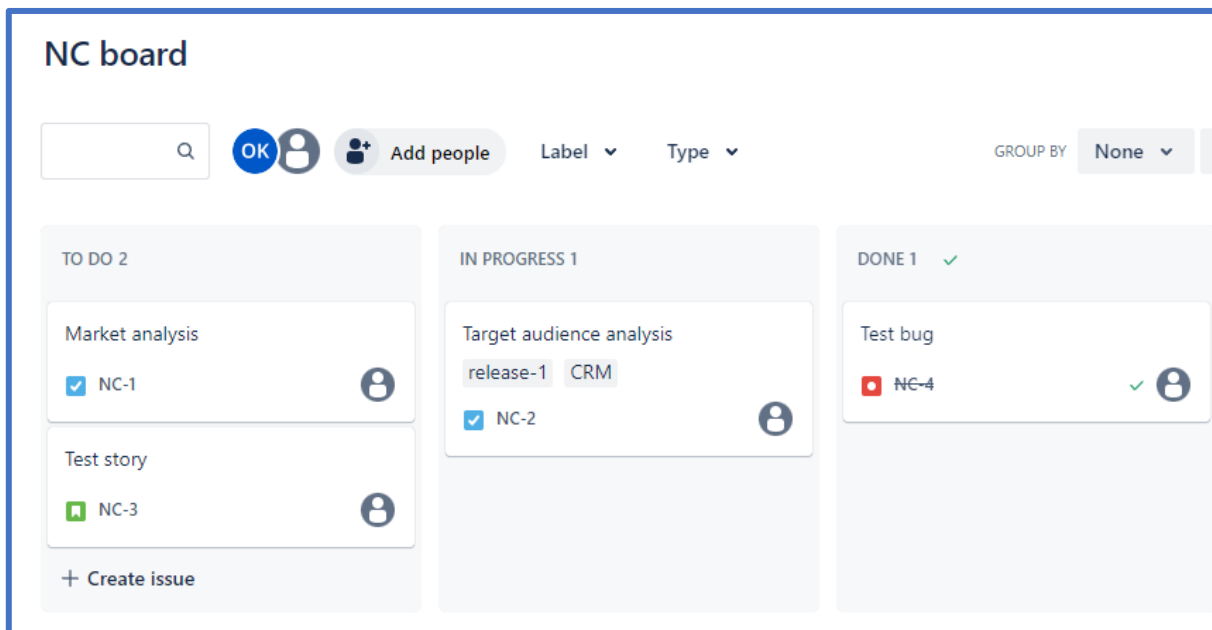


Рис. 2.19 – *WIP*-ліміти окремих колонок

2.3. Управління командами та ролями в Jira

У *Jira* управління командами та ролями забезпечує організований доступ до проєктів і контроль над функціями учасників. Кожному члену команди призначаються конкретні ролі, що визначають їхні права та можливості.

Адміністратор координує всі аспекти проєкту: від налаштування до додавання нових учасників і зміни робочих процесів.

Розробники активно працюють із задачами – створюють, редагують і просуювають їх у процесі виконання.

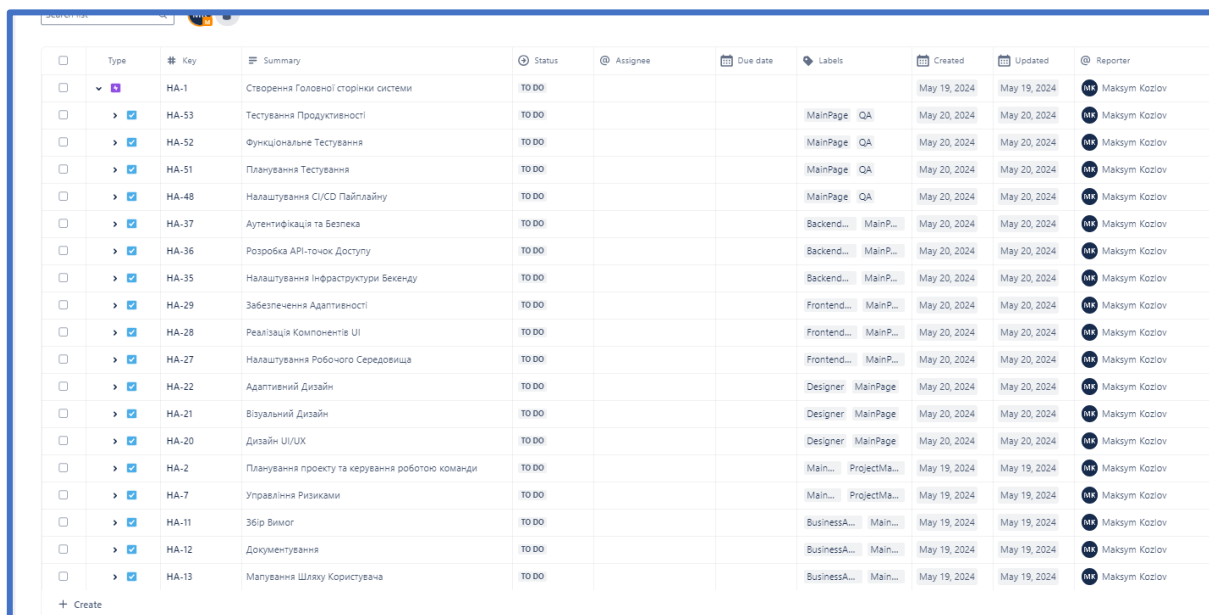
Переглядачі можуть спостерігати за проєктом і його прогресом, але не мають можливості вносити зміни.

Ця система забезпечує чітку структуру доступу, допомагаючи підтримувати баланс між ролями й ефективністю команди.

2.3.1. Призначення ролей та прав доступу

Для призначення ролей та прав доступу в *Jira* вам слід спочатку увійти в систему, ввівши свої облікові дані. Після цього перейдіть до вибору проєкту, натиснувши на вкладку «*Projects*» у верхньому меню. У спадному меню оберіть проєкт, для якого потрібно призначити ролі та права доступу. Далі відкрийте налаштування проєкту, натиснувши на «*Project settings*» у лівій панелі, прокрутивши її донизу.

Тепер перейдіть до розділу «*People*», щоб призначити ролі у проєкті. У налаштуваннях проєкту виберіть цю вкладку з лівого меню. Щоб додати нових користувачів, натисніть кнопку «*Add people*» у верхньому правому куті. Введіть ім'я користувача або його електронну адресу (якщо він вже є в системі). У спадному меню поруч із користувачем оберіть роль. Основні ролі включають «*Administrator*», що надає повний контроль над проєктом і його налаштуваннями, «*Developer*», який має доступ до створення, редагування та виконання задач, і «*Viewer*» (*Переглядач*), що дозволяє лише переглядати задачі без можливості редагування. Натисніть «*Add*», щоб додати користувача до проєкту з обраною роллю (рис. 2.20-2.21).



<input type="checkbox"/>	Type	# Key	Summary	Status	Assignee	Due date	Labels	Created	Updated	Reporter
<input type="checkbox"/>		NA-1	Створення Головної сторінки системи	TO DO				May 19, 2024	May 19, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-53	Тестування Продуктивності	TO DO			MainPage QA	May 20, 2024	May 20, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-52	Функціональне Тестування	TO DO			MainPage QA	May 20, 2024	May 20, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-51	Планування Тестування	TO DO			MainPage QA	May 20, 2024	May 20, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-48	Налаштування CI/CD Пайплайну	TO DO			MainPage QA	May 20, 2024	May 20, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-37	Аутифікація та безпека	TO DO			Backend... MainP...	May 20, 2024	May 20, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-36	Розробка API-точок Доступу	TO DO			Backend... MainP...	May 20, 2024	May 20, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-35	Налаштування Інфраструктури Бекенду	TO DO			Backend... MainP...	May 20, 2024	May 20, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-29	Забезпечення Адаптивності	TO DO			Frontend... MainP...	May 20, 2024	May 20, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-28	Реалізація Компонентів UI	TO DO			Frontend... MainP...	May 20, 2024	May 20, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-27	Налаштування Робочого Середовища	TO DO			Frontend... MainP...	May 20, 2024	May 20, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-22	Адаптивний Дизайн	TO DO			Designer MainPage	May 20, 2024	May 20, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-21	Візуальний Дизайн	TO DO			Designer MainPage	May 20, 2024	May 20, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-20	Дизайн UI/UX	TO DO			Designer MainPage	May 20, 2024	May 20, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-2	Планування проєкту та керування роботою команди	TO DO			Main... ProjectMa...	May 19, 2024	May 19, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-7	Управління Ризиками	TO DO			Main... ProjectMa...	May 19, 2024	May 19, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-11	Збір Вимог	TO DO			BusinessA... Main...	May 19, 2024	May 19, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-12	Документування	TO DO			BusinessA... Main...	May 19, 2024	May 19, 2024	Maksym Kozlov
<input type="checkbox"/>	> <input checked="" type="checkbox"/>	NA-13	Мапування Шляху Користувача	TO DO			BusinessA... Main...	May 19, 2024	May 19, 2024	Maksym Kozlov

Рис. 2.20 – Розподілення задач між членами команди

Щоб у дошці було легше орієнтуватись варто додати лейбли, поки що з однією фічею, над якою почали працювати, тобто Main Page, а також лейбли на позначення відповідального за завдання.

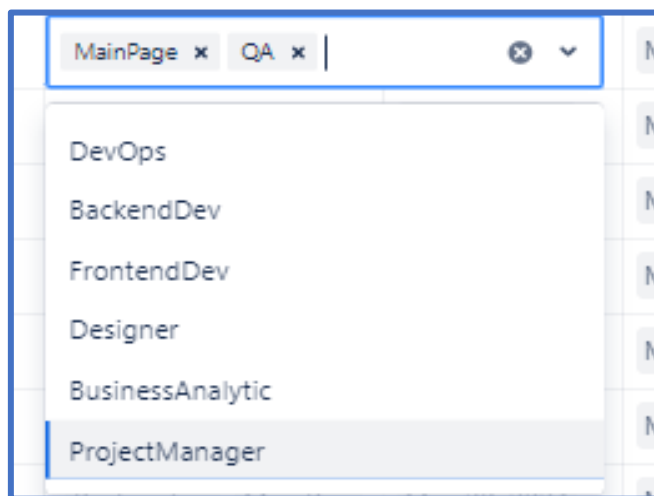
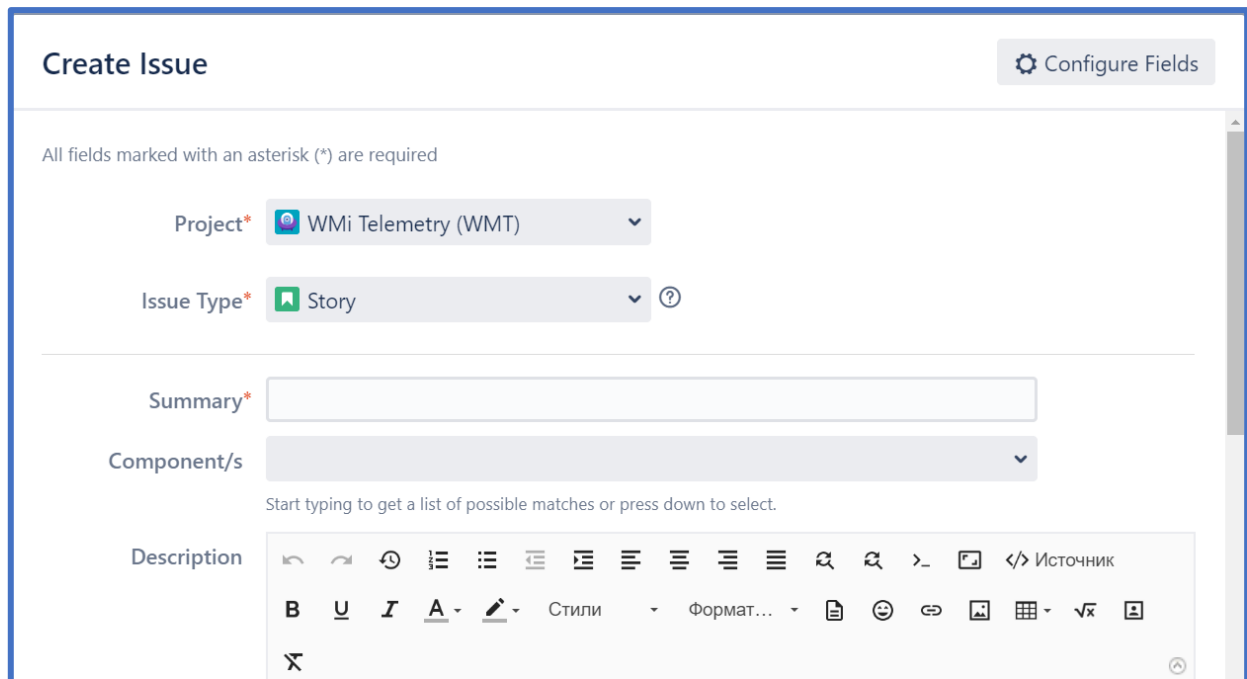


Рис. 2.21 – Позначення відповідального за завдання

Далі налаштуйте права доступу, перейшовши до розділу «*Permissions*» (Права доступу) у меню «*Project settings*». Тут ви побачите «*Permission Scheme*» (Схема прав доступу), яка визначає, які ролі мають доступ до різних дій у проекті, таких як створення, редагування, зміна статусу тощо. Щоб редагувати права доступу, натисніть «*Actions*» у верхньому правому куті та виберіть «*Edit Permissions*» (Редагувати права). Залежно від вашої схеми доступу, ви можете призначати ролі на такі дії, як «*Create Issues*», «*Edit Issues*», «*Assign Issues*», «*Resolve Issues*» та інші. Для кожного «елементу доступу» натисніть «*Edit*» поруч із потрібним пунктом, у спадному меню оберіть ролі, які повинні мати доступ до цієї дії, після чого збережіть зміни (рис. 2.22).

Якщо необхідно, ви також можете додати нові ролі. Для цього перейдіть до глобальних налаштувань **Jira**, натиснувши на значок шестерні «*Settings*» у верхньому правому куті та вибравши «*System*». У лівій частині сторінки знайдіть розділ «*Security*» і виберіть «*Project roles*». Натисніть «*Add project role*» у верхньому правому куті, введіть назву нової ролі (наприклад, QA Engineer або Product Owner) та натисніть «*Add project role*». Поверніться до «*Project settings*

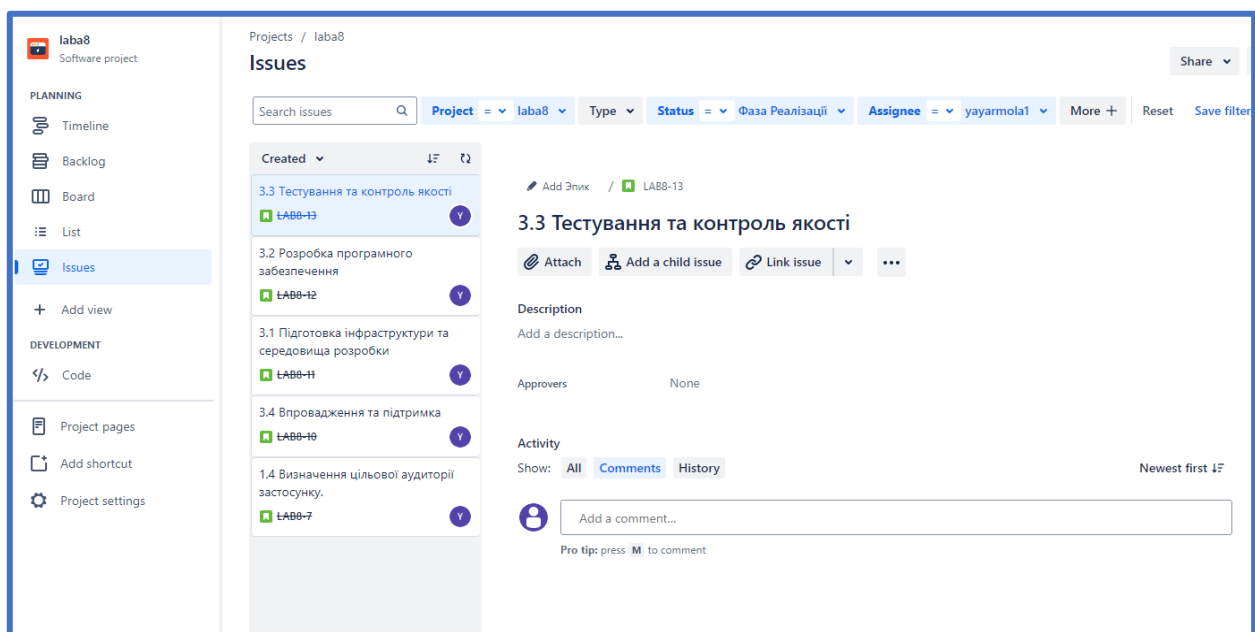
→ *People*» і призначте новостворену роль потрібним користувачам, використовуючи кнопку «Add people» (рис. 2.23).



The screenshot shows the 'Create Issue' interface. At the top right is a 'Configure Fields' button. Below it, a note states: 'All fields marked with an asterisk (*) are required'. The form contains the following fields:

- Project***: A dropdown menu with 'WMT Telemetry (WMT)' selected.
- Issue Type***: A dropdown menu with 'Story' selected.
- Summary***: A text input field.
- Component/s**: A dropdown menu.
- Description**: A rich text editor with a toolbar containing icons for undo, redo, refresh, bulleted list, numbered list, decrease indent, increase indent, bold, italic, underline, link, unlink, link preview, source code, and a 'Source' button.

Рис. 2.22 – Поля для створення завдання



The screenshot displays the 'Issues' page for the 'laba8' project. The left sidebar shows navigation options: PLANNING (Timeline, Backlog, Board, List), Issues (selected), Add view, DEVELOPMENT (Code), Project pages, Add shortcut, and Project settings. The main area shows a list of issues filtered by Project (laba8), Type, Status (Фаза Реалізації), and Assignee (yayarmolat). The selected issue is '3.3 Тестування та контроль якості' (LAB8-13). The right panel shows the issue details, including the title, description, approvers, and activity. The activity section shows a comment input field and a 'Pro tip: press IM to comment' message.

Рис. 2.23 – Фільтрування задачі за різними критеріями

Після цього перевірте права доступу, повернувшись на сторінку проекту. Переконайтеся, що користувачі можуть виконувати дії відповідно до призначених ролей, наприклад, чи можуть вони створювати задачі, редагувати їх тощо. У будь-який момент ви можете оновлювати ролі та права, повертаючись

до розділу People або Permissions у налаштуваннях проєкту (рис. 2.24).

Наприклад, якщо ви додали нового користувача як «*Developer*», він зможе створювати задачі і редагувати їх, а якщо ви призначили його як «*Viewer*», то отримає лише можливість перегляду задач без права редагування. Таким чином, правильно налаштовані ролі та права доступу в *Jira* сприятимуть ефективній роботі команди над проєктом.

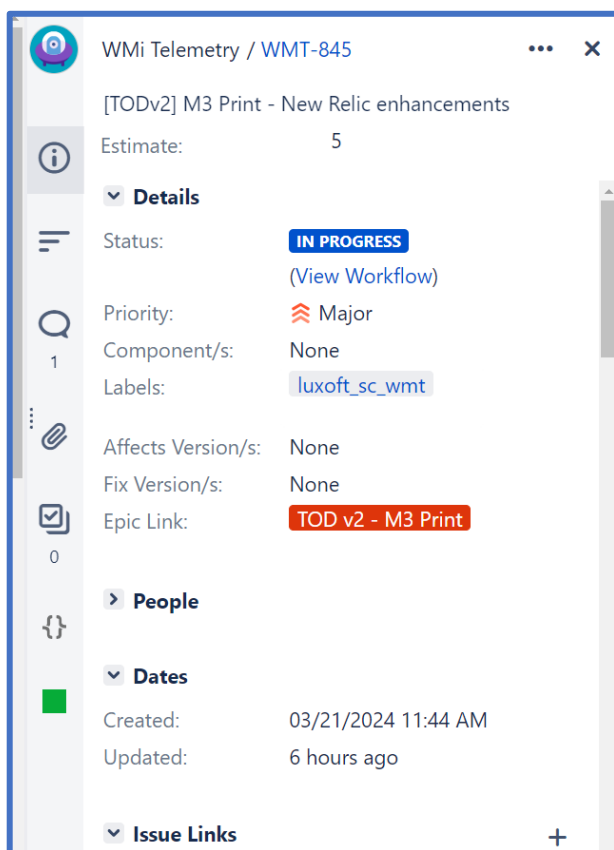


Рис. 2.24 – Приклад створеної задачі

2.3.2. Управління робочими процесами

Управління робочими процесами в *Jira* (Jira Workflows) дає можливість налаштувати різні етапи життєвого циклу задачі від створення до завершення. Ось покроковий алгоритм для управління робочими процесами (Jira Workflows):

1. Здійсніть вибір проєкту. У верхньому меню натисніть на «*Projects*». Далі, у меню, яке з'явиться, оберіть потрібний проєкт, для якого ви хочете змінити робочий процес (рис. 2.25).

2. Відкрийте налаштування проєкту. У лівій панелі проєкту натисніть «*Project settings*».
3. Здійсніть перехід до налаштувань робочого процесу. У налаштуваннях проєкту у лівому меню знайдіть і оберіть вкладку «*Workflows*», де будуть відображатися всі робочі процеси, пов'язані з проєктом.

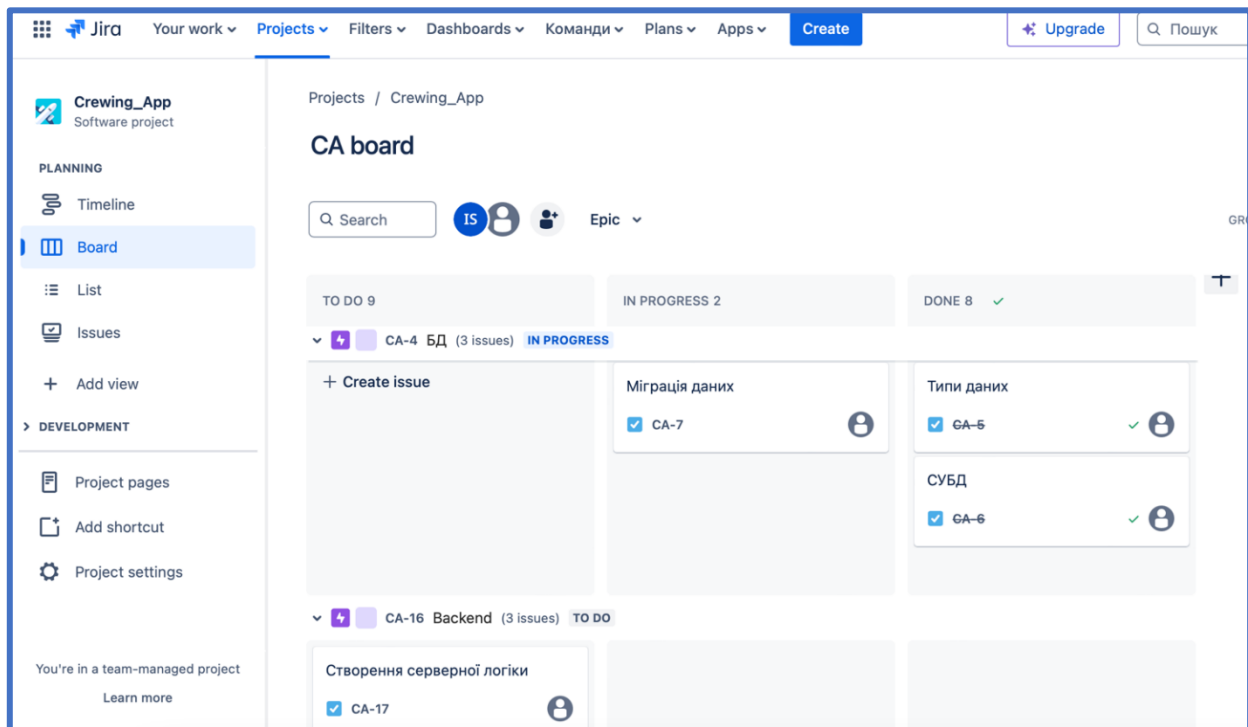


Рис. 2.25 – Управління робочими процесами в *Jira*

Огляд робочих процесів у *Jira* показує, що кожен тип задачі може мати свій власний робочий процес. Якщо всі задачі в проєкті використовують один і той самий процес, він буде відображений як спільний. Щоб додати зміни до робочого процесу, знайдіть у списку робочих процесів той, який потрібно змінити, і натисніть на нього. У правому верхньому куті сторінки натисніть кнопку «*Edit*», що відкриє робочий процес у режимі схеми «*Scheme*».

У режимі редагування ви можете додавати нові статуси, натиснувши кнопку «*Add status*» у верхньому правому куті. Введіть назву нового статусу, наприклад, «*In Review*» або «*Blocked*», виберіть колір для цього статусу (за бажанням) і натисніть «*Add*». Щоб редагувати існуючі статуси, натисніть на статус, який

потрібно змінити, і у спадному меню оберіть «*Edit*». Відредагуйте назву чи колір статусу та натисніть «*Save*».

Додавання переходів між статусами також є важливим кроком. Щоб створити перехід між двома статусами, натисніть на стрілку між ними або виберіть «*Add transition*» (*Додати перехід*). Введіть назву переходу, наприклад, «*Move to In Progress*» (*Перемістити в роботу*) або «*Complete Task*», а потім натисніть «*Add*», щоб зберегти перехід. Щоб налаштувати правила для переходів, натисніть на існуючий перехід у схемі і виберіть «*Edit*». Тут ви можете додати «*Conditions*» (*Умови*), «*Validators*» (*Валідатори*) або «*Post Functions*» (*Пост-функції*). Умови визначають, хто може виконувати перехід, наприклад, лише розробники або менеджери. Валідатори перевіряють, чи виконуються умови для здійснення переходу, наприклад, чи всі обов'язкові поля заповнені. Пост-функції автоматизують дії після переходу, наприклад, автоматичне призначення задачі чи зміна пріоритету.

Після внесення всіх змін у верхньому правому куті натисніть «*Publish Draft*» (*Опублікувати чернетку*), щоб зберегти зміни. Якщо ви створили новий робочий процес, виберіть «*Associate issue types with workflow*» (*Асоціювати типи задач з робочим процесом*). У вікні, що з'явиться, виберіть типи задач, для яких буде використовуватися цей робочий процес, наприклад, «*Bug*» або «*Task*», і натисніть «*Associate*» (*Асоціювати*).

Поверніться до проєкту, натиснувши «*Back to Project*» у верхньому лівому куті або використовуючи меню «*Projects*». Створіть нову задачу або перейдіть до існуючої, щоб перевірити, як нові статуси та переходи працюють для цієї задачі. Якщо потрібно додати зміни до робочого процесу, просто повторіть попередні кроки. Після кожної зміни не забувайте публікувати оновлення, натискаючи «*Publish Draft*».

Для аналізу ефективності робочого процесу використовуйте звіти, такі як «*Control Chart*» або «*Cumulative Flow Diagram*». Цей алгоритм допоможе вам керувати робочими процесами в **Jira**, налаштовувати етапи задач, створювати переходи між статусами і контролювати життєвий цикл задач.

Наприклад, після редагування робочого процесу ви можете додати статус «*In Review*» та перехід «*Complete Task*», що дозволить вашій команді легко відстежувати прогрес у виконанні задач і ефективно керувати ними.

2.2. Огляд робочих процесів

Кожен тип задач в **Jira** може мати свій власний робочий процес. Якщо всі задачі використовують один і той самий процес, він відображається як спільний. Щоб редагувати робочий процес, знайдіть у списку робочих процесів той, який потрібно змінити, і натисніть на нього. У правому верхньому куті сторінки ви побачите кнопку для редагування «*Edit*»; натисніть на неї, щоб відкрити робочий процес у режимі схеми «*Scheme*».

У режимі редагування можна додати нові статуси, натиснувши кнопку «*Add status*» у верхньому правому куті. Введіть назву нового статусу, наприклад, «*In Review*» або «*Blocked*». Виберіть колір для цього статусу (за бажанням) і натисніть «*Add*», щоб зберегти його. Якщо потрібно редагувати існуючі статуси, просто натисніть на статус, який потрібно змінити, і у спадному меню оберіть «*Edit*». Тут ви можете відредагувати назву чи колір статусу, а потім натисніть «*Save*».

Додавання переходів між статусами є важливою частиною налаштування робочого процесу. Щоб створити перехід між двома статусами, натисніть на стрілку між ними або виберіть «*Add transition*». Введіть назву переходу, наприклад, «*Move to In Progress*» (*Перемістити в роботу*) або «*Complete Task*», і натисніть «*Add*», щоб зберегти перехід. Щоб налаштувати правила для переходів, натисніть на існуючий перехід у схемі та виберіть «*Edit*». Ви можете додати «*Conditions*» (*Умови*), «*Validators*» (*Валідатори*) або «*Post Functions*» (*Пост-функції*). Умови визначають, хто може виконувати перехід, наприклад, лише розробники або менеджери. Валідатори перевіряють, чи виконуються умови для здійснення переходу, наприклад, чи всі обов'язкові поля заповнені. Пост-функції автоматизують дії після переходу, такі як автоматичне призначення задачі чи зміна пріоритету.

Після внесення всіх змін, у верхньому правому куті натисніть «*Publish Draft*» для збереження змін. Якщо ви створили новий робочий процес, оберіть «*Associate issue types with workflow*». У вікні, що відкриється, виберіть типи задач, для яких буде використовуватися цей робочий процес, такі як «*Bug*» або «*Task*», і натисніть «*Associate*».

Поверніться до проєкту, натиснувши «*Back to Project*» у верхньому лівому куті або використовуючи меню «*Projects*». Створіть нову задачу або перейдіть до існуючої, щоб перевірити, як нові статуси та переходи працюють для цієї задачі. Якщо потрібно додати зміни до робочого процесу, просто повторіть попередні кроки. Не забувайте публікувати оновлення, натискаючи «*Publish Draft*» після кожної зміни.

Для аналізу ефективності робочого процесу скористайтеся звітами, такими як «*Control Chart*» або «*Cumulative Flow Diagram*».

Наприклад, ви можете створити новий статус «*In Review*» та перехід «*Complete Task*», що допоможе вашій команді легко відстежувати прогрес виконання задач і забезпечити контроль за їх життєвим циклом.

2.4. Інтеграції та автоматизація процесів в Jira

Інтеграції та автоматизація процесів у **Jira** дозволяють підвищити ефективність командної роботи, зменшуючи ручні зусилля та помилки.

Jira підтримує інтеграцію з безліччю сторонніх сервісів і платформ, таких як *Slack*, *GitHub*, *Trello*, *Confluence* та багато інших. Це дозволяє команді з легкістю обмінюватися інформацією, отримувати сповіщення та координувати дії без необхідності переходити між різними інструментами. Інтеграції забезпечують автоматичний імпорт і синхронізацію даних, що зменшує час на виконання рутинних завдань.

Автоматизація в **Jira** реалізується через функцію «*Automation*», яка дозволяє створювати правила для виконання рутинних дій на основі тригерів.

Наприклад, автоматично призначати задачі певному користувачу при зміні статусу, відправляти нагадування або створювати нові задачі на основі подій. Це

не лише скорочує час, який команда витрачає на адміністративні задачі, але й підвищує прозорість процесів, оскільки всі дії виконуються автоматично та за заздалегідь визначеними умовами.

Інтеграції та автоматизація в *Jira* є потужними інструментами, які допомагають оптимізувати робочі процеси, зменшити кількість помилок і підвищити продуктивність команди. Завдяки цим можливостям, команди можуть зосередитися на більш важливих завданнях, а не на рутинних операціях.

2.4.1. Інтеграція з інструментами для розробників (GitHub, Bitbucket)

Інтеграція з *GitHub* та *Bitbucket* у *Jira* дозволяє командам автоматизувати процеси, пов'язані з розробкою, такими як відстеження змін коду, коментарів до задач і автоматизація звітності. Ось покроковий алгоритм інтеграції цих інструментів у *Jira*.

Щоб розпочати роботу з інтеграцією *GitHub* у *Jira*, спочатку необхідно увійти в систему. Використовуйте свої облікові дані, щоб отримати доступ до платформи *Jira*.

Після входу у верхньому меню знайдіть вкладку «*Projects*» та натисніть на неї. Виберіть проєкт, для якого ви плануєте налаштувати інтеграцію. Далі, у лівій панелі, натисніть на «*Project settings*» (*Налаштування проєкту*), щоб потрапити до параметрів налаштувань.

Тепер перейдіть до інтеграцій. У лівому меню знайдіть розділ «*Integrations*», який може бути позначено як «*DVCS accounts*» (*Акаунти DVCS*) або «*Development tools*», залежно від версії *Jira*, яку ви використовуєте.

На цьому етапі вам потрібно додати *GitHub*-акаунт. Натисніть на кнопку «*Link GitHub account*» (*Пов'язати акаунт GitHub*). У відповідь з'явиться вікно, в якому вам потрібно ввести *URL* вашого *GitHub* репозиторію. Також зазначте права доступу, які необхідно надати, зазвичай це доступ на читання. Після цього натисніть «*Connect*». Якщо виникне потреба, надайте необхідні дозволи для інтеграції на *GitHub*.

Тепер, коли акаунт *GitHub* підключено, оберіть репозиторії, які потрібно інтегрувати з *Jira*. Це дозволить вам відстежувати зміни та коміти, пов'язані з вашими задачами. Після вибору репозиторіїв натисніть «*Save*» або «*Done*», щоб зберегти налаштування інтеграції.

Для перевірки успішності інтеграції перейдіть до будь-якої задачі в *Jira*. Відкрийте її та перевірте вкладку «*Development*». Тут ви повинні побачити коміти, «*pull requests*» (заявки на злиття) та іншу інформацію з *GitHub*, що підтверджує успішне налаштування інтеграції.

Наприклад, якщо ви інтегрували репозиторій з задачею, пов'язаною із виправленням помилки, ви зможете побачити всі коміти, які стосуються цієї задачі, безпосередньо в, що суттєво спростить управління проектом.

Інтеграція з Bitbucket у Jira

Щоб почати інтеграцію між *Jira* та *Bitbucket*, спочатку необхідно увійти до системи. Введіть свої облікові дані, щоб отримати доступ до платформи *Jira*.

Після успішного входу у верхньому меню знайдіть розділ «*Projects*» та натисніть на нього. Оберіть проєкт, для якого плануєте налаштувати інтеграцію. Наступним кроком буде відкриття налаштувань проєкту: у лівій панелі натисніть на «*Project settings*».

Тепер вам потрібно перейти до інтеграцій. У лівому меню знайдіть пункт «*Integrations*», який може бути позначений також як «*DVCS accounts*» або «*Development tools*», в залежності від версії *Jira*, яку ви використовуєте.

Тепер розпочнемо процес додавання *Bitbucket*-акаунту. Для цього натисніть на кнопку «*Link Bitbucket account*». Після цього з'явиться спливаюче вікно, де вам потрібно ввести URL вашого *Bitbucket* репозиторію. Також важливо зазначити права доступу, які ви плануєте надати. Зазвичай, це доступ на читання. Після введення всіх необхідних даних натисніть «*Connect*». У разі потреби, вам може знадобитися надати додаткові дозволи для інтеграції на платформі *Bitbucket*.

Тепер, коли акаунт *Bitbucket* підключено, оберіть репозиторії, які ви хочете інтегрувати з *Jira*. Це дозволить вам легко відстежувати зміни та коміти,

пов'язані з вашими задачами. Після вибору відповідних репозиторіїв натисніть «*Save*» або «*Done*», щоб зберегти налаштування інтеграції.

Щоб перевірити успішність інтеграції, перейдіть до будь-якої задачі в **Jira**. Відкрийте її та знайдіть вкладку «*Development*». Тут ви повинні побачити коміти, «*pull requests*» та іншу інформацію з *Bitbucket*, що свідчить про успішну інтеграцію.

Наприклад, якщо ви працюєте над виправленням помилки у задачі, інтеграція дозволить вам безпосередньо в **Jira** бачити всі коміти, які стосуються цієї задачі. Це суттєво спростить вашу роботу, оскільки ви зможете слідкувати за прогресом команди без необхідності переходити між різними платформами, що підвищить продуктивність та забезпечить прозорість у процесах розробки.

2.4.2. Автоматизація через правила Jira. Створення тригерів для автоматизації

Створення тригерів для автоматизації в **Jira** – це потужний інструмент, який дозволяє налаштувати автоматизовані дії, які виконуються у відповідь на певні події. Це допомагає зменшити рутинні завдання та підвищити продуктивність команди. Ось покроковий алгоритм, який допоможе вам створити тригери автоматизації.

Почніть з входу до **Jira**, використовуючи свої облікові дані. Це забезпечить вам доступ до всіх функцій системи. Після входу перейдіть до верхнього меню, де знайдете вкладку «*Projects*». Виберіть проєкт, для якого плануєте налаштувати автоматизацію. Це важливо, оскільки правила автоматизації діють лише в межах конкретного проєкту.

Далі відкрийте налаштування проєкту, натиснувши на «*Project settings*» в лівій панелі. Тут ви знайдете розділ «*Automation*», де можете створити нові правила автоматизації (рис. 2.26).

Щоб розпочати, натисніть кнопку «*Create*» (*Створити правило*) у верхньому правому куті. З'явиться вікно з доступними тригерами. Наприклад, тригер «*Issue created*» активується при створенні нової задачі, «*Issue transitioned*»

– при зміні статусу задачі, а «*Field value changed*» – коли змінюється значення конкретного поля. Виберіть потрібний тригер, натиснувши на нього, і налаштуйте його, якщо це необхідно, додаючи умови, такі як вибір певного типу задачі чи проєкту. Після завершення натисніть «*Save*».

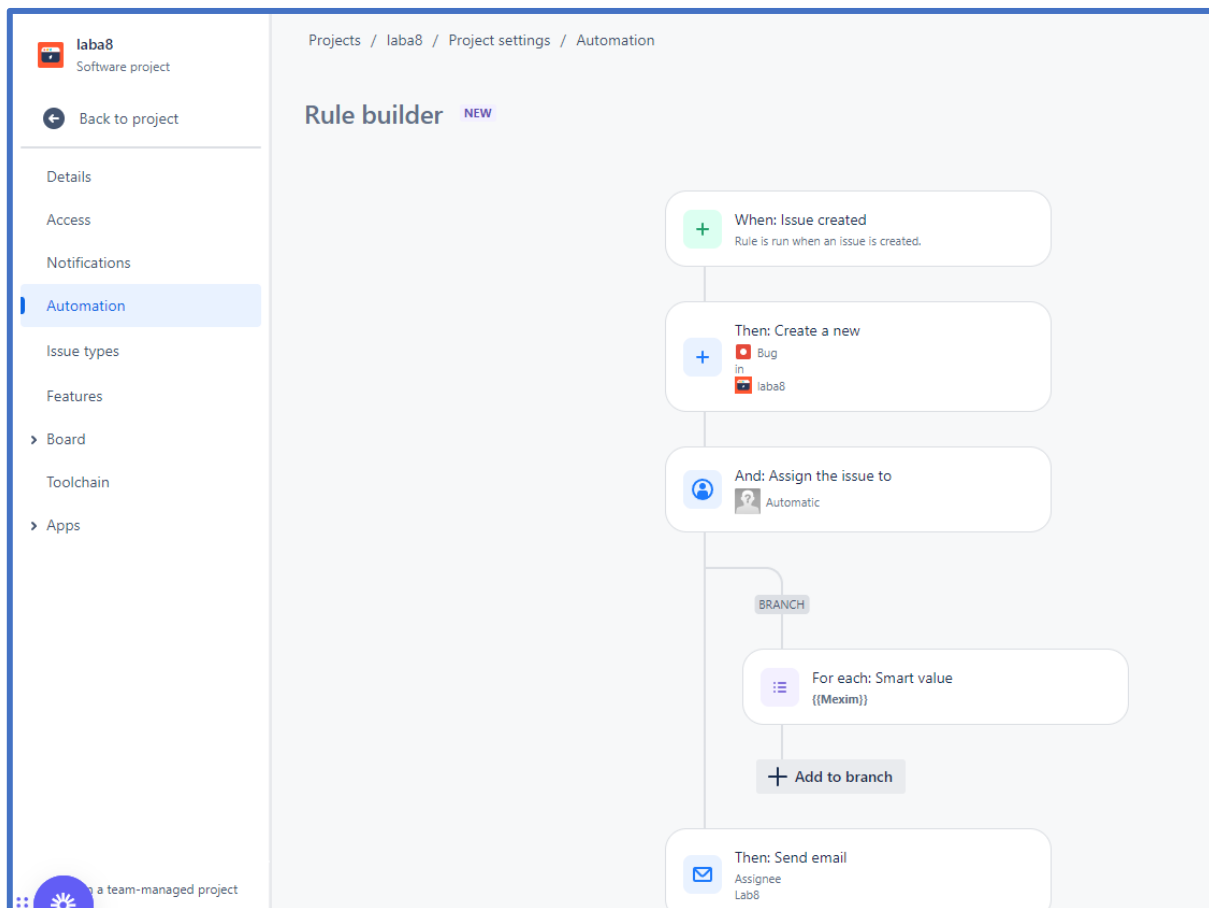


Рис. 2.26 – Розділ «Automation»

Наступним кроком є додавання умов, які визначають, коли тригер має активуватися. Для цього натисніть «*Add condition*» (Додати умову). Ви можете вибрати, наприклад, умову «*Issue matches JQL*» (Задача відповідає JQL), яка перевіряє, чи відповідає задача певному JQL-запиту, або «*User is in group*» (Користувач у групі), щоб перевірити, чи є користувач у визначеній групі. Налаштуйте умову відповідно до ваших потреб і знову натисніть «*Save*».

Тепер перейдемо до додавання дій. Натисніть «*Add action*» та виберіть, що має відбутися, коли тригер активується. Серед доступних дій можуть бути «*Send e-mail*» для автоматичного сповіщення, «*Edit issue*» для зміни полів задачі або

«*Transition issue*» для зміни її статусу. Налаштуйте параметри дії та натисніть «*Save*».

Після налаштування тригера, умов і дій перегляньте своє правило, щоб переконатися, що все виглядає правильно. У верхньому полі введіть назву для вашого правила, наприклад, «*Auto-Assign New Tasks*» (*Автоматичне призначення нових задач*). Коли все готово, натисніть «*Turn it on*» або «*Save*», щоб активувати правило автоматизації.

Щоб перевірити, чи ваше правило працює, створіть або оновіть задачу та подивіться, чи тригер та пов'язані дії спрацьовують коректно. У розділі автоматизації ви також можете переглянути «*Audit log*» (*Журнал перевірки*), щоб побачити виконання правил і виявити можливі проблеми.

Якщо вам потрібно додати зміни до правила, поверніться до налаштувань автоматизації, виберіть правило з списку та натисніть на нього для редагування. Додайте потрібні зміни та знову натисніть «*Save*».

Таким чином, цей покроковий алгоритм дозволить вам ефективно налаштувати тригери для автоматизації в *Jira*, що значно спростить рутинні процеси та підвищить ефективність роботи вашої команди. **Наприклад**, ви можете створити правило, яке автоматично призначає нові задачі певному користувачу, що дозволяє уникнути затримок у роботі та оптимізує управління проектами.

Розглянемо декілька **прикладів** використання тригерів для автоматизації:

1. Автоматичне повідомлення про нову задачу. Коли в проєкті створюється нова задача, можна налаштувати тригер «*Issue created*» для автоматичного надсилання повідомлення всім членам команди. Це дозволяє оперативно інформувати команду про нові задачі.
2. Автоматичне переведення задач у статус «*In Progress*». Якщо ви використовуєте тригер «*Issue transitioned*», ви можете налаштувати автоматичне переведення задач у статус «*In Progress*», коли задача переходить з «*Open*» у «*In Progress*». Це дозволяє швидше відслідковувати статуси задач.

3. Оновлення статусу залежних задач. Використовуючи тригер «*Field value changed*», можна автоматично змінювати статус задачі, якщо поле «*Basic task*» (Базова задача) було оновлено. Це особливо корисно для управління проектами, що мають складну структуру залежностей.
4. Автоматичне закриття задач після виконання: Коли задача завершується, можна використовувати тригер «*Issue transitioned*» для автоматичного закриття пов'язаних задач. Наприклад, якщо основна задача переходить у статус «*Done*», то пов'язані задачі автоматично переводяться у статус «*Done*» (рис. 2.27).

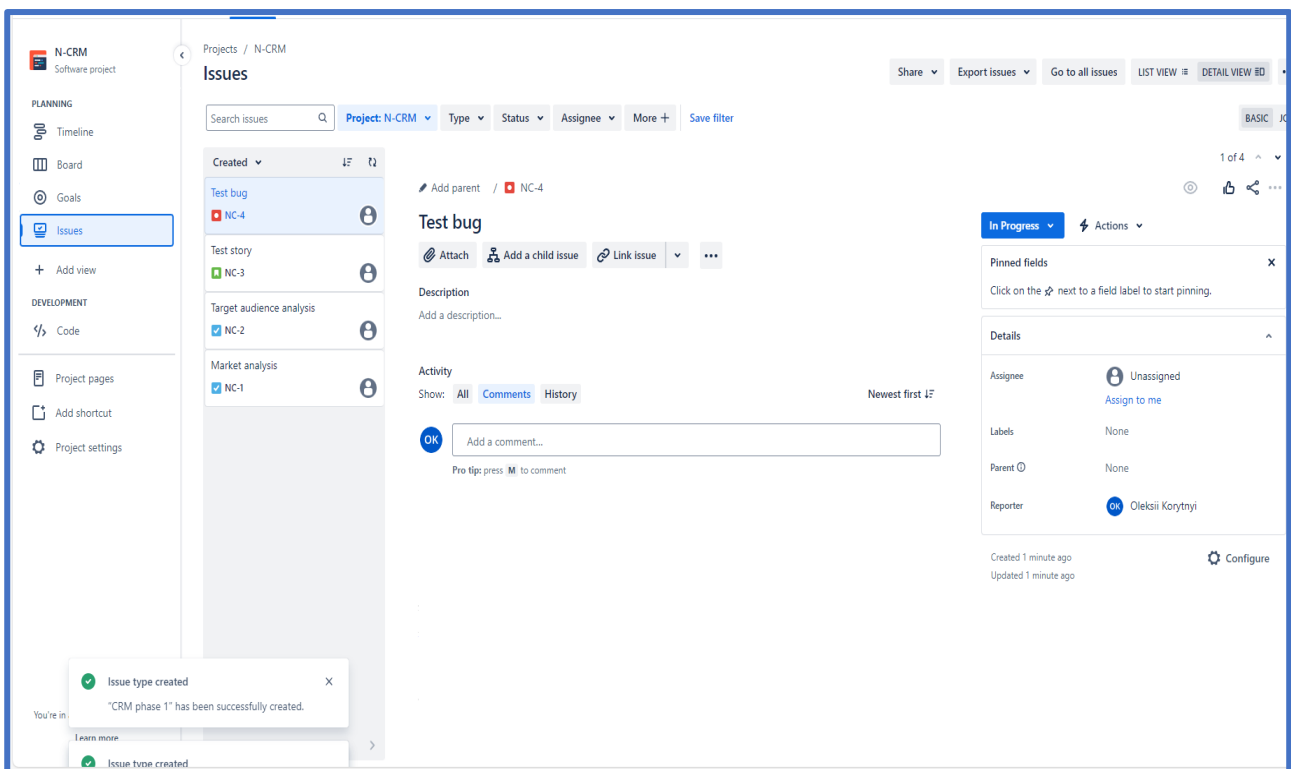


Рис. 2.27 – Створення різних типів задач

Завдяки таким тригерам, ваша команда зможе зекономити час і ресурси, що значно підвищить загальну продуктивність та якість роботи. Автоматизація дозволяє зменшити людський фактор у виконанні рутинних задач, що є важливим аспектом сучасного управління проектами.

2.4.3. Автоматичне оновлення статусів задач в Jira

Автоматичне оновлення статусів задач у *Jira* є важливим інструментом, який дозволяє зменшити рутинні дії та підвищити ефективність управління проектами. Цей процес допомагає вашій команді зосередитися на важливих завданнях, залишаючи автоматизацію для виконання повторюваних дій. Ось покрокова інструкція, як налаштувати автоматичне оновлення статусів задач у *Jira*.

Перш ніж почати, вам потрібно увійти до системи *Jira*. Введіть свої облікові дані, щоб отримати доступ до всіх функцій платформи. Після входу у верхньому меню знайдіть вкладку «*Projects*» та виберіть проєкт, для якого плануєте налаштувати автоматизацію. Це важливо, адже правила автоматизації застосовуються лише до конкретного проєкту (рис. 2.28).

Далі, для переходу до налаштувань автоматизації, натисніть на «*Project settings*» у лівій панелі. У цьому меню знайдіть і виберіть розділ «*Automation*». Тут ви зможете створити нові правила, що спростять ваші робочі процеси (рис. 2.29).

Щоб розпочати, натисніть кнопку «*Create rule*» у верхньому правому куті.

На екрані з'явиться список доступних тригерів.

Для автоматичного оновлення статусу задачі можна вибрати, наприклад, тригер «*Issue transitioned*», який спрацьовує при зміні статусу задачі, або «*Issue created*», що активується при створенні нової задачі.

Виберіть відповідний тригер, натиснувши на нього, і за потреби додайте додаткові умови, наприклад, виберіть конкретний тип задачі чи проєкт. Після цього не забудьте натиснути «*Save*».

Наступним кроком є додавання умов, які визначають, коли тригер повинен активуватися. Для цього натисніть «*Add condition*» (*Додати умову*) та виберіть тип умови. Наприклад, ви можете вибрати умову «*Issue matches JQL*» (*Задача відповідає JQL*), що перевіряє, чи відповідає задача певному *JQL*-запиту, або «*User is in group*» (*Користувач у групі*), щоб перевірити, чи є користувач у конкретній групі. Налаштуйте параметри умови та натисніть «*Save*».

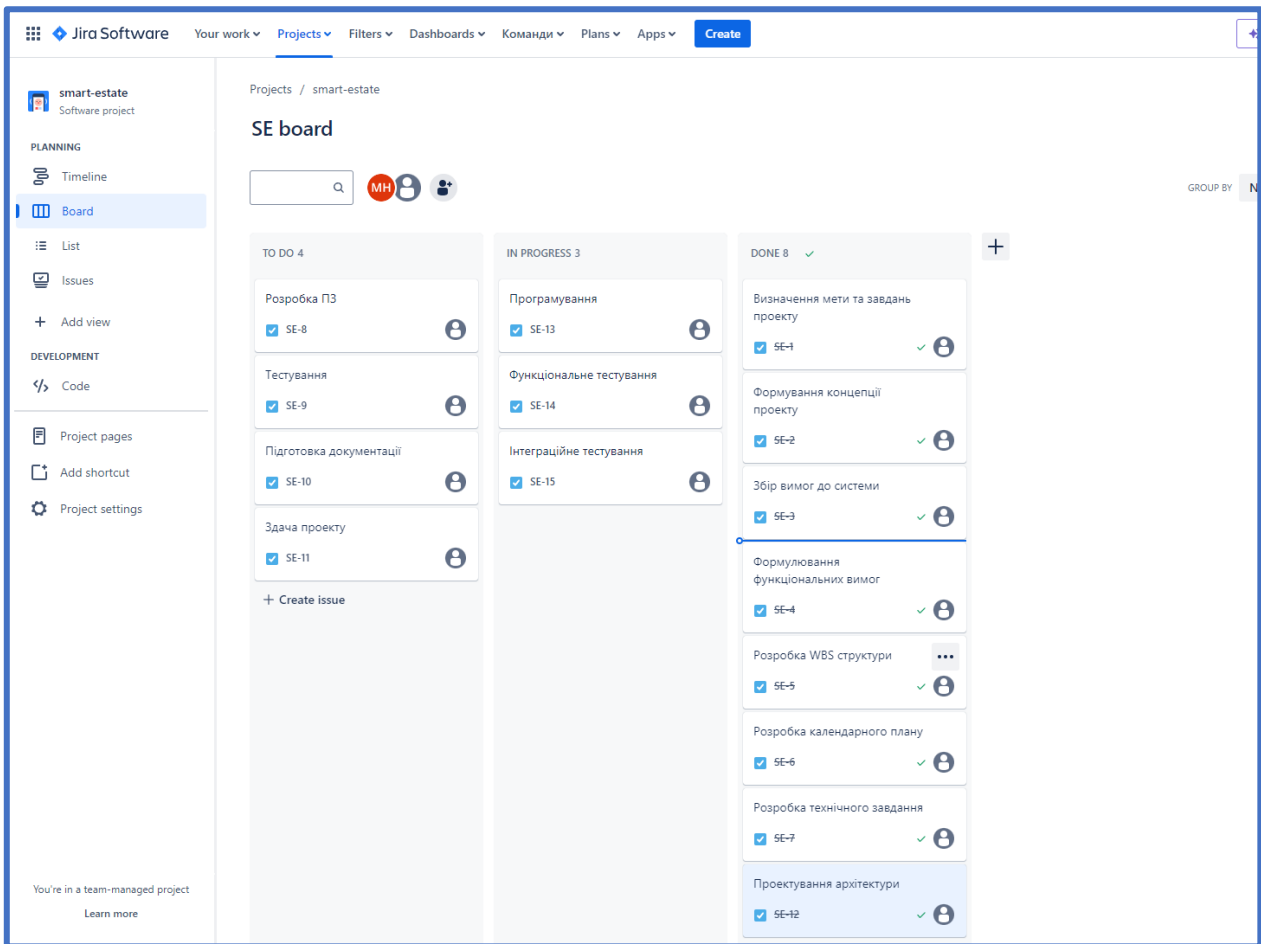


Рис. 2.28 – Канбан дошка

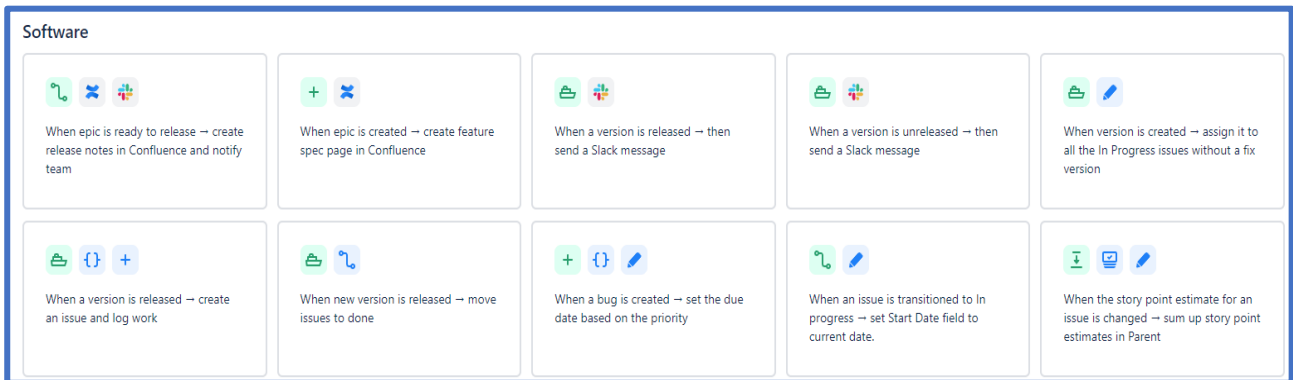


Рис. 2.29 – Налаштування автоматичних процесів

Тепер переходьте до дій, які потрібно виконати для зміни статусу задачі. Натисніть «Add action» та виберіть «Transition issue» зі списку доступних дій. Далі налаштуйте дію, вибравши статус, на який потрібно перевести задачу, наприклад, «In Progress» або «Done». Якщо потрібно, ви можете встановити умови, за яких дія буде виконуватись. Після цього натисніть «Save», щоб

зберегти налаштування дії.

Перегляньте створене правило, щоб упевнитися, що всі налаштування виконані правильно. У верхньому полі введіть назву для вашого правила, наприклад, «*Auto-Transition Tasks*». Коли все готово, натисніть «*Turn it on*» або «*Save*», щоб активувати правило автоматизації.

Щоб перевірити, чи працює ваше правило, спробуйте створити або оновити задачу та спостерігайте, чи тригер та пов'язані дії спрацьовують коректно. У розділі автоматизації також доступний «*Audit log*», де ви можете переглянути виконання правил і виявити можливі проблеми.

Якщо згодом вам потрібно додати зміни до правила, поверніться до налаштувань автоматизації, виберіть правило з списку і натисніть на нього для редагування. Додайте необхідні зміни та знову натисніть «*Save*».

Цей покроковий алгоритм допоможе вам ефективно налаштувати автоматичне оновлення статусів задач у *Jira*. Це значно спростить робочі процеси та підвищить продуктивність вашої команди. **Наприклад**, ви можете створити правило, яке автоматично переміщує задачі в статус «*In Progress*», коли один з членів команди коментує задачу. Таким чином, вся команда буде в курсі актуального стану завдань, а робота над проектом буде більш структурованою.

Розглянемо декілька **прикладів** використання автоматичного оновлення статусів:

1. Автоматичне переведення задач у статус «*In Progress*» при їх створенні. Команда може налаштувати правило так, щоб кожного разу, коли нова задача створюється, вона автоматично переходила в статус «*In Progress*» (рис. 2.30). Це зручно, коли ви хочете, щоб усі нові задачі відразу починали виконуватися.
2. Перехід задач у статус «*Done*» після підтвердження. Після того як задача була завершена та перевірена, команда може автоматично перевести її в статус «*Done*». Це особливо корисно в командах, де завершення задачі потребує підтвердження від декількох членів команди (рис. 2.31).

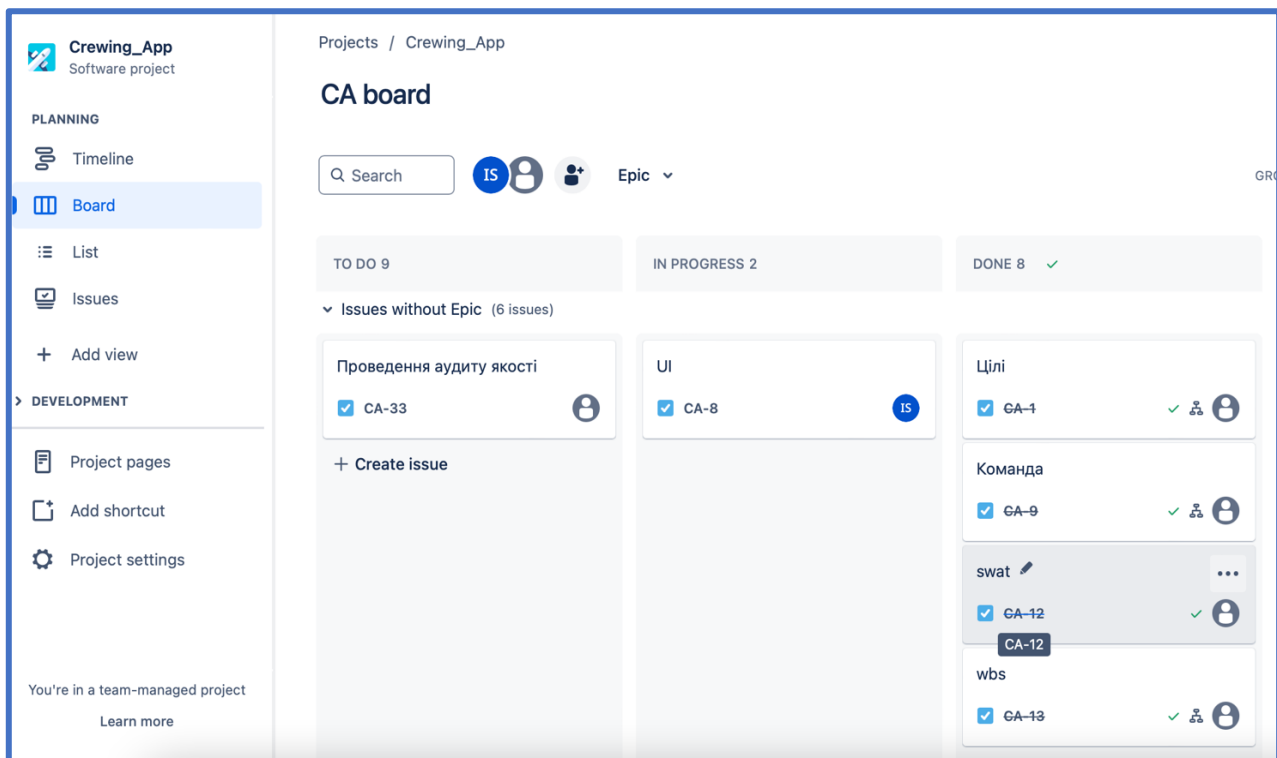


Рис. 2.30 – Статус «In Progress»

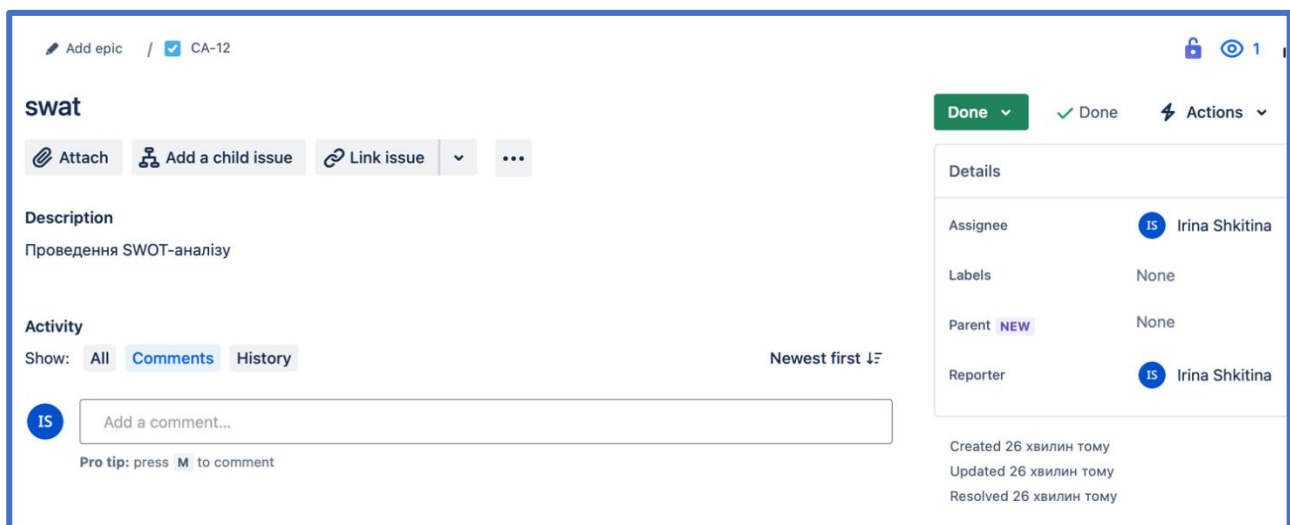


Рис. 2.31 – Статус «Done»

3. Автоматичне оновлення статусу в залежності від зміни значення поля. Якщо у вашій команді існує специфічне поле, наприклад, «*Priority*», ви можете налаштувати правило, щоб автоматично змінювати статус задачі на «*Critical*» (Критичний) або «*Urgent*» (Терміновий) у відповідь на зміну значення цього поля (рис. 2.32).

4. Синхронізація статусів між задачами. У випадку, якщо задача залежить від іншої, ви можете налаштувати автоматичне оновлення статусу залежної задачі, коли основна задача переходить у статус «Done». Це допоможе вашій команді зберігати актуальність статусів у всіх пов'язаних задачах.

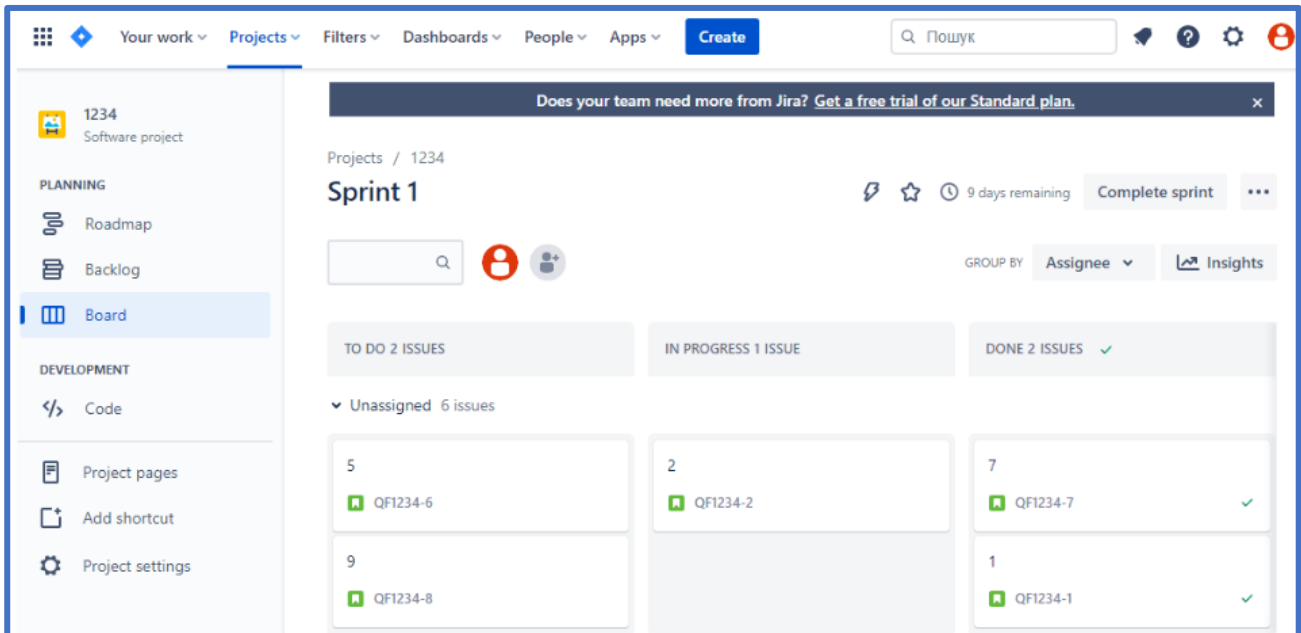


Рис. 2.32 – Гнучке налаштування Workflow

Завдяки автоматизації оновлення статусів, ваша команда зможе швидше реагувати на зміни, підвищуючи ефективність і спрощуючи робочі процеси.

2.5. Звіти та аналітика в Jira

Jira – це потужний інструмент управління проектами, який забезпечує командам гнучкість у плануванні, виконанні та моніторингу завдань. Однак його можливості не обмежуються лише управлінням завданнями. Однією з ключових функцій **Jira** є здатність генерувати звіти та надавати аналітику, що дозволяє командам отримувати цінну та корисну інформацію для покращення процесів і прийняття рішень (рис. 2.33).

Коли ви створюєте проєкт, то за замовчуванням пропонується дуже спрощений Workflow, в якому завдання можна переміщати зі статусу в статус

абсолютно довільно. Для цього достатньо перетягнути вибране завдання у потрібну колонку.

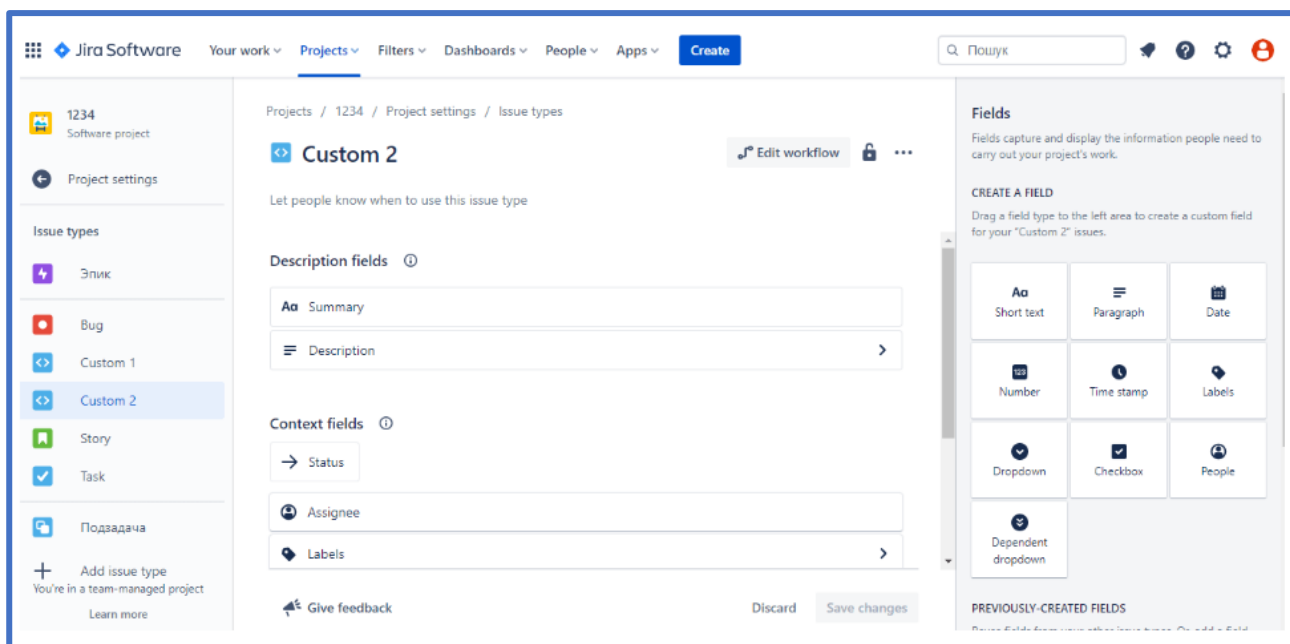


Рис. 2.33 – Гнучке налаштування Workflow

Все просто й інтуїтивно зрозуміло, але коли багато завдань і процеси стають складнішими, подібна свобода дій може призвести до хаосу. Щоб такого не було, у **Jira** передбачена можливість налаштування робочих процесів за певними правилами. Для цього потрібно зайти в проєкт у будь-який тип завдань та в інтерфейсі редагування натиснути кнопку «Змінити робочий процес».

У **Jira** передбачена можливість налаштування робочих процесів за певними правилами. Для цього потрібно зайти в проєкт у будь-який тип завдань та в інтерфейсі редагування натиснути кнопку «Змінити робочий процес» (рис. 2.34).

Перед вами відкриється вікно зі схемою Workflow за промовчанням.

Зображена логіка бізнес-процесів для підготовки подальших звітів дозволяє змінювати статус завдань у будь-якому порядку. За допомогою простих інструментів – «Статус», «Перехід» і «Правило» – ви можете створити новий ланцюжок переходів завдань з одного стану в інший і навіть додати нову колонку-статус на дошку (рис. 2.35).

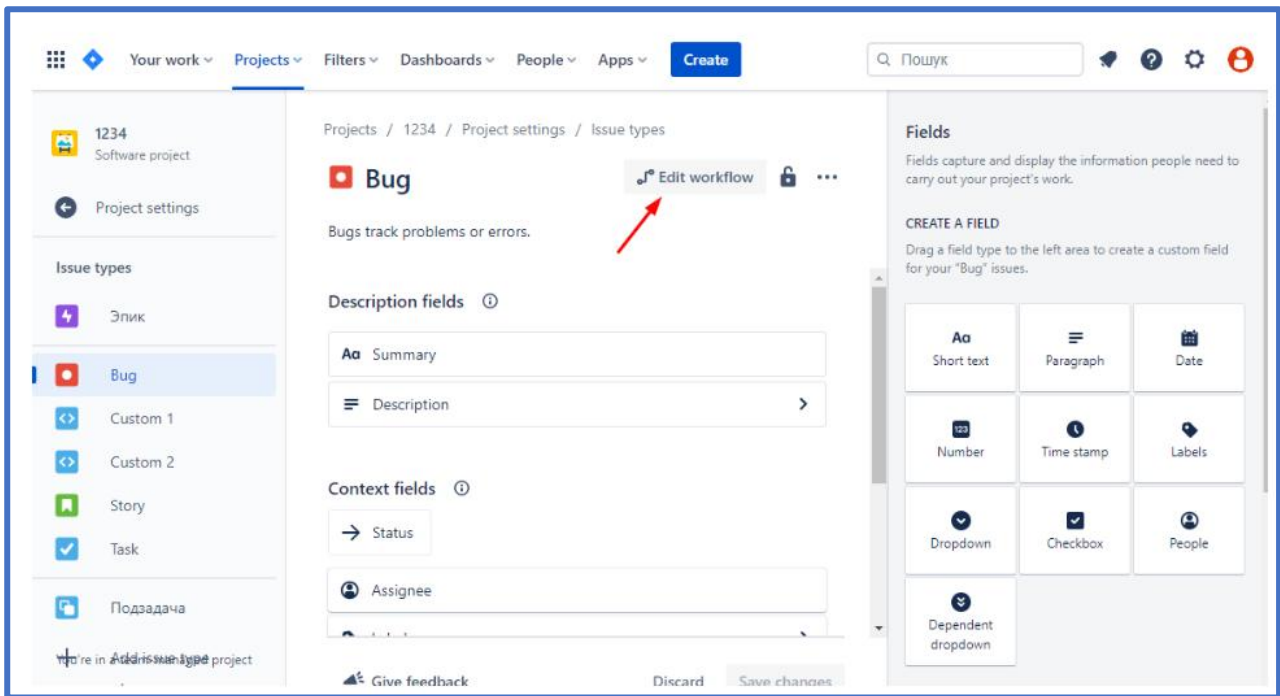


Рис. 2.34 – Налаштування робочих процесів

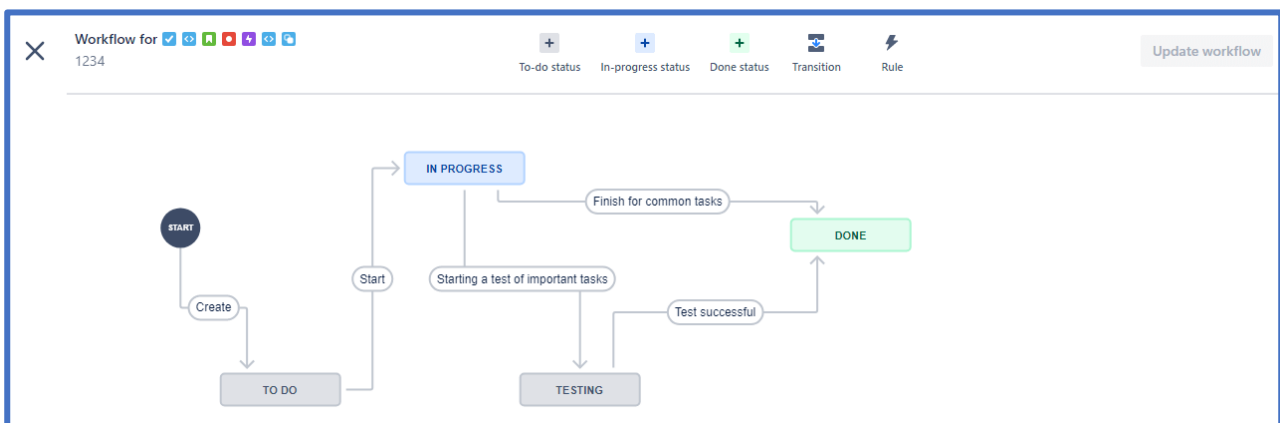


Рис. 2.35 – Логіка бізнес-процесів у звітах

Типи звітів

Jira пропонує різноманітні типи звітів, які можуть бути використані для відстеження прогресу проєкту, ефективності команди та виявлення потенційних проблем (2.36).

Найпопулярніші типи звітів включають:

Звіт про статуси задач. Він надає загальний огляд статусів усіх задач у проєкті. Він допомагає зрозуміти, скільки задач знаходяться в кожному статусі (нові, в процесі, завершені) та визначити вузькі місця в процесі.

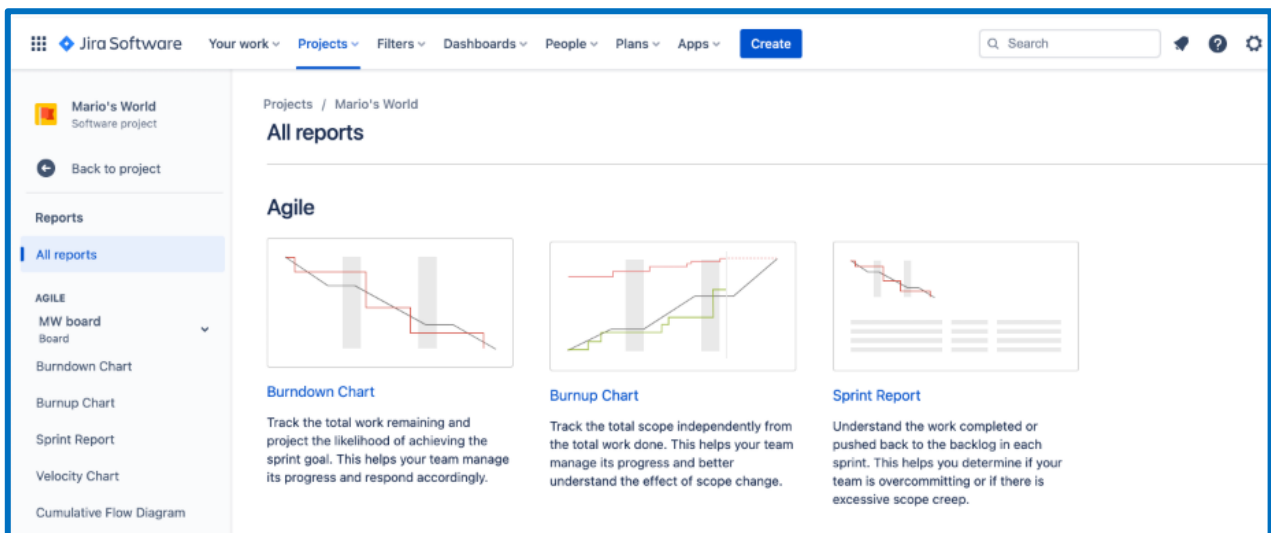


Рис. 2.36 – Вибір необхідного звіту

Звіт про швидкість (Velocity Report). Даний звіт корисний для *Scrum*-команд, оскільки відображає, скільки роботи (у вигляді *Story Points* або інших метрик) команда виконала за спринт. Він дозволяє командам планувати майбутні спринти на основі попередніх результатів.

Звіт про залишковий час (Burn Down Chart). Звіт демонструє, скільки часу залишилося до завершення проєкту або спринту. Цей звіт корисний для відстеження прогресу та виявлення відхилень від запланованого графіка.

Звіт про ефективність команди (Team Performance Report). Звіт надає можливість аналізувати продуктивність окремих членів команди, що допомагає виявити сильні та слабкі сторони.

Звіт про проблеми (Issue Statistics). Звіт надає можливість групувати задачі за різними критеріями, такими як пріоритет, тип або відповідальна особа. Це допомагає зрозуміти, які області потребують додаткової уваги.

Аналітика та інтеграції

Крім вбудованих звітів, **Jira** також підтримує інтеграції з багатьма аналітичними інструментами, такими як *Google Analytics*, *Tableau* або *Power BI*. Це дозволяє командам створювати детальні дашборди та проводити глибокий аналіз даних.

Інтеграція з *BI-інструментами* дає можливість не лише візуалізувати дані, а й застосовувати більш складні аналітичні методи для виявлення трендів, кореляцій та прогнозів. Наприклад, компанії можуть аналізувати, які типи задач найчастіше затримуються, або в якій фазі проєкту виникають найбільші затримки.

Користувацькі дашборди

Jira також дозволяє створювати користувацькі дашборди, які можуть містити різноманітні віджети, такі як графіки, списки задач і статистику. Це надає командам можливість здійснювати моніторинг ключових показників продуктивності (*KPI*) в режимі реального часу, що сприяє швидкому прийняттю рішень (рис. 2.37).

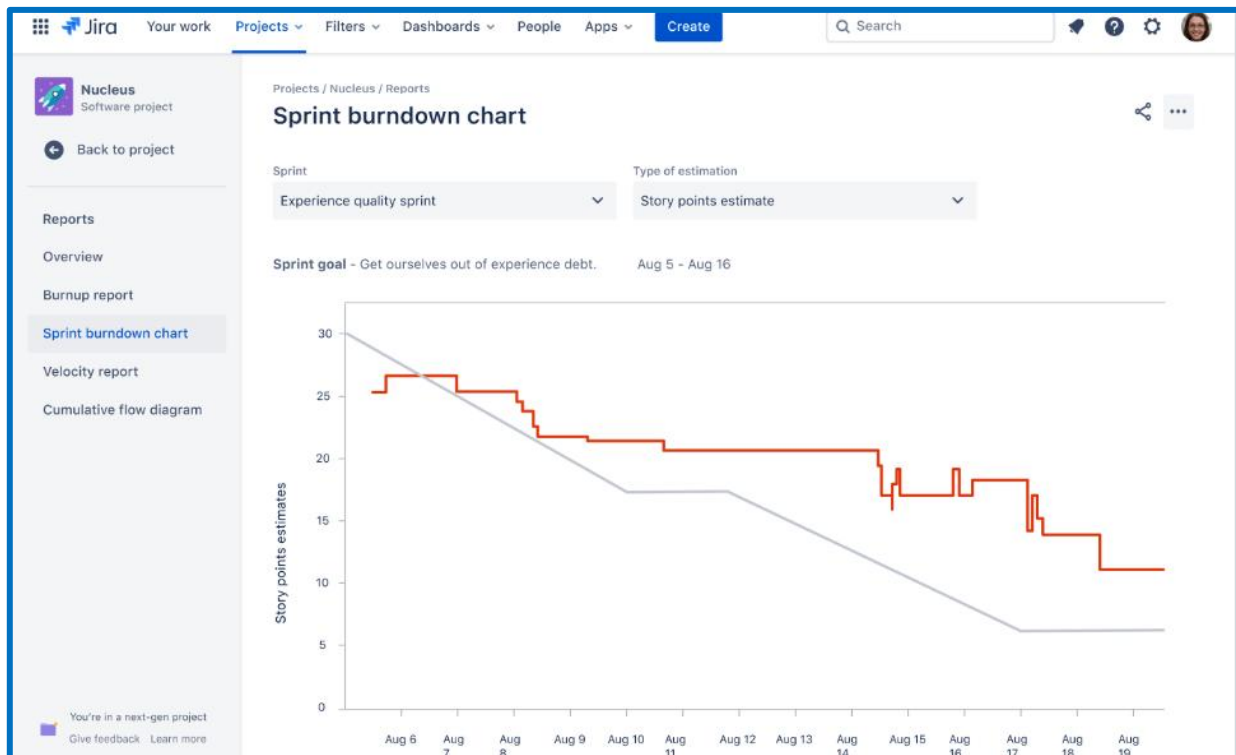


Рис. 2.37 – Логіка бізнес-процесів у звітах

Користувацькі дашборди можуть бути налаштовані відповідно до потреб команди або окремих учасників, що забезпечує гнучкість у відображенні необхідної інформації. **Наприклад**, менеджер проєкту може створити дашборд, що показує статус всіх задач, у той час як розробник може сконцентруватися на задачах, призначених лише для нього (рис.2.38).

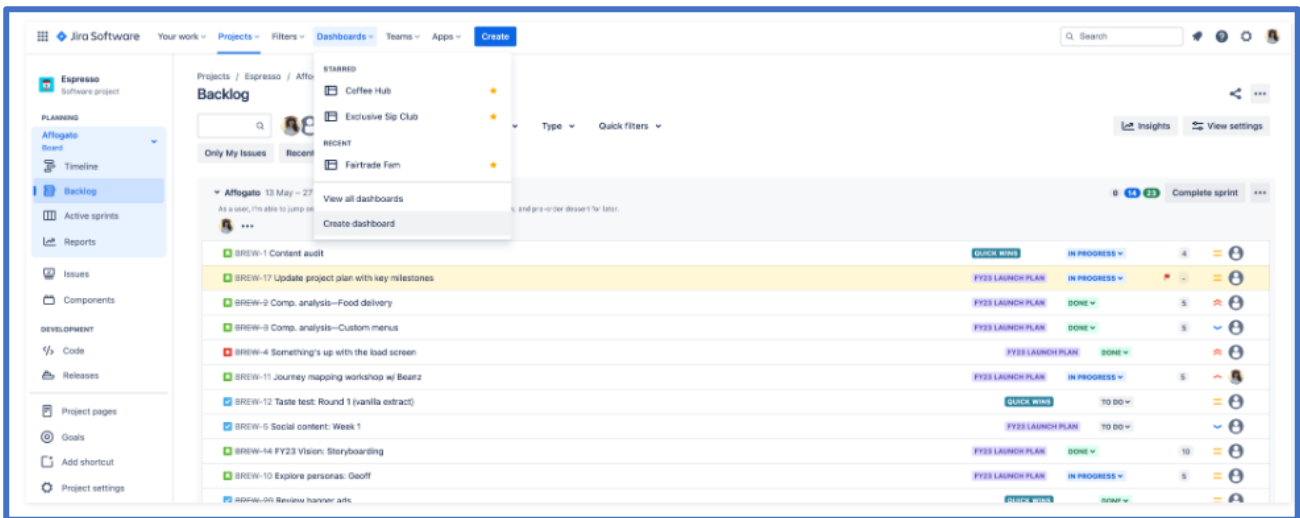


Рис. 2.38 – Користувацькі дашборди

Звіти та аналітика в **Jira** є критично важливими для успішного управління проектами. Вони надають командам важливу інформацію, що дозволяє не лише відстежувати прогрес, але й виявляти проблеми, покращувати процеси і приймати обґрунтовані рішення. Завдяки можливостям інтеграції та створення користувацьких дашбордів, **Jira** стає потужним інструментом для аналітики, який підтримує команди у досягненні їхніх цілей.

2.5.1. Створення звітів для Scrum-команд

Створення звітів у **Jira** є важливим аспектом управління проектами, особливо для Scrum-команд, які прагнуть аналізувати свою продуктивність і відстежувати прогрес. У цій інструкції наведено покроковий алгоритм, як створити різноманітні звіти в **Jira**, щоб отримати важливу інформацію про роботу вашої команди (рис. 2.39).

На початку, для входу до **Jira**, вам потрібно ввести свої облікові дані для доступу до системи. Після цього у верхньому меню знайдіть вкладку «*Projects*» та виберіть проєкт, для якого плануєте створити звіт. Це дозволить вам зосередитися на специфіці саме цього проєкту.

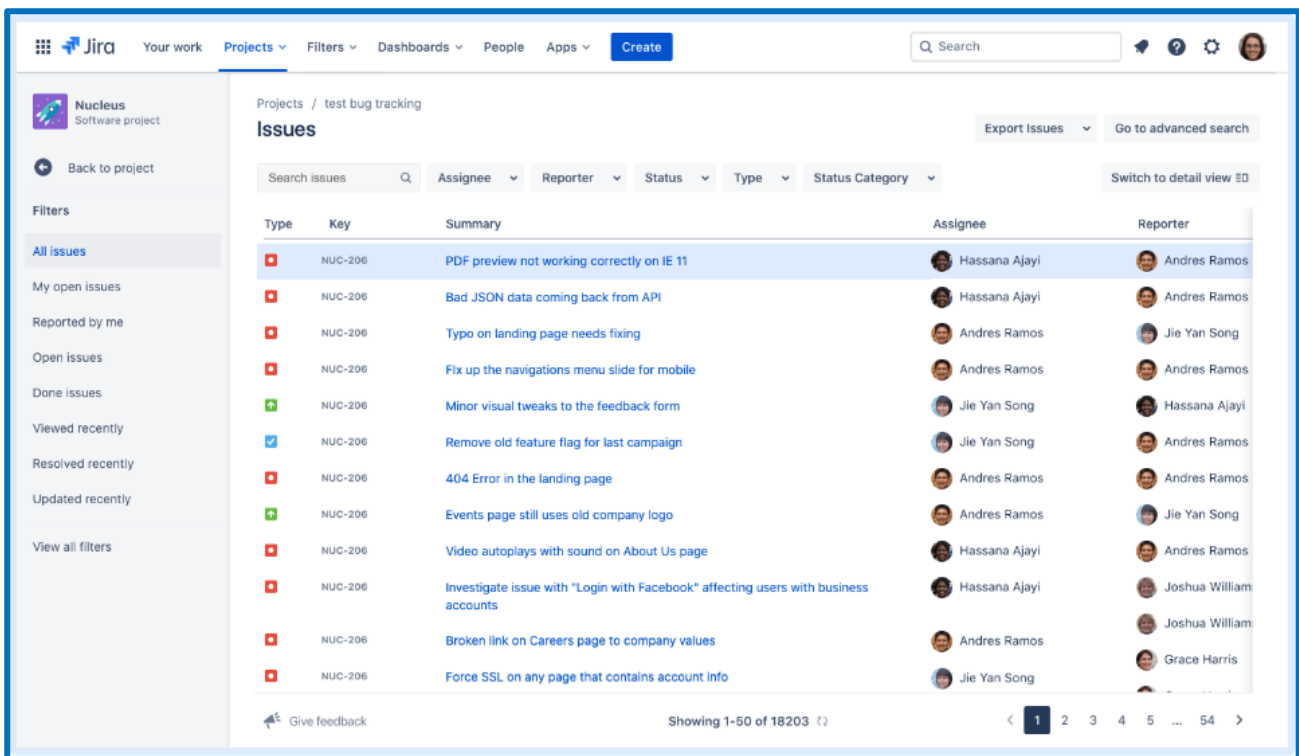


Рис. 2.39 – Інформація про роботу команди

Наступним кроком є перехід до звітів. У лівій панелі навігації знайдіть і натисніть на «*Reports*». Тут ви зможете обрати тип звіту, що найбільше відповідає вашим потребам. Для Scrum-команд особливо корисними є такі звіти, як «*Sprint Report*», «*Velocity Chart*» і «*Burn Down Chart*». Натисніть на потрібний звіт, наприклад, «*Velocity Chart*», щоб перейти до налаштувань (рис. 2.40).

У вікні налаштувань звіту ви зможете вибрати спринт, для якого хочете згенерувати звіт. Знайдіть список спринтів і виберіть потрібний. Залежно від типу звіту, можуть бути доступні додаткові параметри для налаштування.

Наприклад, для «*Burn Down Chart*» ви можете вказати, які задачі слід включити до звіту. Після внесення необхідних налаштувань натисніть на «*View report*» або «*Generate report*» (Згенерувати звіт), в залежності від версії **Jira**.

Після створення звіту ознайомтеся з його змістом.

Наприклад, *Velocity Chart* покаже, скільки роботи команда виконала за спринт, а *Burn Down Chart* відобразить прогрес у виконанні задач. Це важлива інформація, яка допоможе вам зрозуміти, наскільки ефективно команда справляється з виконанням завдань.

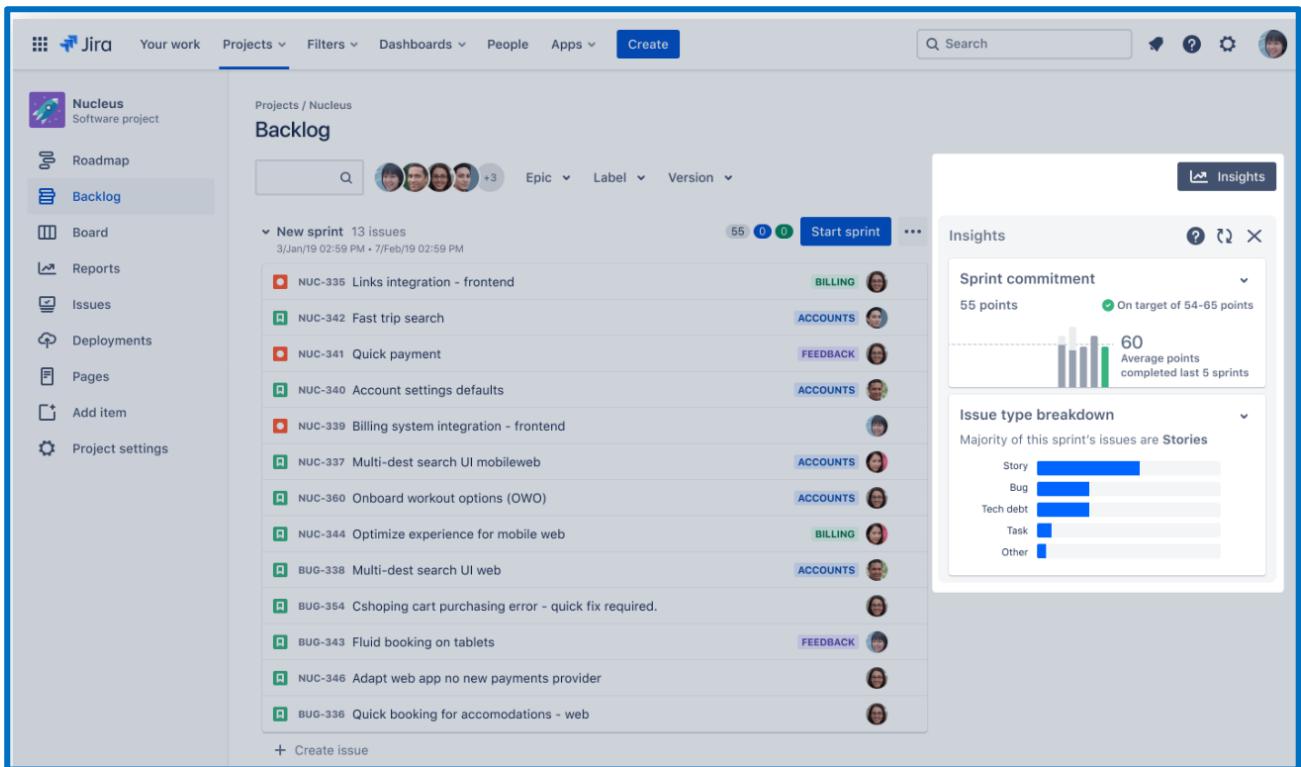


Рис. 2.40 – Налаштування користувацьких дашбордів

У деяких версіях **Jira** доступна можливість зберігати звіти для подальшого використання. Якщо ви бажаєте зберегти звіт, знайдіть опцію «*Save report*» або «*Bookmark this report*» (Додати цей звіт до закладок). Також, якщо потрібно, ви можете експортувати звіт у зручному для вас форматі, наприклад, PDF чи Excel. Для цього знайдіть кнопку «*Export*», яка зазвичай розташована у верхньому правому куті.

Щоб повернутися до списку звітів, натисніть «*Back to Reports*» або скористайтеся навігаційними кнопками. Якщо ви бажаєте створити інший звіт, наприклад, «*Sprint Report*» або «*Burn Down Chart*», повторіть етапи з вибору типу звіту та налаштування його під ваші потреби.

Цей покроковий алгоритм дозволяє *Scrum*-командам створювати різноманітні звіти в **Jira**, що надає важливу інформацію про їхню продуктивність, прогрес і області для покращення. Звіти є незамінним інструментом для аналізу роботи команди та прийняття обґрунтованих рішень у процесі управління проектами.

Розглянемо декілька **прикладів** використання звітів:

1. *Velocity Chart*. Команда може використовувати цей звіт для аналізу своєї продуктивності в кількісному вимірі. Наприклад, якщо команда помічає, що швидкість зменшується за кілька спринтів, це може вказувати на проблеми в плануванні або надмірне навантаження.
2. *Burn Down Chart*. Цей звіт дозволяє команді візуально відстежувати прогрес виконання задач під час спринту. Якщо графік не знижується так, як планувалося, команда може оперативного переглянути обсяг задач, які залишилися, і вжити заходів для їх виконання.
3. *Sprint Report* Використовується для підбиття підсумків спринту. Команда може проаналізувати, які задачі були завершені, які – ні, і які причини затримки. Цей звіт допомагає в ідентифікації областей, де можна поліпшити процеси.

Таким чином, створення звітів у *Jira* не тільки спрощує управління проектами, але й допомагає команді зосередитися на постійному вдосконаленні своїх робочих процесів.

2.5.2. Використання метрик та дашбордів

Використання метрик продуктивності команди в *Jira* дозволяє ефективно оцінювати результати роботи, виявляти проблеми та визначати сфери для вдосконалення. Правильна реалізація цього процесу допомагає команді покращити свої робочі процеси та досягти поставлених цілей.

Для початку вам слід увійти в систему, використовуючи свої облікові дані. Після цього, у верхньому меню натисніть на «*Projects*» та оберіть той проєкт, для якого ви хочете переглянути метрики продуктивності. Коли ви перейдете до потрібного проєкту, зверніть увагу на ліву панель навігації, де знайдете розділ «*Reports*». Натиснувши на цю опцію, ви відкриєте доступ до різноманітних звітів, які надають цінну інформацію про виконання задач.

Серед доступних звітів ви знайдете такі метрики продуктивності, як «*Velocity Chart*», що показує, скільки роботи було виконано за спринт; «*Burn*

Down Chart», яка демонструє, як залишок роботи зменшується протягом спринту та «*Sprint Report*», що надає інформацію про виконані та невиконані задачі; а також «*Control Chart*», що відображає час, витрачений на виконання задач.

Вибравши потрібний звіт, наприклад, *Velocity Chart*, ви зможете перейти до його налаштування. У вікні налаштувань оберіть спринт, для якого ви бажаєте згенерувати метрики продуктивності. Переконайтеся, що ви правильно налаштували параметри, адже, в залежності від вибраного звіту, можуть бути доступні додаткові налаштування.

Наприклад, у *Control Chart* ви можете вказати, за якими задачами відображати дані – всі задачі, лише завершені або активні.

Після завершення налаштувань натисніть на «*View report*» або «*Generate report*», залежно від версії *Jira*. Тепер ви зможете переглянути звіт, щоб отримати детальну інформацію про продуктивність команди.

Наприклад, *Velocity Chart* покаже, скільки *Story Points* було виконано за спринт, в той час як *Burn Down Chart* відобразить залишок роботи на день, що дозволить вам вчасно реагувати на можливі затримки.

Аналізуючи дані, зверніть увагу на закономірності у виконанні задач. Якщо ви помічаєте, що продуктивність команди знизилася, спробуйте виявити причини, які можуть бути пов'язані з технічними проблемами, недолік ресурсів або інші фактори. Цей аналіз допоможе вам зрозуміти, в яких областях потрібно працювати для покращення продуктивності.

Важливо також обговорити результати з командою, адже їхня думка може виявитися цінною для виявлення можливих шляхів покращення. Під час обговорення команди можуть пропонувати нові ідеї або вказувати на проблеми, які не були помічені раніше.

Деякі версії *Jira* дозволяють зберігати звіти для подальшого використання. Якщо це ваш випадок, знайдіть опцію «*Save report*» або «*Bookmark this report*». Якщо потрібно, ви також можете експортувати звіт, натиснувши на кнопку «*Export*», зазвичай розташовану у верхньому правому куті, і вибравши потрібний формат, наприклад, PDF або Excel.

Після цього ви можете повернутися до списку звітів, натиснувши «*Back to Reports*» або використовуючи навігаційні кнопки. Якщо вам потрібно створити інші звіти або переглянути інші метрики продуктивності, повторіть етапи з вибору звіту та налаштування.

Регулярно переглядаючи метрики продуктивності, ви зможете вчасно виявляти проблеми та адаптувати процеси команди. Заплануйте регулярні огляди, щоб зберегти високий рівень продуктивності.

Розглянемо декілька **прикладів** використання метрик продуктивності в **Jira**:

1. Оцінка спринту. Після завершення спринту команда вивчає *Velocity Chart* і виявляє, що виконали на 10% більше задач, ніж запланували. Це надихає команду планувати більш амбітні цілі в наступних спринтах.
2. Виявлення проблем. Під час перегляду *Burn Down Chart* команда помічає, що графік знижується повільніше, ніж очікувалося, і вирішує провести додаткові обговорення для аналізу причин, що призводять до затримок у виконанні задач (рис. 2.41).
3. Аналіз результатів. У *Sprint Report* команда бачить, що певні задачі часто залишаються незавершеними. Це спонукає їх проаналізувати, чому ці задачі виявляються складними, і розробити нові підходи до їх виконання.
4. Планування майбутніх спринтів. Використовуючи дані з *Velocity Report*, команда розуміє, що їм потрібно підвищити свою продуктивність. Вони вирішують переглянути обсяг задач у майбутніх спринтах, щоб зробити їх більш реалістичними та досяжними.

Завдяки ефективному використанню метрик продуктивності в **Jira**, команди можуть не лише підвищувати свою продуктивність, але й досягати кращих результатів у реалізації проєктів.

Візуалізація даних через дашборди

Візуалізація даних через дашборди в **Jira** є потужним інструментом, що дозволяє командам ефективно відстежувати прогрес, аналізувати продуктивність та швидко отримувати ключову інформацію в зручному форматі.

Дашборди забезпечують персоналізацію вигляду інформації, інтегруючи різноманітні віджети, які відповідають специфічним потребам команди. Це особливо важливо у контексті управління проєктами, де ефективна візуалізація даних може суттєво вплинути на прийняття рішень та координацію дій.

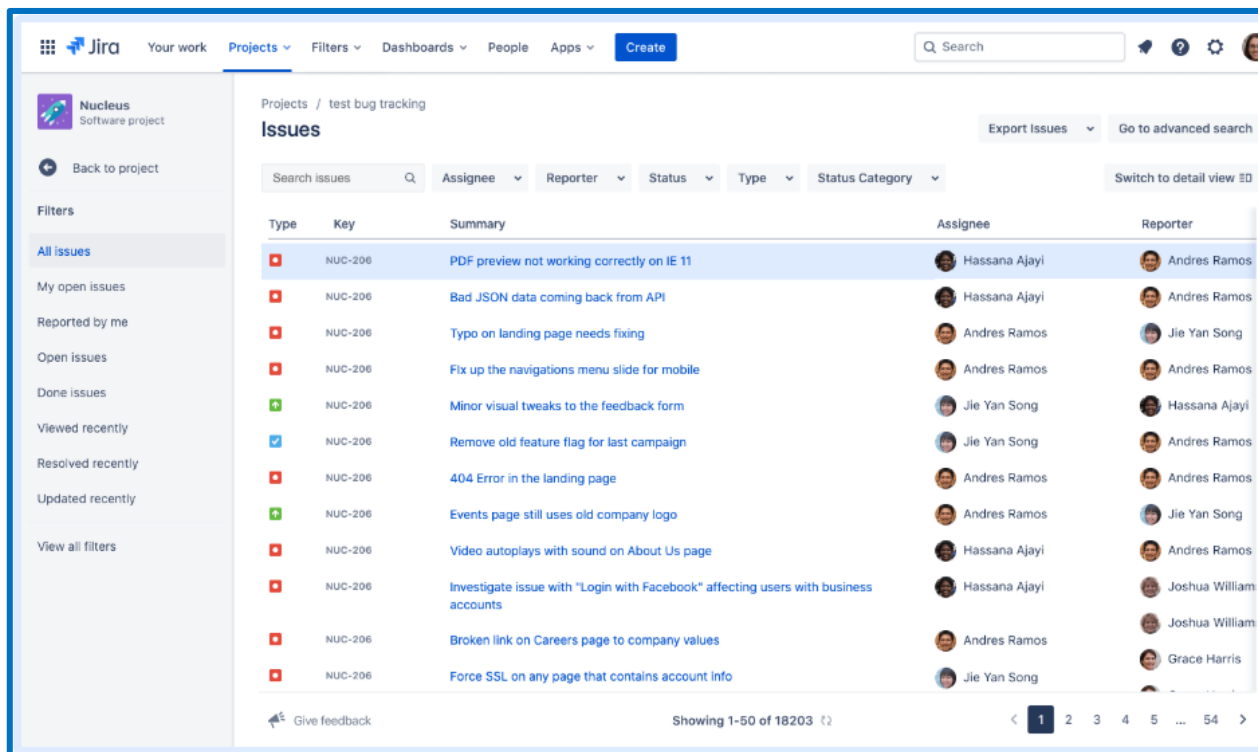


Рис. 2.41 – Шаблон звіту про проблему

Перш ніж почати, увійдіть до системи, використовуючи свої облікові дані. Після успішного входу у верхньому меню знайдіть вкладку «Dashboards» (Дашборди). Це відкриє доступ до вже наявних дашбордів або дозволить створювати нові. Для створення нового дашборда натисніть на «Create dashboard». У вікні, що з'явиться, ви зможете вказати назву дашборда та обрати його конфіденційність – чи буде він доступний лише для вас, чи для всієї команди (рис. 2.42).

Після створення дашборда ви потрапите на його редагування, де зможете додавати різноманітні віджети. Віджети – це маленькі модулі, які відображають специфічну інформацію в графічному вигляді. Серед найпопулярніших віджетів можна знайти «Filter Results», «Sprint Burndown», «Pie Chart» (Коловий графік) та «Two-Dimensional Filter Statistics» (Двовимірна статистика фільтра).

Кожен з цих віджетів має свої особливості, що дозволяє командам адаптувати дашборди під свої потреби. Наприклад, віджет «*Filter Results*» дозволяє переглядати конкретні задачі на основі заданих критеріїв. Ви можете налаштувати фільтри, щоб відобразити тільки ті задачі, які вас цікавлять, наприклад, всі невиконані задачі за останній спринт. Це дає змогу команді зосередитися на пріоритетних завданнях і швидко виявляти проблеми.

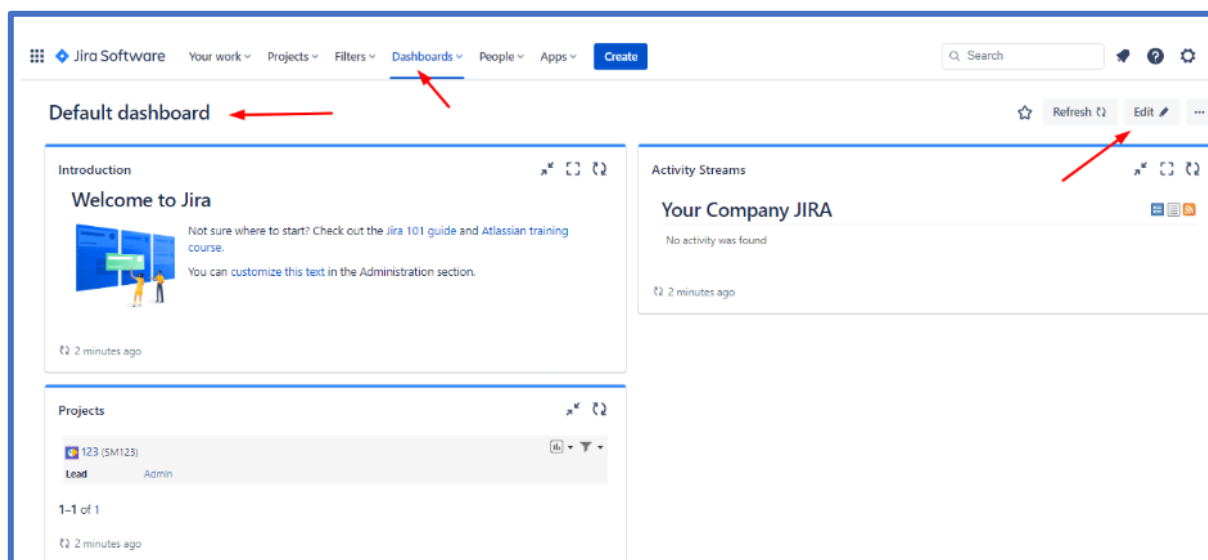


Рис. 2.42 – Створення та налаштування дашбордів

Щоб додати віджет, натисніть на «*Add Widget*» (*Додати віджет*). Після цього вам буде представлений вибір доступних віджетів. Ви можете вибрати віджети, які найкраще підходять для вашої команди та проекту, і налаштувати їх відповідно до специфічних вимог. Наприклад, ви можете додати «*Burndown Chart*» для відстеження залишку роботи по спринту. У налаштуваннях віджету виберіть спринт, який ви хочете аналізувати, і збережіть зміни (рис. 2.43).

Коли ваш дашборд буде налаштовано, ви зможете легко переглядати важливу інформацію у візуальному форматі, що зручно для сприйняття. В залежності від потреб команди, ви можете додавати графіки та таблиці, що відображають ключові показники, такі як швидкість виконання задач, залишок роботи, статуси задач та багато іншого. Це допоможе всім членам команди бути в курсі справ і зрозуміти, на яких етапах реалізації проекту потрібно зосередити увагу (рис. 2.44).

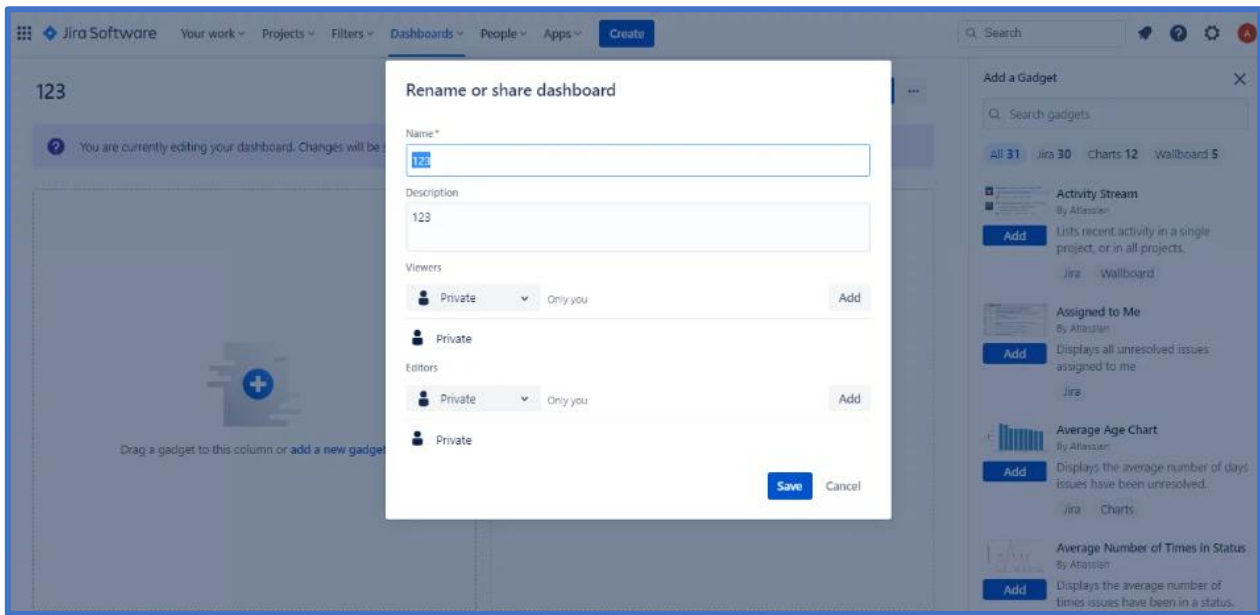


Рис. 2.43 – Створення нового дашборду

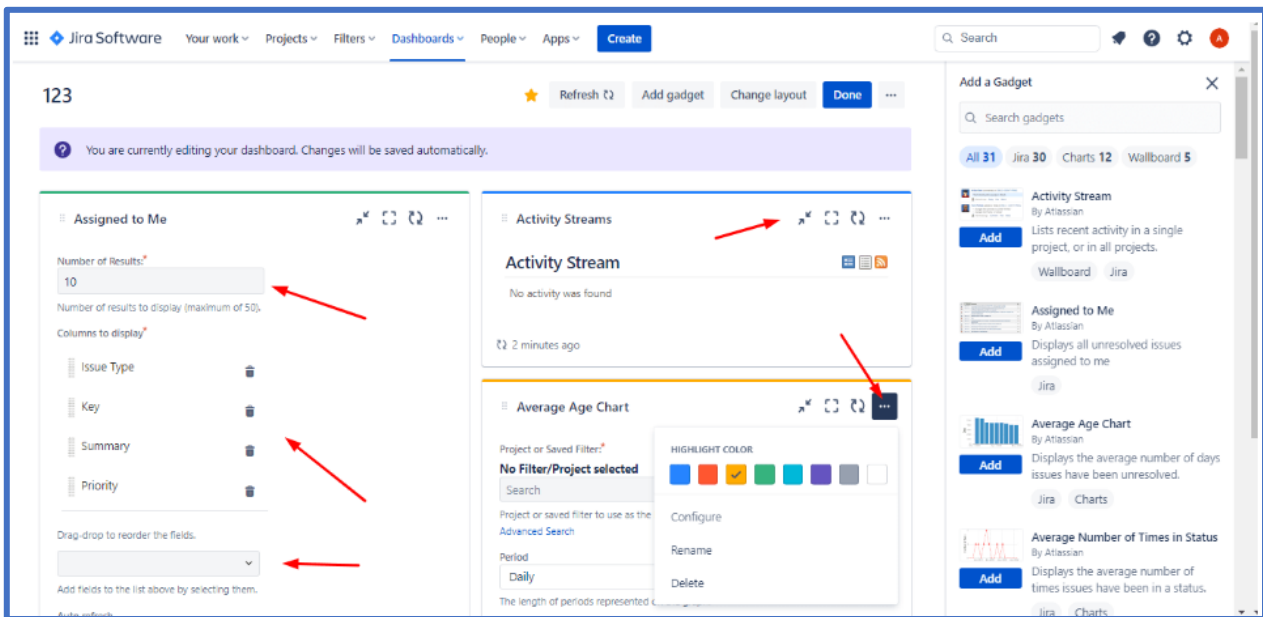


Рис. 2.44 – Налаштування нового дашборду

Аналіз даних дашбордів

Аналізуючи дані, представлені на дашборді, ви зможете швидко виявити проблеми та можливості для вдосконалення. Наприклад, якщо ви помічаєте, що задачі, пов'язані з певним компонентом проекту, часто затримуються, це може свідчити про необхідність додаткової підтримки або ресурсів. Завдяки дашбордам ви можете також виявити закономірності у виконанні задач, що дає

можливість проводити регулярні обговорення з командою.

Важливо також регулярно оновлювати дашборди, додаючи нові віджети та налаштовуючи їх відповідно до змін у проєкті. Команда може обговорювати та аналізувати дані, отримані з дашбордів, що сприяє покращенню процесів і збільшенню продуктивності. Зокрема, щотижневі або щомісячні огляди дашбордів можуть стати чудовою практикою для виявлення потенційних затримок і коригування планів на наступні етапи (рис. 2.45).

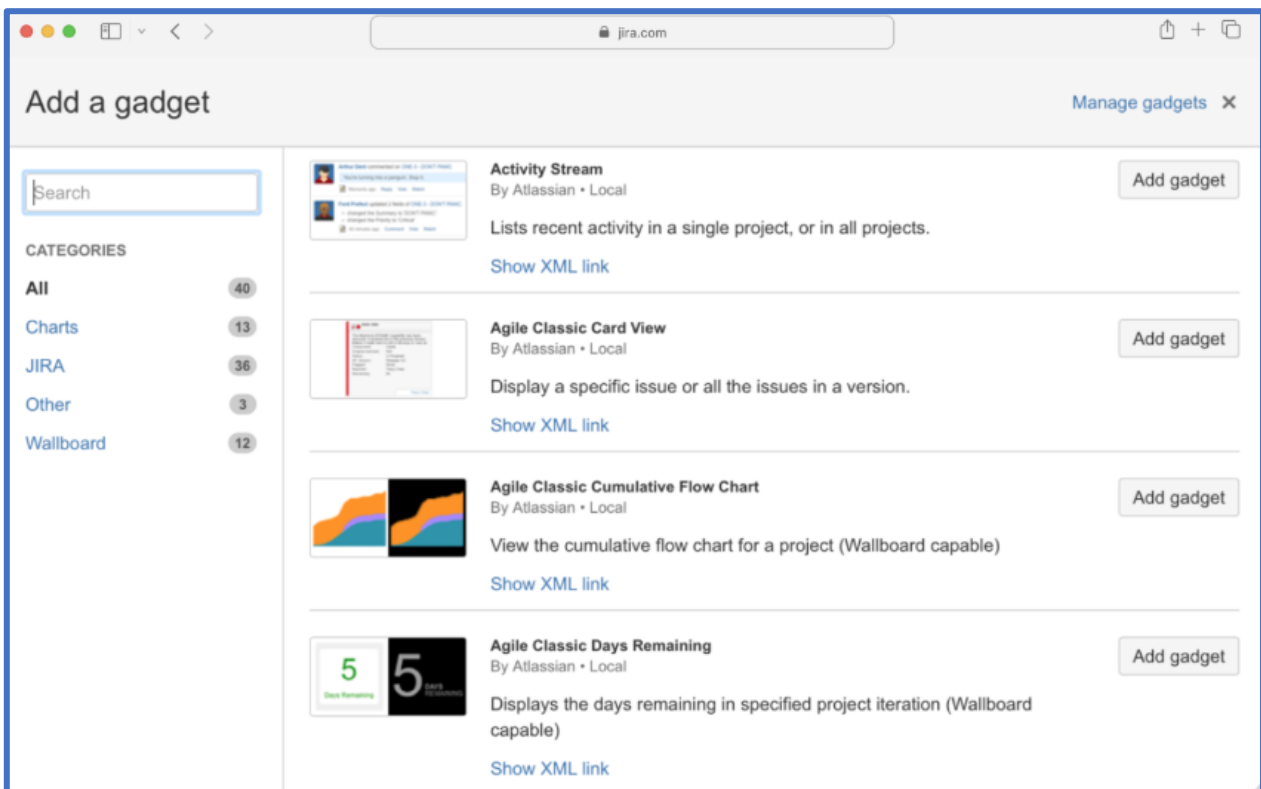


Рис. 2.45 – Додавання гаджетів та налаштування

Приклади використання дашбордів у **Jira** наведені нижче:

1. Моніторинг спринтів. Дашборд зі «*Sprint Burndown*» дозволяє команді відстежувати, як швидко вони виконують задачі в рамках спринту, а також вчасно виявляти можливі затримки. Команда може побачити, скільки роботи залишилося на кожен день спринту, і коригувати свої зусилля відповідно до цього (рис. 2.46).
2. Аналіз ефективності. Використання «*Pie Chart*» для відображення статусів задач допомагає команді зрозуміти, яка частина роботи завершена, а яка ще

перебуває в процесі. Це дозволяє вчасно виявити блокери і спланувати роботу в найближчій перспективі.

3. Статистика виконання. Дашборд, що містить «*Two-Dimensional Filter Statistics*», дозволяє команді побачити, скільки задач було виконано по кожному члену команди та які задачі ще потрібно виконати. Це допомагає розподілити навантаження та визначити, хто потребує підтримки.
4. Візуалізація прогресу. За допомогою «*Filter Results*» команда може створити дашборд, що показує всі задачі, які не були завершені за минулий спринт, що дозволяє зосередити зусилля на їх виконанні. Це може стати основою для обговорення на щоденних або тижневих зборах.
5. Визначення тенденцій: За допомогою дашборда, що містить «*Control Chart*», команда може аналізувати, як довго виконуються задачі протягом часу. Це дозволяє виявити тенденції в продуктивності команди і робити висновки для покращення процесів.

Rename or share dashboard

Required fields are marked with an asterisk *

Name *

Travel Team Dashboard

Description

Viewers

Group jira-service... Add

jira-servicemanagement-customers X

Editors

Group Group Add

Marcus Jones X

Travel Team Request Queue, Administrators X

Save Cancel

Рис. 2.46 – Спільний доступ до дашборду

Візуалізація даних через дашборди в *Jira* дозволяє командам краще контролювати свій прогрес, адаптувати свої стратегії та досягати значно кращих результатів у реалізації проєктів. Інтеграція візуалізації даних в повсякденну практику команди забезпечує прозорість процесів, покращує взаємодію між членами команди та дозволяє швидше реагувати на зміни. Завдяки дашбордам команди можуть отримати не тільки огляд поточної ситуації, а й глибокий аналіз, що сприяє постійному вдосконаленню.

Завдання для самостійного опрацювання

Завдання 1. Управління Scrum-проектом у Jira

Хід виконання:

Етап 1. Створення Scrum-проекту.

Крок 1. Увійдіть у Jira і створіть Scrum-проект під назвою (наприклад) «Розробка CRM системи».

Крок 2. Налаштуйте шаблон проекту на основі Scrum-методології (наприклад):

- Створіть команди з ролями: Product Owner, Scrum Master, Developers.
- Встановіть тривалість спринтів – 2 тижні.
- Додайте релізний календар із ключовими фазами: планування, тестування, реліз.

Етап 2. Формування беклогу.

Крок 1. Додайте ключові епіки (наприклад):

- «Розробка інтерфейсу користувача»
- «Інтеграція CRM з зовнішніми системами»
- «Налаштування бізнес-логіки».

Крок 2. Розбийте кожен епік на User Story (наприклад):

- Епік «Інтерфейс користувача» → «Як користувач системи, я хочу мати можливість, щоб».

Крок 3. Додайте підзадачі для кожної User Story.

Крок 4. Здійсніть пріоритизація беклогу, ставлячи найважливіші елементи для MVP спочатку.

Етап 3. Оцінка завдань.

Крок 1. Оцініть складність кожної User Story у Story Points:

- «Форма реєстрації» – 3 Story Points.
- «Інтеграція з базою даних» – 8 Story Points.

Крок 2. Визначте загальну кількість Story Points, яку команда може виконати за спринт, на основі попередньої статистики (наприклад, 30 Story Points за спринт).

Етап 4. Проведення спринту.

Крок 1. Запустіть перший спринт, додавши в нього User Stories на 30 Story Points.

Крок 2. Щодня проводьте Daily Scrum:

- відслідковуйте статуси завдань: To Do, In Progress, In Review, Done.

Крок 3. Використовуйте Burndown Chart для моніторингу виконання завдань.

Етап 5. Завершення спринту.

Крок 1. Проведіть Sprint Review:

- оцініть виконані завдання та визначте, які залишилися у беклозі.

Крок 2. Проведіть Sprint Retrospective:

- визначте проблеми та запропонуйте покращення для наступного спринту.

Етап 6. Сформувані короткий висновок за результатами виконання завдання 1.

Завдання 2. Управління Kanban-проектом у Jira

Хід виконання:

Етап 1. Створення Kanban-проекту.

Крок 1. Увійдіть у Jira та створіть новий Kanban-проект під назвою (наприклад): «Розробка інформаційної системи для підтримки хмарної інфраструктури».

Крок 2. Створіть Kanban-дошку з наступними колонками (наприклад) (рис.2.42):

- «List of Tasks».
- «To Do».
- «In Progress».
- «Code Review».
- «Done».

Етап 2. Створення завдань.

Крок 1. Додайте завдання типу Task та Bug:

- «Моніторинг серверів» (Task).
- «Виправлення багу в системі бекапів» (Bug).

Крок 2. Визначте пріоритети для кожного завдання:

- «Моніторинг серверів» – середній пріоритет.

- «Виправлення багу в системі бекапів» – високий пріоритет.

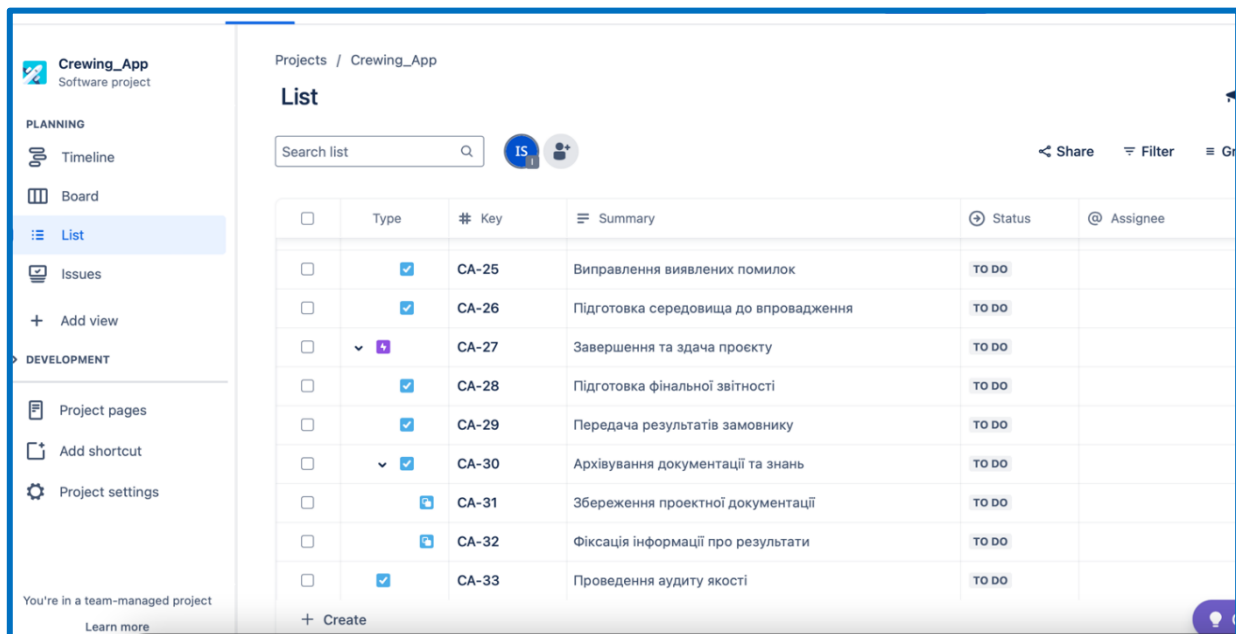


Рис. 2.42 – Створення Kanban-дошки для обраного проекту

Етап 3. Налаштування WIP лімітів.

Крок 1. Встановіть WIP ліміти для колонки «In Progress» – не більше 3 завдань одночасно.

Крок 2. Додайте до колонки «In Progress» завдання та слідкуйте за дотриманням лімітів.

Етап 4. Виконання завдань.

Крок 1. Переводьте завдання між колонками (наприклад):

- коли розпочинаєте роботу над завданням, перемістіть його до «In Progress»;
- після завершення – до «Code Review», а потім – до «Done».

Етап 5. Оцінка процесу.

Крок 1. Регулярно переглядайте поточний статус завдань на Kanban-дошці.

Крок 2. Визначте вузькі місця в процесі та за потреби корегуйте розподіл завдань.

Етап 6. Сформулювати короткий висновок за результатами виконання завдання 2.

Контрольні запитання до розділу

1. Що таке Jira і яку роль вона відіграє в управлінні проєктами?
2. Які основні функції Jira для управління проєктами?
3. Як у Jira організована структура проєктів, епік, завдань і підзадач?
4. Що таке Scrum і Kanban дошки в Jira, і в чому їх відмінність?
5. Як створювати і налаштовувати проєкти в Jira?
6. Як у Jira використовуються спринти для управління проєктами в Scrum?
7. Що таке Workflow у Jira, і як можна його налаштувати для команди?
8. Як відслідковувати прогрес команди за допомогою дашбордів у Jira?
9. Як у Jira налаштовуються фільтри для пошуку та організації завдань?
10. Як використовуються типи завдань у Jira: Epic, Story, Task, Bug?
11. Що таке backlog у Jira і як його управляти?
12. Як у Jira проводиться оцінка завдань за допомогою Story Points?
13. Як використовувати функціонал звітів у Jira для аналізу продуктивності команди?
14. Що таке Roadmap у Jira і як він допомагає планувати довгострокові цілі проєкту?
15. Як налаштувати повідомлення та сповіщення у Jira для команди?
16. Які інструменти інтеграції доступні в Jira для покращення процесів?
17. Як Jira допомагає в управлінні ризиками у проєктах?
18. Які кроки необхідно зробити для налаштування прав доступу в Jira?
19. Як у Jira проводити автоматизацію задач та процесів за допомогою автоматичних правил?
20. Як Jira може бути інтегрована з іншими інструментами для покращення управління проєктами?

РОЗДІЛ 3. ОРГАНІЗАЦІЯ ПРОЦЕСІВ УПРАВЛІННЯ ПРОЄКТАМИ ШЛЯХОМ ВИКОРИСТАННЯ ВЕБ-ПЛАТФОРМИ USPACU

3.1. Основні можливості Uspacy для управління проєктами

Uspacy – це сучасний українськомовний застосунок, створений для оптимізації управління проєктами та командною роботою. Він орієнтований на забезпечення максимальної зручності та ефективності для проєктних команд, бізнесів та організацій, що працюють у різних сферах. Завдяки інтуїтивному інтерфейсу та широкому набору інструментів, *Uspacy* допомагає командам виконувати завдання вчасно, слідкувати за прогресом, взаємодіяти між собою та контролювати навантаження. Загальний стартовий інтерфейс роботи з даною веб-платформою (процес реєстрації) наведено на рис. 3.1.

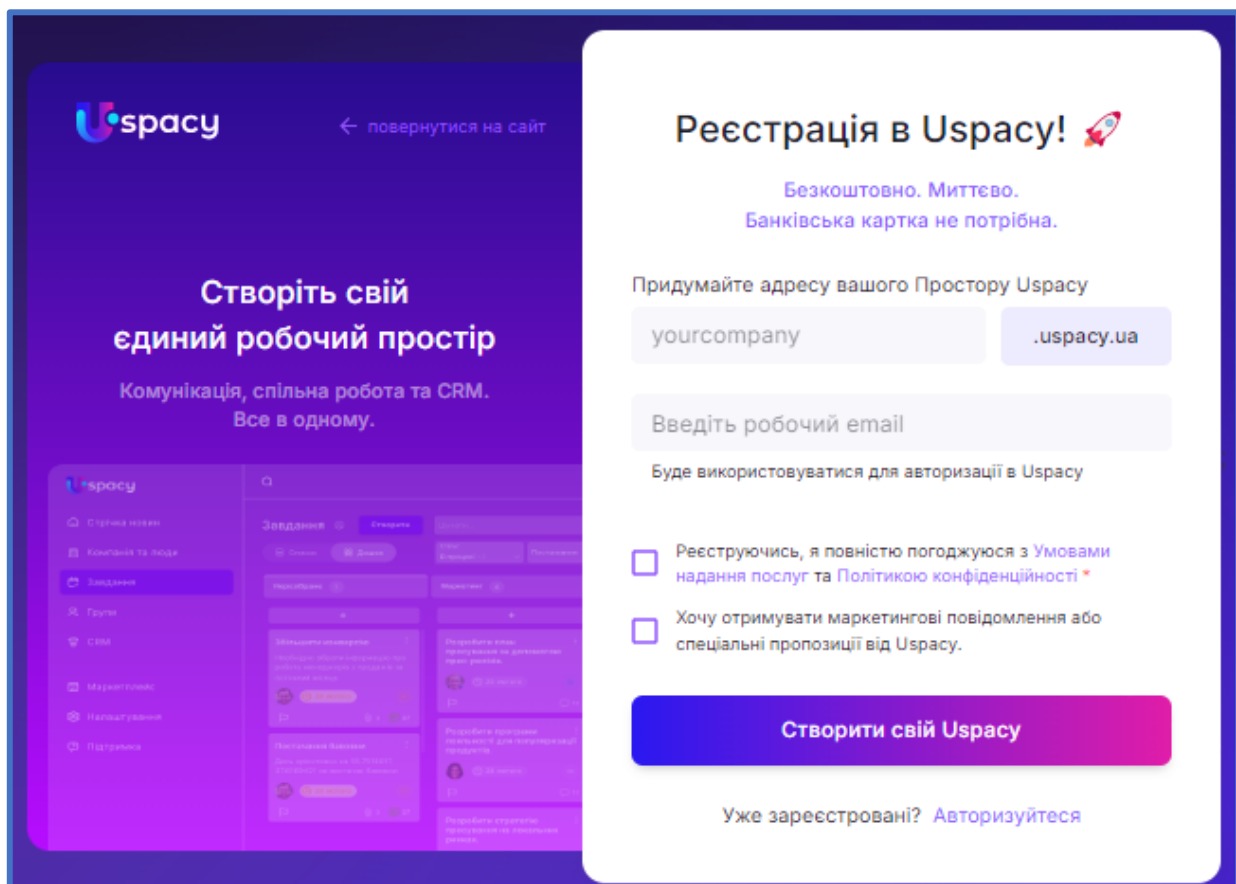


Рис. 3.1 – Реєстрація в Uspacy

Uspacy – це інноваційна інформаційна система, що стала справжнім союзником для команд, залучених до управління проєктами. Її потужні

можливості не лише спрощують процеси, але й надихають на ефективну співпрацю, що є критично важливим у динамічному світі технологій. Розглянемо декілька основних характеристик та можливостей використання *Uspacy*. Вигляд стартової сторінки робочого середовища *Uspacy* наведено на рис. 3.2.

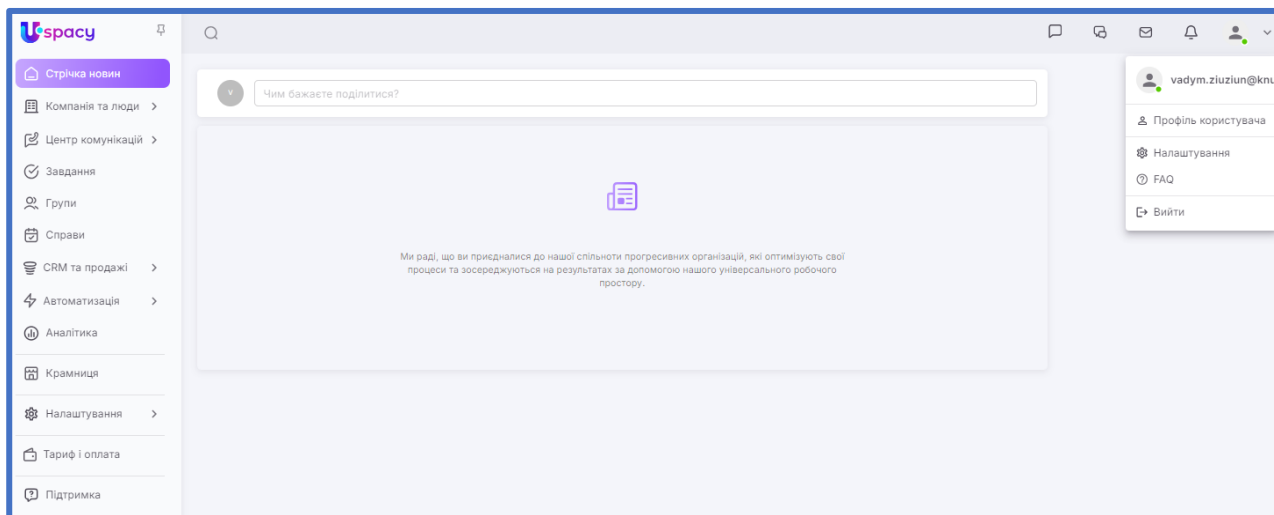


Рис. 3.2 – Стартова сторінка робочого середовища *Uspacy*

1. Планування проєктів. *Uspacy* відкриває перед користувачами можливість створення детальних планів проєктів, які стають основою для успішного виконання завдань. Завдяки інтуїтивно зрозумілому інтерфейсу, користувачі можуть визначати завдання, встановлювати терміни та формувати етапи виконання, надаючи проєкту чітку структуру і напрямок.

2. Координація команди. Співпраця в команді набуває нових обертів завдяки функціям комунікації, які пропонує *Uspacy*. Через вбудовані чати, коментарі та можливість обміну файлами, члени команди можуть вільно ділитися думками та ідеями, що сприяє швидкому вирішенню питань і покращує загальну атмосферу спільної роботи.

3. Моніторинг прогресу. *Uspacy* надає потужні інструменти для відстеження виконання завдань та загального прогресу проєкту. Користувачі можуть спостерігати за динамікою виконання, що дозволяє своєчасно виявляти затримки і вживати необхідних заходів для їх усунення.

4. **Управління ресурсами.** Управління ресурсами – це одна з ключових функцій *Uspacy*, яка допомагає командам ефективно організувати свою діяльність. Платформа дозволяє планувати навантаження на кожного члена команди, контролюючи використання часу та матеріалів, що забезпечує оптимізацію робочого процесу.

5. **Звіти та аналітика.** *Uspacy* пропонує можливості для генерації звітів, які стають безцінними інструментами для аналізу продуктивності команди. Користувачі можуть отримувати глибоке розуміння результатів виконання проекту, що допомагає приймати обґрунтовані рішення на основі даних.

6. **Інтеграції з іншими інструментами.** У сучасному світі, де інформація та ресурси розподілені між різними платформами, *Uspacy* забезпечує зручну інтеграцію з популярними інструментами, такими як CRM-системи, електронна пошта та інші платформи управління проектами. Це дозволяє створити єдину екосистему, де кожен елемент працює в гармонії.

7. **Гнучкість у налаштуваннях.** *Uspacy* відзначається своєю гнучкістю, дозволяючи користувачам налаштовувати інтерфейс і робочі процеси відповідно до специфічних потреб команди чи проекту. Ця універсальність робить платформу ідеальною для різних галузей і стилів роботи.

8. **Трекінг завдань.** Користувачі можуть безперешкодно створювати, редагувати та призначати завдання, а також контролювати їх виконання. Цей функціонал забезпечує прозорість у розподілі роботи та допомагає підтримувати організацію в команді.

9. **Календар і тайм-менеджмент.** Вбудований календар *Uspacy* стає надійним помічником для планування важливих подій, дедлайнів та зустрічей. Це допомагає командам залишатися на плаву в потоці завдань та ефективно управляти своїм часом.

10. **Управління ризиками.** Платформа також надає можливості для ідентифікації, оцінки та управління ризиками, які можуть вплинути на успішність проекту. Користувачі можуть документувати потенційні загрози та

розробляти стратегії для їх мінімізації, забезпечуючи таким чином більшу впевненість у виконанні проєктів.

Uspacy є потужним інструментом для управління проєктами, що сприяє не лише підвищенню ефективності, а й зміцненню командного духу. Використання цієї платформи може суттєво полегшити процеси управління та забезпечити успішне виконання проєктів, відкриваючи нові горизонти для інновацій та розвитку в динамічному світі технологій.

3.1.1. Інтеграція управління завданнями та комунікації

Управління зв'язками між задачами та коментарями в *Uspacy* є важливою складовою ефективного виконання проєктів. Коментарі дозволяють команді обговорювати деталі, ставити запитання та надавати зворотний зв'язок, що підвищує загальну продуктивність і прозорість процесу. Щоб оптимально працювати із зв'язком між задачами та коментарями, варто дотримуватися кількох простих етапів.

Перш за все, щоб розпочати роботу в системі, відкрийте *Uspacy* у своєму браузері та увійдіть у свій обліковий запис, ввівши логін і пароль. Після успішного входу ви потрапите на головну сторінку, де вам потрібно вибрати проєкт, з яким ви плануєте працювати. Зазвичай у лівій частині екрана знаходиться панель, на якій відображаються всі ваші проєкти. Просто натисніть на назву відповідного проєкту.

Після вибору проєкту вам слід знайти задачу, до якої ви хочете додати коментар. Це можна зробити як зі списку завдань, так і з *Kanban-дошки* (рис. 3.3), якщо вона використовується в проєкті. Клацнувши на конкретну задачу, ви відкриєте вікно або панель із деталями цієї задачі. У цьому вікні рухайтесь до низу до секції коментарів, яка зазвичай розташована в нижній частині.

У секції коментарів знайдіть поле для введення тексту, де ви можете ввести свій коментар. Після введення тексту натисніть кнопку «*Надіслати*» (рис. 3.4). Ваш коментар з'явиться у списку коментарів, пов'язаних із цим завданням.

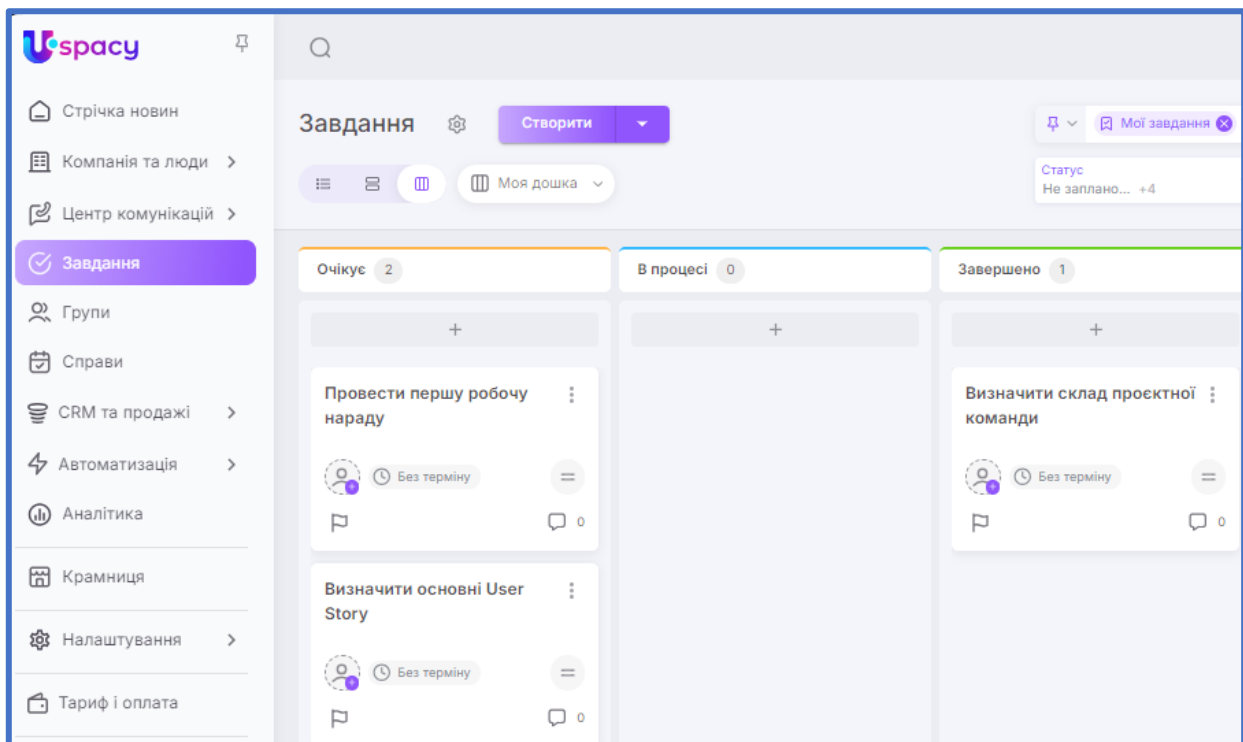


Рис. 3.3 – Робота із завданнями у форматі дошки (за прикладом Канбан-дошки)



Рис. 3.4 – Середовище формування коментаря

Крім того, ви зможете переглядати всі коментарі, що стосуються даної задачі, і відповідати на них, якщо виникне така необхідність. Для цього знайдіть потрібний коментар і натисніть кнопку «Відповісти», щоб додати своє повідомлення.

Якщо вам потрібно редагувати чи видалити свій коментар, знайдіть його у списку коментарів. Для редагування натисніть на кнопку «Редагувати» (може

бути позначена як іконка олівця) біля вашого коментаря, додайте потрібні зміни та натисніть «Надіслати». Якщо ж ви вирішите видалити коментар, просто натисніть на кнопку «Видалити» (може бути позначена як іконка смітника) і надайте підтвердження щодо видалення.

Uspacy також має систему сповіщень, яка дозволяє користувачам бути в курсі нових коментарів до їхніх задач. Зазвичай сповіщення відображаються на панелі, позначеній як «Сповіщення». Рекомендується регулярно перевіряти цю панель, щоб не пропустити важливі коментарі чи запитання.

Коли ви завершите роботу над задачею і всі обговорення будуть завершені, ви зможете закрити задачу. Для цього у вікні задачі натисніть кнопку «Завершити» (рис. 3.5).

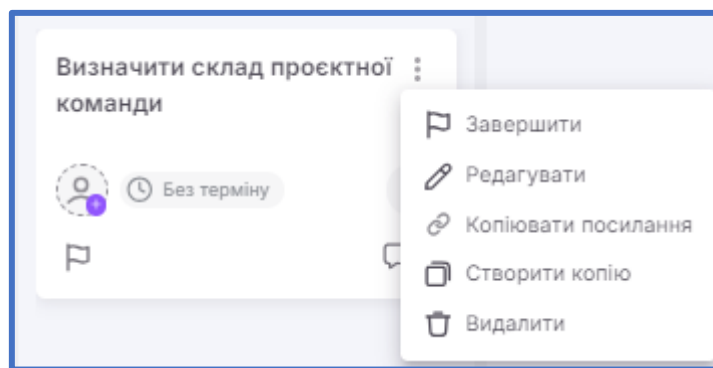


Рис. 3.5 – Інструментарій завершення завдання

Цей етап важливий для підтримки порядку в проєкті та чіткого розуміння того, які задачі вже виконані.

Розглянемо декілька **прикладів** використання зв'язку між задачами та коментарями в *Uspacy*:

1. Обговорення деталей задачі. Припустимо, ваша команда працює над розробкою нового функціоналу. Один із членів команди додає коментар до задачі, запитуючи про специфікації. Інші учасники команди можуть відповісти на це запитання, надаючи додаткову інформацію.
2. Зворотний зв'язок. Після завершення певної задачі, коментар може містити зворотний зв'язок від керівника проєкту. Він може похвалити команду за ефективність та вказати на можливі напрямки для вдосконалення.

3. Обговорення проблем. Якщо під час виконання задачі виникають труднощі, учасник команди може залишити коментар із детальним описом проблеми, на що інші члени команди зможуть відреагувати, обговоривши можливі рішення.
4. Закриття задачі. Коли всі обговорення завершено, останній коментар може містити підтвердження, що всі вимоги виконані, що дозволяє команді закрити задачу без жодних зауважень.

Таким чином, *Uspacy* надає зручний інструмент для управління завданнями та комунікацією, що сприяє більшій ефективності та злагодженості в роботі. Правильне використання цієї функції забезпечує прозорість процесів і допоможе уникнути непорозумінь у команді.

3.1.2. Панелі інструментів для проєктів

Панелі інструментів у системі *Uspacy* відіграють ключову роль у спрощенні управління проєктами, надаючи користувачам швидкий доступ до основних функцій і інструментів, необхідних для ефективної роботи. Щоб максимально використати можливості панелей інструментів, важливо знати, як правильно ними користуватися.

Почати роботу потрібно з того, щоб відкрити *Uspacy* у вашому браузері та увійти до свого облікового запису, ввівши логін і пароль. Після успішного входу ви потрапите на головну сторінку системи. Тут важливо вибрати проєкт, з яким плануєте працювати. Вам потрібно просто натиснути на назву відповідного проєкту в списку, що відображається на панелі ліворуч або на головній панелі (рис. 3.6).

Коли ви потрапите у вікно обраного проєкту, зверніть увагу на панель інструментів, яка зазвичай розташована вгорі або з боку екрана. Ця панель містить різні кнопки, які дозволяють управляти проєктом і виконувати необхідні дії. Однією з перших дій, яку ви, напевно, матимете бажання здійснити, стане створення нового завдання. Для цього знайдіть кнопку «Завдання» на панелі інструментів і натисніть на неї. Відкриється вікно, де вам потрібно буде

заповнити інформацію про завдання: ввести назву, терміни виконання, відповідальних осіб і будь-які інші деталі, що стосуються задачі.

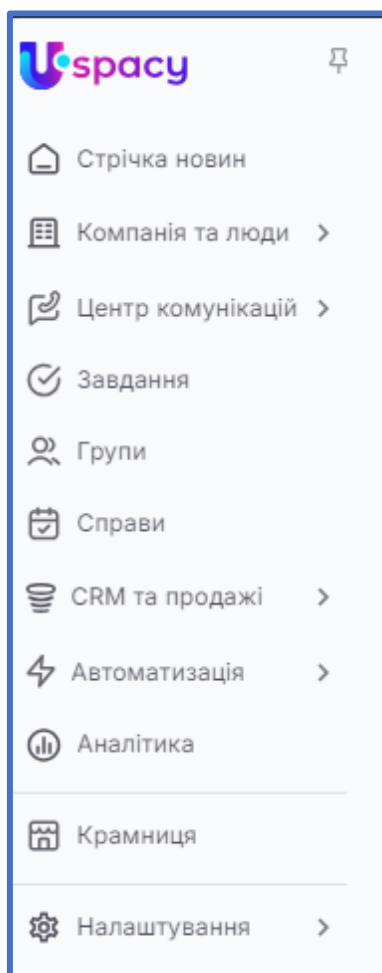


Рис. 3.6 – Панель інструментів Uspasy

Наступним кроком є використання календаря проєкту. Знайдіть кнопку «Терміни» на панелі інструментів і натисніть на неї, щоб відкрити календар. У календарі ви зможете переглядати терміни виконання завдань. Натисніть на конкретну дату, щоб побачити заплановані події та терміни, що дозволить вам краще планувати свою роботу та слідкувати за дедлайнами.

Комунікація в рамках проєкту є важливою складовою успішного виконання завдань. Щоб відкрити інструменти для спілкування, знайдіть кнопку «Центр комунікацій» на панелі інструментів. Натиснувши на цю кнопку, ви отримаєте доступ до чату або коментарів, що допоможе вам оперативно вирішувати питання та обговорювати деталі з командою.

Для моніторингу прогресу проєкту важливо регулярно переглядати звіти. Для цього знайдіть кнопку «Звіти» на панелі інструментів. Натиснувши на цю кнопку, ви зможете отримати доступ до звітів, які надають інформацію про виконання завдань і загальний стан проєкту.

Якщо ви плануєте додати зміни в налаштування проєкту, вам слід знайти кнопку «Налаштування» на панелі інструментів. Натиснувши на цю кнопку, ви відкриєте вікно налаштувань, де зможете відредагувати параметри проєкту відповідно до своїх потреб.

Після завершення роботи над проєктом важливо не забути вийти з системи. Для цього натисніть «Вийти» або «Вихід» на панелі інструментів, щоб завершити сесію.

Розглянемо декілька **прикладів** використання панелей інструментів в *Uspacy*:

1. Створення нових завдань. Припустимо, ваша команда запускає новий проєкт. Ви натискаєте кнопку «Додати завдання», вводите деталі та призначаєте відповідальних осіб, щоб одразу ж почати роботу.
2. Перегляд термінів. У вас є проєкт, де декілька завдань мають однаковий дедлайн. Ви відкриваєте календар, щоб переглянути всі заплановані події на цю дату і впевнитися, що у вас є достатньо часу для їх виконання.
3. Комунікація з командою. Коли виникає питання щодо виконання задачі, ви відкриваєте інструменти комунікації, щоб надіслати термінове повідомлення вашій команді, забезпечуючи швидкий зворотний зв'язок.
4. Моніторинг прогресу. Ви регулярно перевіряєте звіти про прогрес проєкту, щоб оцінити, як виконується робота, і вчасно вжити заходів, якщо проєкт починає відставати від графіка.

Використання панелей інструментів в *Uspacy* забезпечує зручний доступ до всіх необхідних функцій для ефективного управління проєктами, спрощуючи організацію роботи, планування завдань і комунікацію в команді.

3.2. Управління командами та завданнями

3.2.1. Створення команд та розподіл задач

Управління командами та завданнями в системі *Uspacy* є ключовим аспектом для забезпечення успішного виконання проєктів. Ця платформа пропонує зручний інтерфейс, що дозволяє адміністраторам легко управляти ролями учасників, призначати завдання та контролювати прогрес роботи.

Після відкриття проєкту, зверніть увагу на панель інструментів, де знайдете кнопку «Компанія та люди». Натиснувши на цю кнопку, ви відкриєте вікно з інформацією про всіх учасників проєкту, що дозволяє вам легко відстежувати їхній статус і ролі.

Якщо вам потрібно додати нових учасників до команди, натисніть кнопку «Співробітники» → «Список співробітників» → «Запросити». Введіть *e-mail* або ім'я людини, яку ви хочете запросити, а потім натисніть «Запросити співробітників». Цей процес забезпечує швидке поповнення команди новими фахівцями (рис. 3.7).

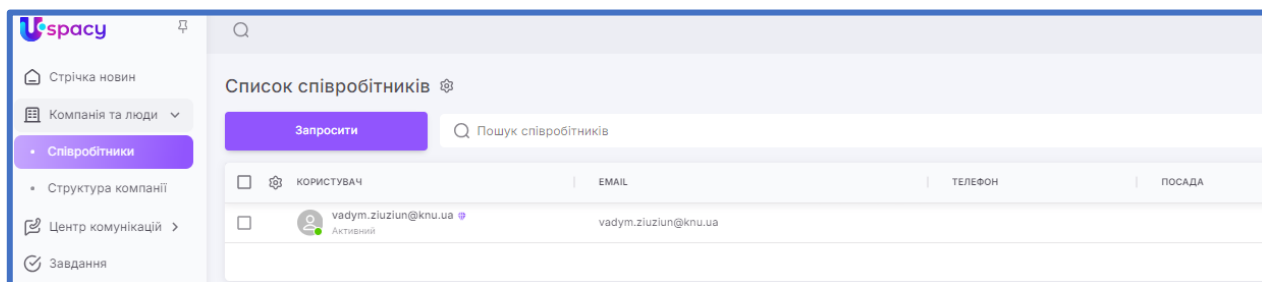


Рис. 3.7 – Панель інструментів *Uspacy*

Якщо ж потрібно видалити когось із проєкту, знайдіть кнопку «Звільнити» поруч із іменем учасника. Підтвердіть дію, натиснувши «Звільнити» або «Підтвердити звільнення». Ці дії гарантують, що ваша команда залишається актуальною та ефективною.

Після того, як команда сформована, важливо правильно призначити ролі учасникам. У списку учасників ви зможете бачити інформацію про кожного члена. Щоб призначити роль, натисніть на випадне меню поруч із іменем учасника (зазвичай це невелика стрілка або кнопка з текстом «Посада»). У

спадному списку з'являться різні варіанти ролей, які можуть включати «*Product Owner*», «*Scrum Master*», «*Developer*» або «*QA*». Виберіть відповідну роль для кожного учасника, що дозволяє зручно контролювати їхні обов'язки та доступи.

Також з налаштуваннями ролей можна працювати натиснувши «*Налаштування*», далі «*Ролі та права*» (рис. 3.8).

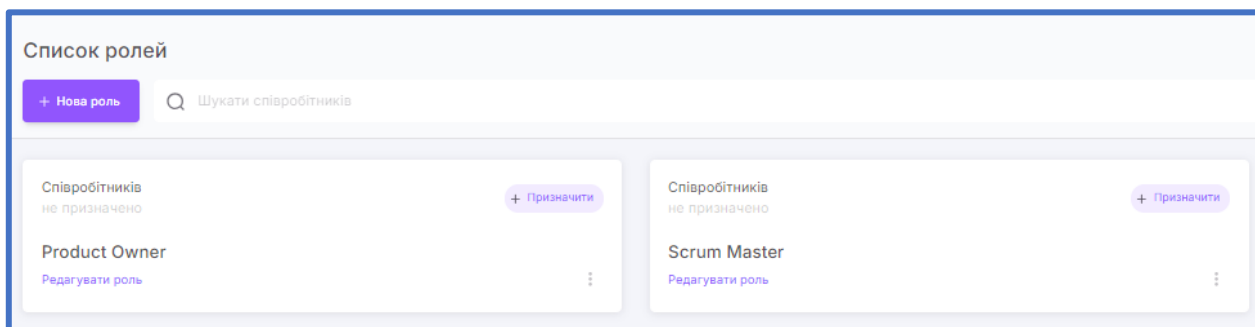


Рис. 3.8 – Адміністрування категорії «Ролі та права»

Зміни можуть зберігатися автоматично, але важливо перевірити, чи всі ролі призначені правильно. У майбутньому, якщо вам потрібно змінити ролі учасників, просто повторіть ці дії. Завжди можна повернутися до панелі управління командою, щоб змінити ролі або додати нових учасників.

Якщо виникне потреба видалити когось із команди, на панелі управління командою поруч із ім'ям учасника також може бути іконка «*Видалити*» або «*Видалити з проекту*». Ролі в *Uspasy* забезпечують гнучкість і контроль над роботою, допомагаючи команді ефективніше співпрацювати.

Управління завданнями

Для управління завданнями, знайдіть кнопку «*Завдання*» на панелі інструментів вашого проекту. Натиснувши на цю кнопку, ви отримаєте доступ до списку всіх завдань, пов'язаних із проектом. Щоб створити нове завдання, натисніть кнопку «*Створити завдання*» (рис. 3.9). У відкритому вікні введіть назву завдання, опис, терміни виконання та призначте відповідальних осіб. Цей процес допомагає вам чітко сформулювати завдання та визначити відповідальних.

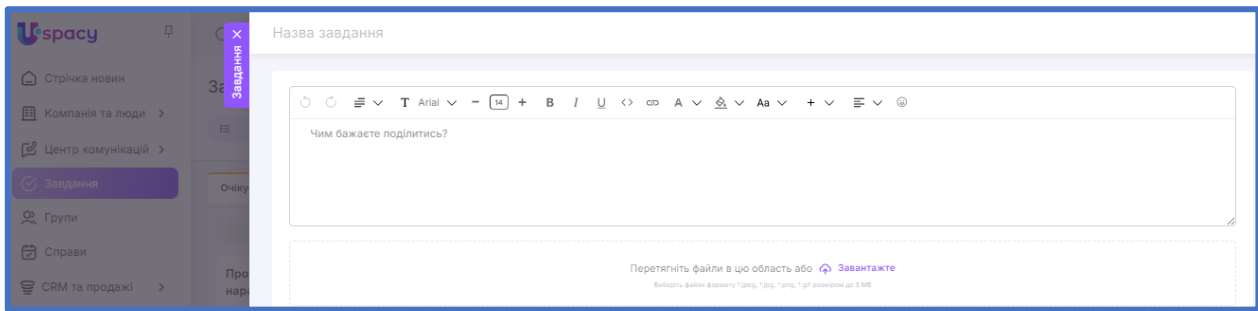


Рис. 3.9 – Вікно для створення нового завдання

При створенні або редагуванні завдання, ви можете призначити його конкретному учаснику. Виберіть ім'я учасника з випадного списку в полі «Відповідальний». Щоб додати зміни в завдання, просто клацніть на його назву у списку. Це відкриє вікно з деталями завдання, де ви зможете оновити терміни, змінити відповідальну особу або додати нові підзадачі.

Uspacy дозволяє вам легко відстежувати прогрес виконання завдань. У списку завдань або на *Kanban*-дошці ви можете бачити статус кожного завдання, наприклад, «В процесі», «Завершено» або «Очікує» (рис. 3.10). Коли завдання буде виконано, ви можете його закрити, натиснувши на завдання, а потім на кнопку «Завершити» або «Закрити завдання».

Розглянемо декілька **прикладів** управління командами та завданнями в *Uspacy*:

1. Додавання нового учасника. Ви вирішили розширити команду, щоб впоратися зі збільшенням обсягу роботи. Після переходу до проєкту, натискаєте «Співробітники», потім «Запросити», вводите *e-mail* нового члена команди, і натискаєте «Запросити співробітника». Після отримання запрошення новий учасник приєднується до вашого проєкту.
2. Призначення ролі. У вашій команді є новий розробник. У списку учасників ви знаходите його ім'я, натискаєте на випадне меню поруч із ним і обираєте роль «Розробник». Це гарантує, що він матиме доступ до всіх необхідних ресурсів для виконання своїх завдань.
3. Редагування завдання. Після обговорення прогресу проєкту з командою ви вирішуєте змінити термін виконання завдання «Тестування модуля А».

Клацнувши на його назву, ви відкриваєте деталі завдання, змінюєте термін з 3 днів на 5 днів, і натискаєте «Зберегти». Таким чином, ви зберігаєте актуальність інформації.

4. Закриття завдання. Після успішного завершення завдання «Розробка модуля В», ви натискаєте на його назву, а потім на кнопку «Завершити». Це сигналізує всій команді про те, що завдання виконано, і дозволяє зосередитися на нових етапах проєкту.

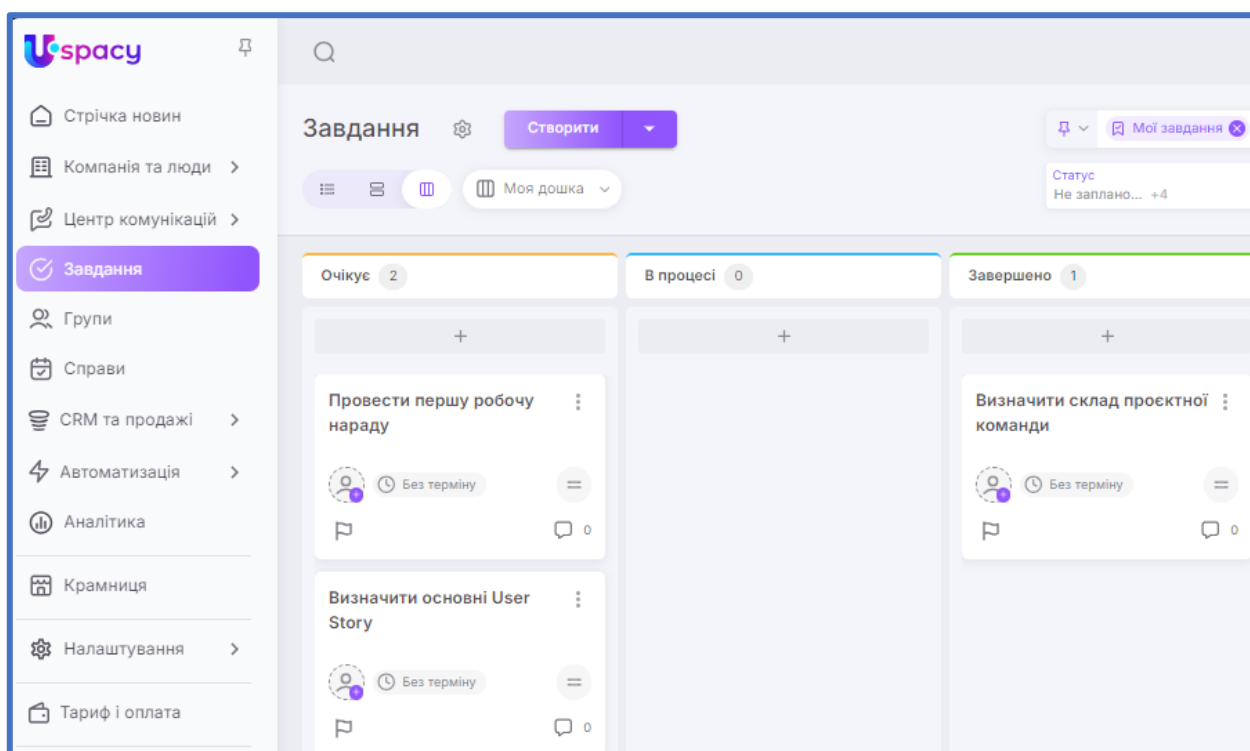


Рис. 3.10 – Вікно відображення завдань та статусів на Kanban

Таким чином, *Uspacy* пропонує потужні інструменти для управління командами та завданнями, забезпечуючи зручність та ефективність у роботі над проєктами.

3.2.2. Спостереження за навантаженням команди

Управління робочим навантаженням команди є важливою складовою успішного завершення проєктів. Система *Uspacy* надає зручний інтерфейс для моніторингу та аналізу навантаження учасників команди, що дозволяє менеджерам проєктів ефективно організувати роботу.

Для початку роботи у списку розділів вашого проекту знайдіть той, який присвячений навантаженню команди. Клацніть на кнопку або вкладку з назвою «Справи» (рис. 3.11). Ця вкладка надає інформацію про робоче навантаження кожного учасника команди, дозволяючи вам переглядати кількість активних завдань, їх статус виконання, терміни завершення та рівень зайнятості.

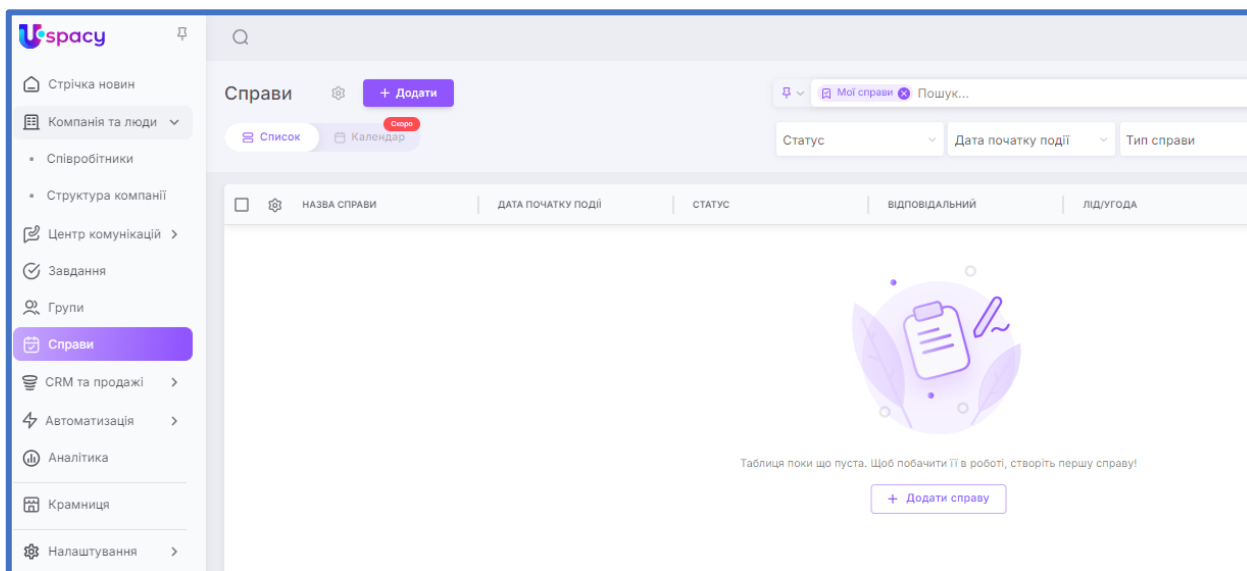


Рис. 3.11 – Вікно для управління категорією «Справи»

Для детальнішого аналізу ви можете фільтрувати інформацію за певними періодами, такими як тиждень або місяць, щоб побачити зміни в навантаженні команди в конкретні часові рамки. Скористайтеся кнопкою вибору періоду «Календар», щоб отримати уявлення про те, як робоче навантаження змінюється на різних етапах проекту.

Якщо вам потрібно детальніше розглянути окремі завдання, просто клацніть на назву завдання в списку навантаження. Натиснувши на завдання, ви зможете переглянути його статус, терміни, відповідальних осіб та інші важливі деталі. Це дозволяє швидко оцінити стан виконання завдань.

Для аналізу вільного часу команди ви можете переглядати графіки завантаженості або кругові діаграми, що показують поточний стан зайнятості учасників. Використовуйте візуалізації, щоб легко виявити, які члени команди перенавантажені, а хто має вільні ресурси для нових завдань.

На завершення, якщо вам потрібно поділитися інформацією про навантаження команди з іншими або зберегти звіт, *Uspacy* пропонує функцію експорту даних. Просто натисніть кнопку «Повний експорт» (рис. 3.12), оберіть бажаний формат, наприклад, PDF або Excel, і збережіть звіт для подальшого використання.

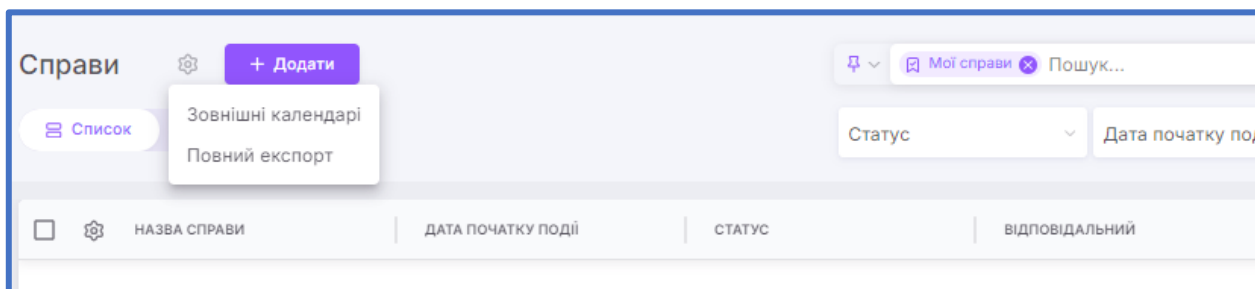


Рис. 3.12 – Проведення процедури експорту щодо справ учасників команди

Якщо ви виявили, що певні учасники команди перевантажені, варто приймати рішення щодо коригування навантаження. Ви можете перерозподілити завдання, редагуючи їх і призначаючи нових відповідальних. Це дозволить оптимізувати робочий процес і уникнути вигорання.

Розглянемо декілька *прикладів* спостереження за навантаженням команди в *Uspacy*:

1. Перегляд навантаження команди. Ви щодня перевіряєте вкладку «Навантаження команди», щоб відстежувати, як йдуть справи у вашій команді. Наприклад, ви помічаєте, що один з розробників має п'ять активних завдань, тоді як інший лише два. Це викликає занепокоєння, і ви вирішуєте вжити заходів для перерозподілу.
2. Фільтрація за періодами. Під час щотижневої наради з командою ви вирішуєте подивитися, як змінилося навантаження за останній місяць. За допомогою функції фільтрації за періодами ви встановлюєте діапазон дат з 1 по 30 вересня і виявляєте, що у третій тиждень навантаження команди різко зросло, що потребує додаткового аналізу.
3. Перерозподіл завдань. Після перегляду графіка завантаженості ви помічаєте, що один з членів команди має перевантаження. Ви вирішуєте

призначити деякі його завдання іншому розробнику, щоб зменшити навантаження. Це дає можливість зберегти баланс в роботі команди.

4. Експорт даних. Наприкінці місяця вам потрібно підготувати звіт для керівництва. Ви використовуєте функцію експорту, щоб згенерувати документ у форматі PDF, який містить детальну інформацію про навантаження команди, а також графіки виконання завдань. Це допомагає вам підкріпити свою доповідь конкретними даними.

У підсумку, система *Uspacy* пропонує потужні можливості для моніторингу навантаження команди, що дозволяє менеджерам проєктів оперативно виявляти проблеми, оптимізувати ресурси та забезпечити ефективну роботу.

3.2.3. Відстеження прогресу по завданнях

Розпочавши роботу зверніть увагу на вкладку «Завдання», яка зазвичай розташована на верхній панелі або в бічному меню. Перейдіть у цей розділ, щоб переглянути всі активні завдання. *Uspacy* дозволяє зручно фільтрувати завдання за статусом, наприклад, «В роботі», «Завершені», або «Очікують». Це допоможе вам швидко відсіяти зайве і зосередитися на тих задачах, які потребують вашої уваги.

У кожному завданні вказано відсоток його виконання або поточний статус, що дозволяє вам миттєво зрозуміти, наскільки просунувся процес. Щоб переглянути деталі, натисніть на завдання, де ви знайдете такі важливі показники, як призначений термін виконання, відповідальні особи та інші метрики. Зокрема, ви зможете відкрити вкладки «Центр комунікації», «Завдання» та «Справи», де детально представлені останні зміни, додані файли та комунікація між членами команди (рис. 3.13).

Якщо вам необхідно оновити статус завдання, це можна зробити всередині його детальної сторінки. Натисніть на поточний статус, наприклад, «Призупинено» і виберіть інший із запропонованого списку, як-от «Готово» або «Завершено». Це допоможе швидко і точно відслідковувати зміни у процесі (рис. 3.14). Але також дуже зручно статуси відслідковувати на *Kanban-дошці* (рис. 3.15).

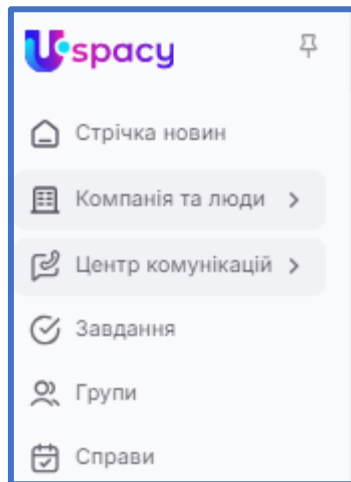


Рис. 3.13 – Навігація роботи із вкладками «Центр комунікації», «Завдання» та «Справи»

Щоб отримати загальну картину прогресу по всьому проєкту, перейдіть до вкладки «Аналітика». Тут ви зможете переглянути графіки та діаграми, які наочно демонструють відсоток виконаних завдань або поточний стан проєкту в цілому. Якщо вам потрібно фільтрувати завдання за виконавцями, можна скористатися додатковими фільтрами у меню завдань і вибрати конкретних учасників команди для відстеження їхньої роботи.

НОМЕР	НАЗВА ЗАВДАННЯ	КІНЦЕВИЙ ТЕРМІН	ВІДПОВІДАЛЬНИЙ	ПРІОРИТЕТ	СТАТУС
6	Модуль інтеграції з базою даних	Без терміну	vadym.zluzlun@knu.ua	=	Грипушено
5	Зустріч зі стейкхолдерами	Без терміну	vadym.zluzlun@knu.ua	=	Готово
4	Обговорення User Story 1	Без терміну	vadym.zluzlun@knu.ua	=	Не заплановано
3	Визначити основні User Story	Без терміну	+ Додати	=	Не заплановано
2	Визначити склад проєктної команди	Без терміну	+ Додати	=	Не заплановано
1	Провести першу робочу нараду	Без терміну	+ Додати	=	Не заплановано

Рис. 3.14 – Перелік завдань проєкту та робота зі статусами

Uspasy також дозволяє експортувати звіти про прогрес у різні формати, зокрема PDF або Excel. Для цього перейдіть до панелі аналітики і натисніть на кнопку «Експорт», після чого виберіть бажаний формат для подальшого зберігання або розсилки.

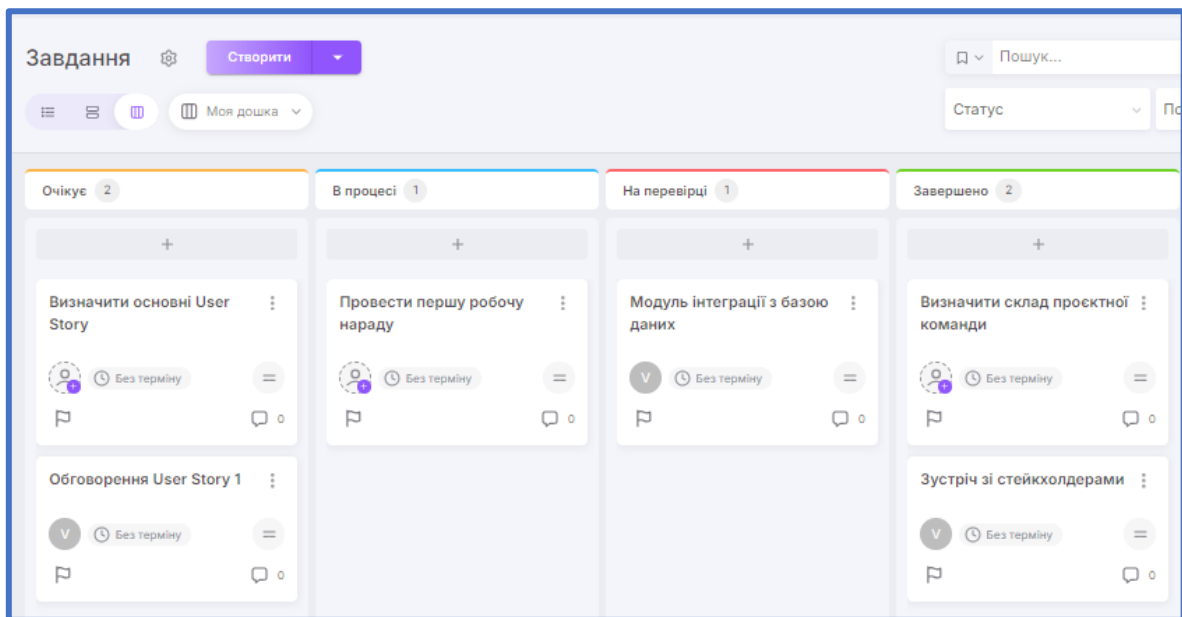


Рис. 3.15 – Перелік завдань проекту та робота зі статусами використовуючи Kanban-дошку

Наприклад, працюючи в *Uspacy* над завданням «Створення документації», ви можете відкрити це завдання, переглянути деталі виконання, термін та додаткові коментарі від команди. Після цього ви можете оновити його статус на «На перевірі» та переконатися, що всі відповіді команди і додані файли правильно відображаються у вкладці «Завдання» або «Справи».

Uspacy надає інтуїтивно зрозумілий і гнучкий підхід до відстеження та управління завданнями, дозволяючи швидко отримувати інформацію про стан проекту та забезпечуючи командну ефективність.

3.3. Комунікація та співпраця в реальному часі

На початку знайдіть розділ «Центр комунікацій» → «Зовнішні чати», «Внутрішні чати», «Пошта», який дозволяє спілкуватися з вашими колегами в реальному часі (рис. 3.16).

У цьому розділі ви зможете обмінюватися повідомленнями, ділитися думками та отримувати швидкі відповіді на ваші питання. Якщо вам потрібно обговорити конкретне завдання, ви можете прикріпити його до повідомлення в чаті. Для цього натисніть кнопку «Прикріпити завдання» або скористайтеся

вкладкою для вибору завдання зі списку проєктів. Це забезпечить зручність у роботі та допоможе зосередитися на важливих аспектах проєкту.

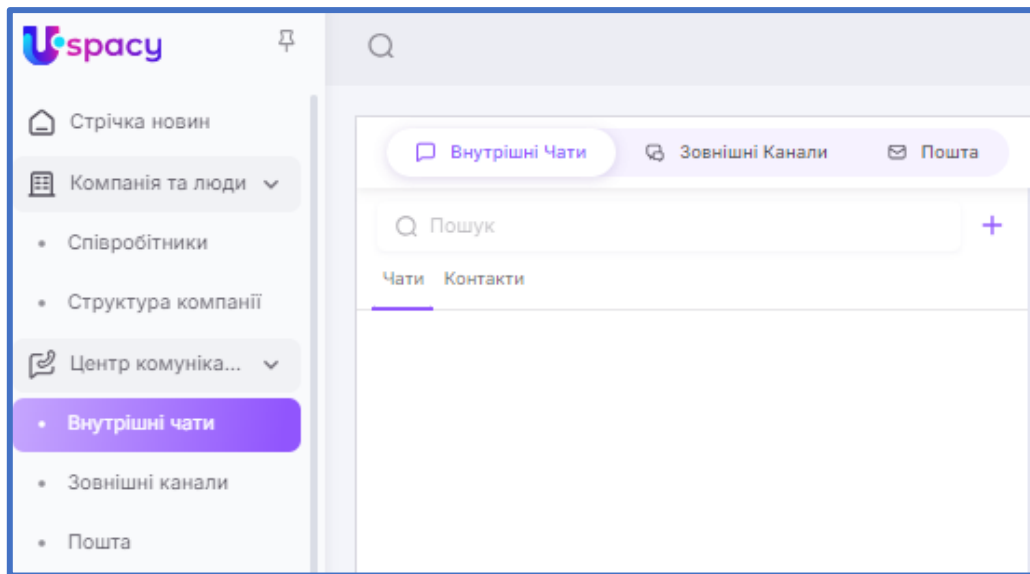


Рис. 3.16 – Робота із вкладкою «Центр комунікацій»

В *Uspasy* також доступна функція згадки учасників команди. Якщо ви хочете звернутися до конкретного члена команди, просто введіть символ «@» та почніть вводити його ім'я. Це приверне увагу необхідної особи до вашого повідомлення і спростить комунікацію.

Для більш детального обговорення завдань ви можете додавати коментарі безпосередньо в самих завданнях. Відкрийте потрібне завдання та знайдіть символ «Коментарі» внизу віконця із відповідним завданням. Тут ви можете залишити свої думки, запитання або пропозиції. У полі для введення коментаря також є можливість прикріпити документи, файли або зображення, натиснувши на іконку прикріплення файлу. Це дозволить вам надати додаткову інформацію або обговорити важливі матеріали.

Крім того, якщо ваш проєкт передбачає спільну роботу над документами, дозволяє вам завантажувати файли та надавати доступ до них для редагування або коментування. Ви можете легко завантажити необхідні документи до вкладки «Завдання» або «Справи». у рамках або проєкту, що забезпечить усім учасникам доступ до важливої інформації.

Для організації нарад або обговорень у реальному часі ви можете скористатися функцією відеоконференцій. Просто перейдіть до розділу «*Відео-дзвінки*» або натисніть на відповідну кнопку у проєкті. Це надає можливість команді обговорювати питання у візуальному форматі, що є надзвичайно корисним для прийняття рішень.

Також важливо пам'ятати про можливість планування зустрічей у вбудованому календарі. Відкрийте календар і додайте нову подію, вказуючи учасників, час та деталі зустрічі. Це дозволить всім учасникам бути в курсі запланованих подій і підготуватися до них.

Uspacy автоматично надсилає сповіщення про нові коментарі, повідомлення в чаті або інші дії, пов'язані з співпрацею в проєкті. Ви можете перевіряти сповіщення у верхній частині екрану або в панелі повідомлень, що дозволить вам не пропустити важливу інформацію.

Uspacy надає потужні інструменти для комунікації та співпраці в реальному часі. Завдяки чатам, коментарям, відеоконференціям та календарю зустрічей, команди можуть ефективно взаємодіяти та оперативно вирішувати питання, що виникають під час роботи над проєктом.

Наприклад, у проєкті, пов'язаному з організацією конференції, ви можете створити чат для обговорення всіх аспектів підготовки. Тут ви прикріпите завдання «Оренда приміщення» та згадаєте членів команди, які відповідають за це. У коментарях до цього завдання ви можете запитати про статус і прикріпити план приміщення, щоб кожен учасник міг ознайомитися з ним.

Ще один приклад: у команді, що працює над розробкою програмного забезпечення, ви можете використати відеоконференцію для обговорення нових функцій. В чаті ви прикріпите завдання «Розробка інтерфейсу користувача» та запросите дизайнера до участі. У коментарях ви зможете обговорити кольорові схеми та елементи дизайну, а також прикріпити ескізи для обговорення.

Таким чином, *Uspacy* є зручним інструментом для оптимізації комунікації та співпраці в реальному часі, дозволяючи командам досягати високих результатів у своїй роботі.

Завдання для самостійного опрацювання

Завдання 1. Створення проєкту та управління завданнями в Usrasy

Хід виконання:

Етап 1. Створення проєкту.

Крок 1. Увійдіть в Usrasy і створіть новий проєкт під назвою «Розробка інформаційної системи з автоматизації бізнес-процесів».

Крок 2. Встановіть терміни проєкту – дата початку через 2 дні та тривалість 3 місяці.

Крок 3. Додайте основні категорії для завдань (наприклад): «Планування», «Розробка», «Тестування», «Запуск».

Етап 2. Додавання завдань.

Крок 1. Додайте основні завдання до кожної категорії (наприклад):

- для планування: «Збір вимог», «Аналіз процесів», «Розробка технічного завдання».
- для розробки: «Створення модулів», «Інтеграція API», «Налаштування системи».
- Для тестування: «Функціональне тестування», «Стрес-тестування».

Крок 2. Вкажіть терміни та призначте відповідальних для кожного завдання:

- «Збір вимог» – Ім'я 1.
- «Розробка технічного завдання» – Ім'я 2.
- «Створення модулів» – Ім'я 3.

Етап 3. Використання тегів і пріоритетів.

Крок 1. Для кожного завдання додайте теги (наприклад, #критичне, #високий-пріоритет) для кращої організації.

Крок 2. Визначте пріоритети для завдань, щоб команда знала, на чому фокусуватися в першу чергу.

Крок 3. Використовуйте фільтри Usrasy, щоб відфільтрувати завдання з найвищими пріоритетами.

Етап 4. Призначення відповідальних та спільна робота.

Крок 1. Додайте до кожного завдання декількох відповідальних осіб (де це можливо) для кращої координації командної роботи.

Крок 2. Встановіть автоматичні сповіщення для учасників про зміни у завданнях.

Етап 5. Сформувати короткий висновок за результатами виконання завдання 1.

Завдання 2: Відстеження прогресу та використання інтеграцій у Uspacey

Хід виконання:

Етап 1. Оновлення статусів завдань.

Крок 1. Переконайтеся, що кожен член команди оновив статуси своїх завдань (наприклад):

- «Збір вимог» – завершено.
- «Створення модулів» – виконується (50%).
- «Функціональне тестування» – виконується (30%).

Крок 2. Відкоригуйте терміни для завдань, які зазнали затримок.

Етап 2. Використання тайм-трекінгу.

Крок 1. Увімкніть тайм-трекінг для завдань «Створення модулів» і «Інтеграція API», щоб відстежувати витрачені години.

Крок 2. Налаштуйте щотижневі звіти для автоматичного збору інформації про витрачений час для кожного члена команди.

Етап 3. Створення звіту та аналіз прогресу

Крок 1. Створіть звіт за допомогою вбудованих інструментів Uspacey, щоб оцінити загальний прогрес проекту.

Крок 2. Відобразіть у звіті виконані завдання, поточний статус невиконаних завдань і можливі ризики затримки проекту.

Крок 3. Експортуйте звіт у PDF для подальшого аналізу та обговорення на командній нараді.

Етап 4. Сформувати короткий висновок за результатами виконання завдання 2.

Контрольні запитання до розділу

1. Що таке Usrasy і для яких цілей використовується в управлінні проєктами?
2. Які основні функції платформи Usrasy для управління проєктами?
3. Як у Usrasy організуються проєкти, завдання та підзадачі?
4. Як налаштовується робоча панель (дашборд) в Usrasy для відстеження прогресу?
5. Які типи проєктних дошок пропонує Usrasy (Kanban, Scrum тощо)?
6. Як створювати і налаштовувати нові проєкти в Usrasy?
7. Як Usrasy дозволяє керувати командою та розподіляти завдання між учасниками?
8. Що таке таймлайн у Usrasy, і як він допомагає планувати проєкт?
9. Як у Usrasy працює система комунікацій та співпраці в команді?
10. Як використовуються теги та категорії для організації завдань в Usrasy?
11. Як у Usrasy можна відстежувати та керувати ризиками в проєктах?
12. Як відбувається пріоритизація завдань у Usrasy?
13. Що таке автоматизація процесів у Usrasy і як її налаштувати?
14. Як платформа Usrasy інтегрується з іншими інструментами та сервісами?
15. Як використовувати звітність у Usrasy для аналізу продуктивності команди?
16. Які можливості для управління ресурсами надає Usrasy?
17. Як у Usrasy працює система оцінки завдань та трекінг часу?
18. Як проводиться контроль дедлайнів і термінів у Usrasy?
19. Як Usrasy забезпечує безпеку даних і конфіденційність інформації в проєктах?
20. Які є інструменти для зворотного зв'язку та моніторингу виконання завдань в Usrasy?

РОЗДІЛ 4. ЕФЕКТИВНЕ УПРАВЛІННЯ ПРОЄКТАМИ ЗА ДОПОМОГОЮ ВЕБ-ПЛАТФОРМИ MONDAY.COM

4.1. Основні функціональні можливості monday.com

monday.com – це багатофункціональний веб-сервіс, який надає користувачам зручні та гнучкі інструменти для управління проєктами, командами та процесами. Він призначений для покращення організації роботи, забезпечення прозорості, контролю за виконанням завдань та автоматизації рутинних операцій. Від моменту свого запуску **monday.com** здобув визнання як один із найзручніших і найефективніших інструментів для управління проєктами та командною співпрацею. Вигляд стартової сторінки веб-платформи **monday.com** наведено на рис. 4.1.

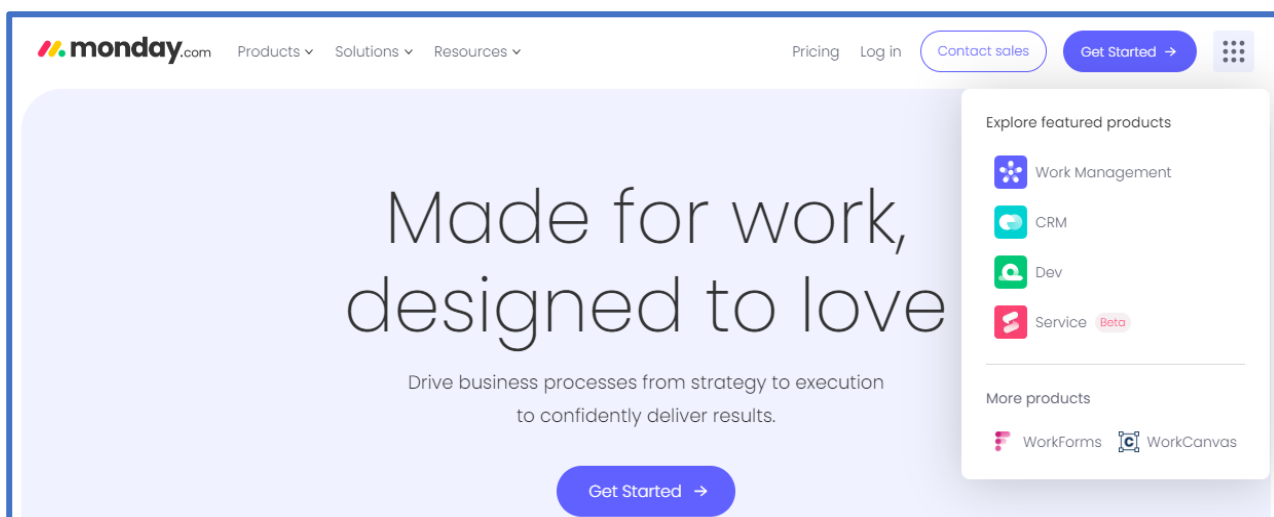


Рис. 4.1 – Початок роботи з веб-ресурсом monday.com

Розглянемо основні можливості **monday.com**.

1. *Управління проєктами.* **monday.com** пропонує зручні інструменти для планування, моніторингу та контролю всіх етапів проєкту. У системі можна створювати окремі проєкти або кампанії, розбивати їх на завдання та підзадачі, призначати відповідальних і встановлювати терміни виконання. Кожен учасник проєкту має доступ до своєї панелі управління, де може бачити прогрес, пріоритети та дедлайни. Це дозволяє уникнути плутанини та забезпечує чітку

структуру процесів.

2. *Гнучкі робочі простори (борди)*. Центральним елементом **monday.com** є так звані «борди» – робочі простори, які користувачі можуть налаштовувати під свої потреби. Борди дозволяють відслідковувати хід виконання проєктів, фіксувати різні етапи завдань, вести комунікацію з командою та збирати необхідну інформацію. Від перегляду списків до відображення процесів у вигляді календаря чи діаграм Ганта, **monday.com** пропонує кілька типів візуалізації для кращого розуміння статусу проєктів.

3. *Автоматизація рутинних процесів*. Автоматизація є ключовим елементом **monday.com**. Вона дозволяє значно скоротити час, що витрачається на виконання повторюваних завдань. Наприклад, можна налаштувати автоматичне сповіщення про зміну статусу завдання або надсилання нагадувань про наближення дедлайнів. Автоматизація допомагає усунути помилки, що виникають через людський фактор, і забезпечує постійний контроль за процесами без додаткових зусиль з боку користувачів.

4. *Колаборація та комунікація в режимі реального часу*. Одним із найбільших плюсів **monday.com** є можливість командної співпраці в режимі реального часу. Користувачі можуть коментувати завдання, залишати повідомлення для своїх колег та додавати важливі документи чи файли. Крім цього, вбудовані сповіщення про нові коментарі чи зміни статусу завдань допомагають залишатися в курсі всіх важливих подій проєкту.

5. *Аналітика та звітність*. Сервіс надає широкий спектр інструментів для збору даних і формування звітів. За допомогою діаграм і графіків можна аналізувати ефективність команди, продуктивність на кожному етапі проєкту, а також спостерігати за загальним прогресом. Користувачі можуть створювати як попередньо налаштовані звіти, так і кастомізовані для конкретних потреб.

6. *Інтеграція з іншими сервісами*. **monday.com** підтримує інтеграції з різними популярними інструментами, такими як *Slack, Google Drive, Dropbox, Jira, Zoom* та багатьма іншими. Це дозволяє організувати роботу таким чином, щоб усі важливі дані та інструменти були доступні в одному місці, що підвищує ефективність роботи.

7. *Широкі можливості кастомізації.* Ще однією перевагою **monday.com** є можливість повної кастомізації під потреби бізнесу або команди. Користувачі можуть створювати власні шаблони робочих процесів, налаштовувати різні форми представлення даних, змінювати поля та налаштовувати автоматизацію під конкретні робочі процеси. Це робить платформу універсальним інструментом, що підходить як для малих стартапів, так і для великих корпорацій.

8. *Мобільний додаток.* Для забезпечення постійного доступу до проєктів **monday.com** має зручний мобільний додаток, який дозволяє керувати завданнями та взаємодіяти з командою навіть на ходу. Це особливо корисно для тих, хто працює віддалено або часто перебуває в русі.

monday.com є надзвичайно універсальною платформою, яка підходить для управління проєктами в різних галузях: від ІТ та маркетингу до виробництва та фінансового управління. Унікальна гнучкість сервісу дозволяє адаптувати його для різних команд та бізнес-моделей:

- проєктні команди можуть використовувати **monday.com** для відстеження розробки програмного забезпечення, управління задачами та взаємодії з різними департаментами;
- маркетингові агентства часто застосовують цей інструмент для управління рекламними кампаніями, координації роботи з клієнтами та відстеження результатів;
- виробничі компанії використовують **monday.com** для моніторингу логістичних процесів, відстеження запасів і управління поставками.

monday.com – це сучасний інструмент для управління проєктами, який поєднує в собі потужну функціональність, гнучкість та простоту у використанні. Завдяки можливості автоматизації рутинних процесів, інтеграції з іншими інструментами та підтримці командної роботи в реальному часі, платформа допомагає оптимізувати робочі процеси та підвищити ефективність управління проєктами в будь-якій сфері діяльності.

4.1.1. Візуалізація робочих процесів

Створення візуальних дашбордів у *monday.com* – це процес, який дозволяє командам наочно відображати дані з різних проєктів і бордів. Для початку вам потрібно увійти до свого облікового запису на платформі. Після успішного входу ви зможете обрати проєкт або борд, для якого плануєте створити дашборд, просто натиснувши на його назву.

Робота в *monday.com* розпочинається з меню яке дозволяє розпочати створення робочого середовища. Приклад наведено на рис. 4.2.

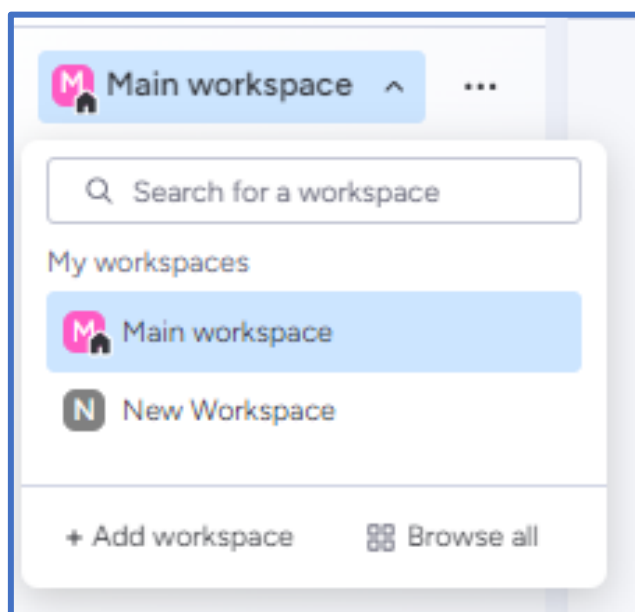


Рис. 4.2 – Початок роботи. Створення робочого середовища

Після вибору проєкту, перейдіть до розділу дашбордів, натиснувши на кнопку «*Dashboards*» у верхньому правому куті робочого простору. Це перенесе вас до меню дашбордів, де ви зможете переглядати та створювати нові візуалізації. Щоб розпочати створення нового дашборду, натисніть кнопку «*Create New Dashboard*». Вам буде запропоновано вибрати «*Boards*», з яких ви хочете отримувати дані. Ви можете обрати один або кілька бордів, просто позначивши їх у списку. Це дозволить дашборду відображати інформацію з різних джерел, надаючи вам більш повну картину прогресу вашої команди (рис. 4.3-4.4).

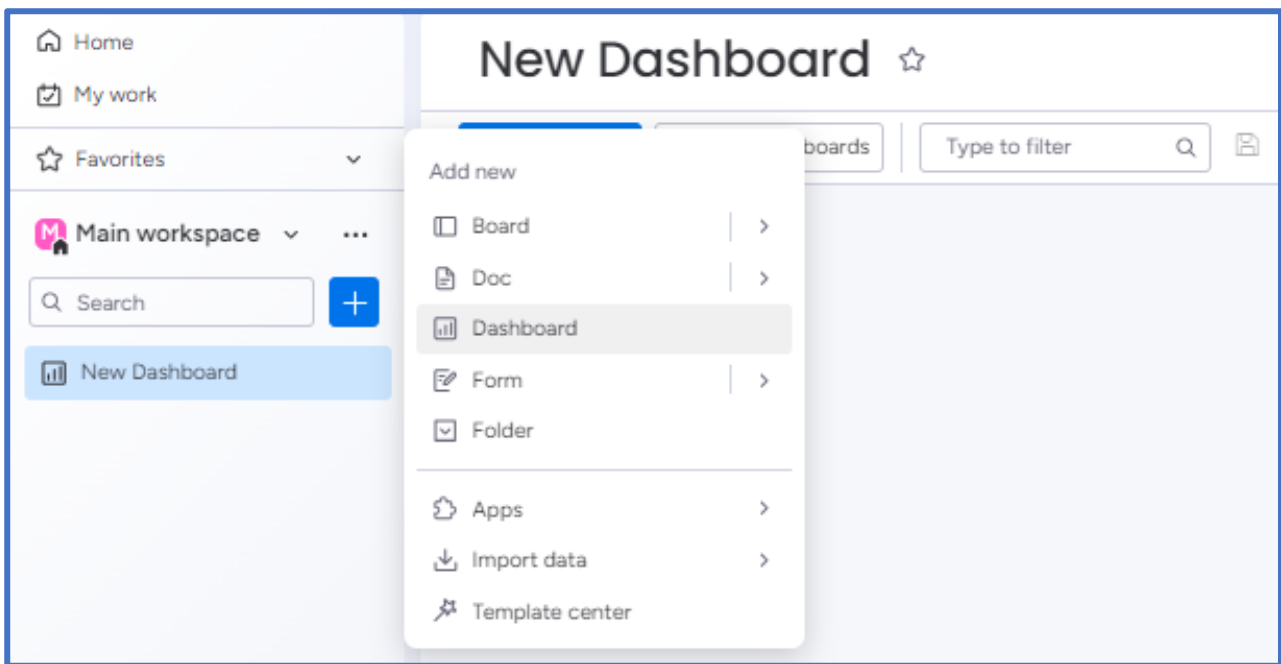


Рис. 4.3 – Створення нового робочого середовища

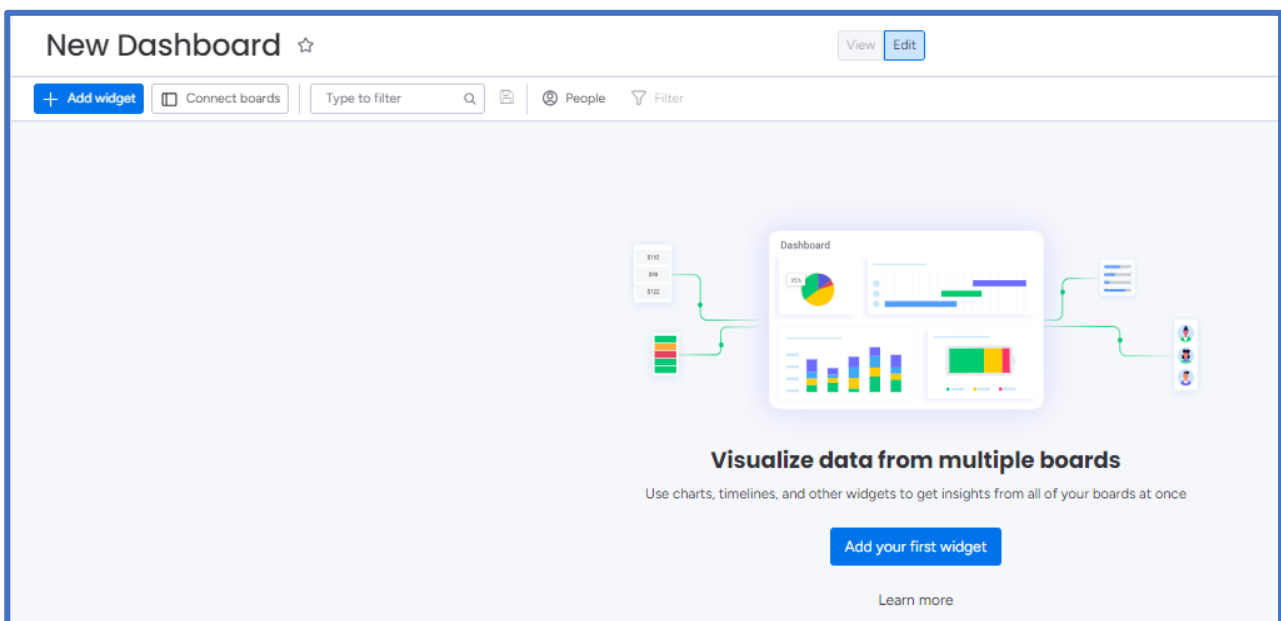


Рис. 4.4 – Новий «Dashboard»

Після створення дашборду ви зможете додати віджети, які дозволяють візуалізувати дані у різних формах, таких як графіки, таблиці або інші елементи. Для цього натисніть «Add Widget» (рис. 4.4) і виберіть тип віджета, який вам потрібен. У меню віджетів ви знайдете кілька варіантів, таких як «Battery Widget» для відображення прогресу у вигляді батареяного індикатора, «Chart Widget» для

створення діаграм (лінійних, стовпчикових тощо) і «*Timeline Widget*» для візуалізації часових рамок завдань. Оберіть тип віджета, що найкраще відповідає вашим потребам у візуалізації даних (рис. 4.5).

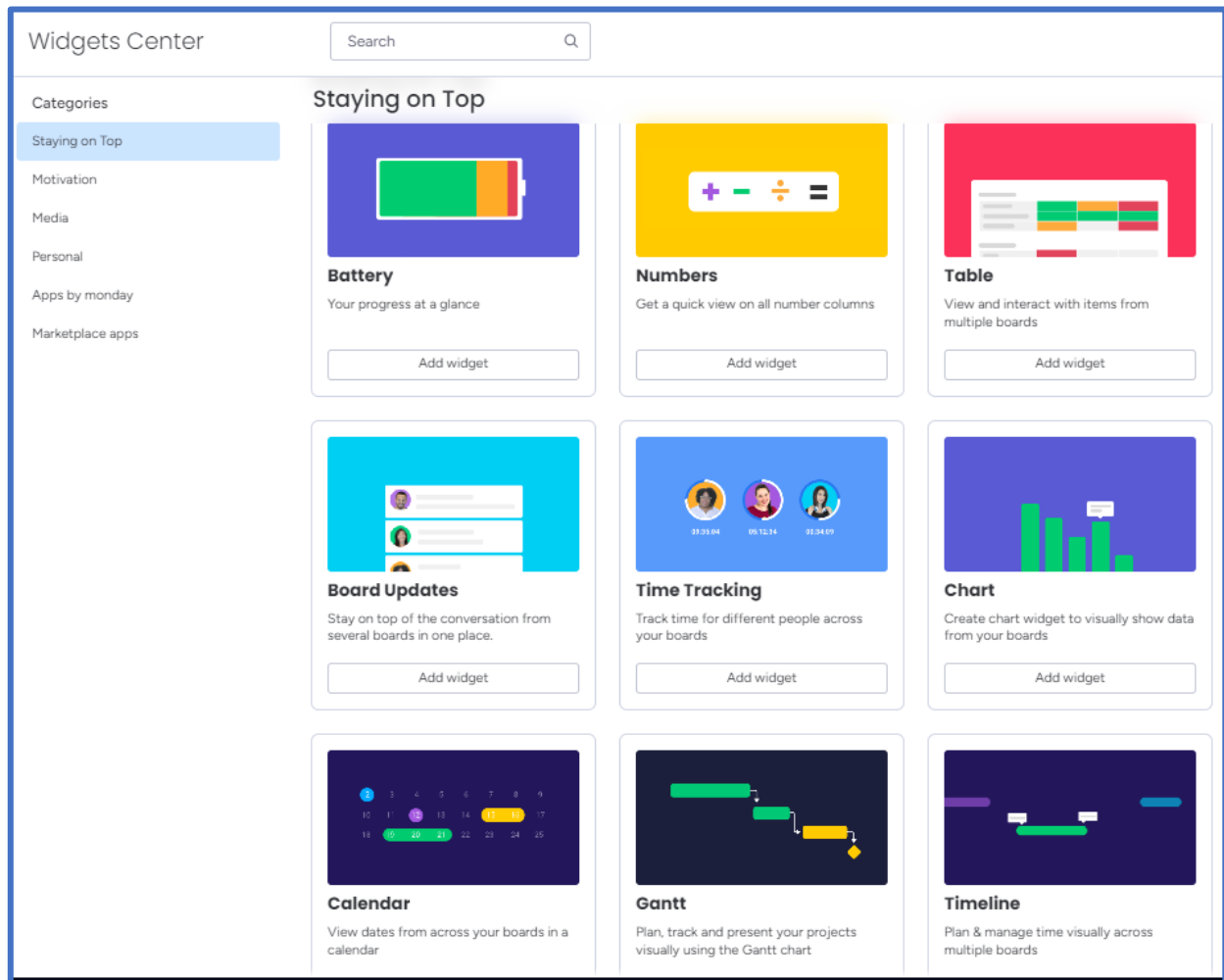


Рис. 4.5 – Робота з віджетами

Після вибору віджета, ви зможете налаштувати його для відображення конкретних даних з вибраних бордів. Це включає вибір полів, які ви хочете відобразити, наприклад, статуси завдань, строки виконання, або відповідальних осіб. Важливо, щоб дані, які ви обираєте, були актуальними і корисними для вашої команди.

Коли ви завершите додавання і налаштування віджетів, пам'ятайте про те, що потрібно натиснути «*Save*» або просто вийти з редактора, оскільки дашборд буде збережено автоматично. Переконайтеся, що всі ваші налаштування збережені і що дашборд відображає необхідні дані.

Додатково, у правій частині екрана ви можете налаштувати фільтри, щоб обмежити відображення даних за певними критеріями, наприклад, за часом або статусом. Це дозволяє вам адаптувати дашборд до конкретних потреб вашої команди.

Щоб забезпечити доступ іншим членам команди до дашборду, ви можете налаштувати права доступу. Натисніть на іконку «Share» (Поділитися) і виберіть, кому з команди можна надати доступ до перегляду або редагування дашборду. Це дасть змогу іншим учасникам команди слідкувати за прогресом і брати участь у управлінні проектом.

Створення візуальних дашбордів у *monday.com* – це інтуїтивно зрозумілий і гнучкий процес, який дозволяє командам візуалізувати дані з різних бордів у формі віджетів. Наприклад, команда маркетологів може створити дашборд, який показує статус усіх рекламних кампаній за допомогою «Chart Widget» (рис. 4.6), відображаючи дані про витрати та ефективність.

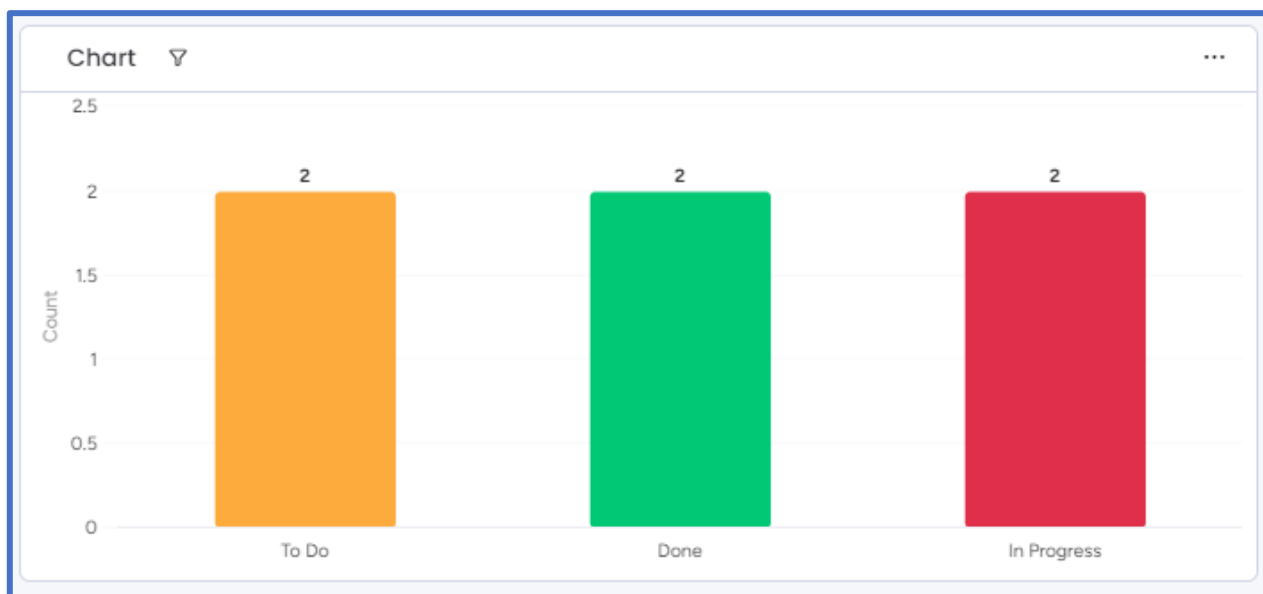


Рис. 4.6 – Chart Widget

У випадку з командою розробників, дашборд може містити «Timeline Widget» (рис. 4.7), щоб візуалізувати етапи розробки програмного забезпечення. Це дозволить всім учасникам бачити, на якому етапі знаходяться різні завдання та коли вони плануються до завершення.

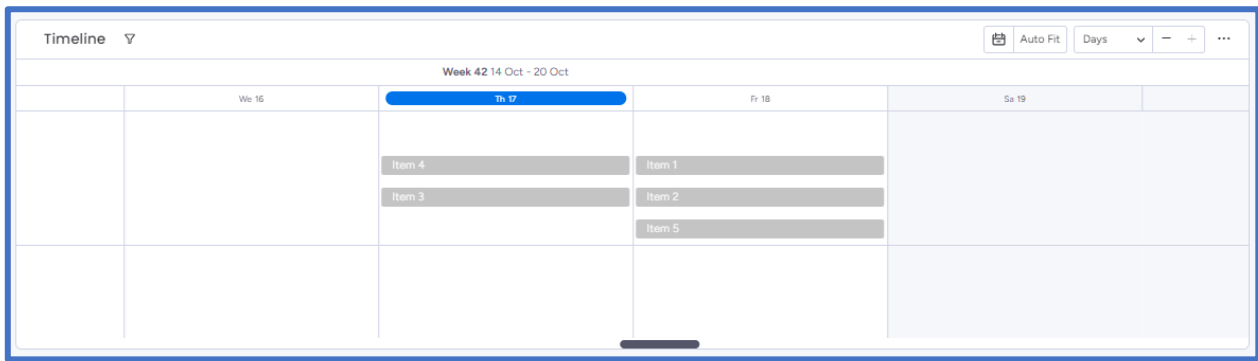


Рис. 4.7 – Timeline Widget

Також, команда з управління проєктами може скористатися «*Battery Widget*» (рис. 4.8) для моніторингу загального прогресу проєкту. Це дозволить швидко оцінити, наскільки близько команда до завершення своїх цілей.

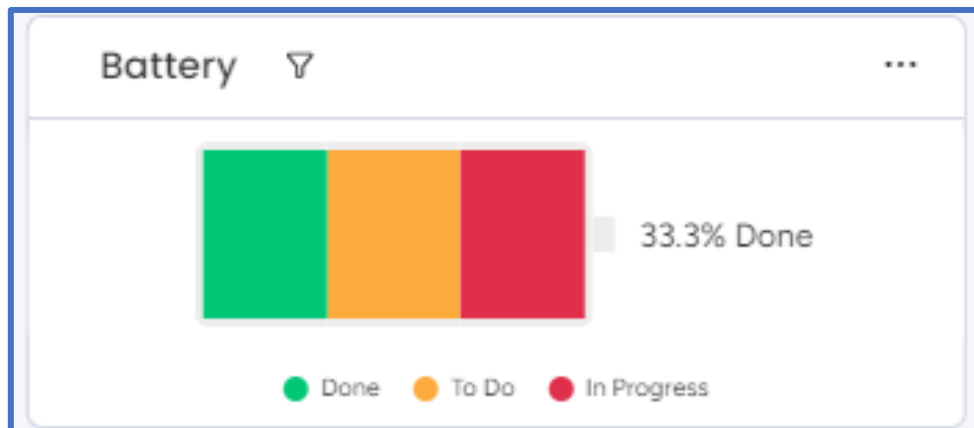


Рис. 4.8 – Battery Widget

В результаті, завдяки цим можливостям, дашборди в **monday.com** стають ефективним інструментом для візуалізації даних, що допомагає командам слідкувати за прогресом, оцінювати ефективність та швидко реагувати на зміни у проєктах.

4.1.2. Відображення прогресу в реальному часі

Функція відображення прогресу в реальному часі в **monday.com** надає можливість відслідковувати прогрес виконання завдань і проєктів у реальному часі, використовуючи різноманітні інструменти візуалізації та індикатори статусу. Для початку вам слід увійти до свого облікового запису на платформі,

після чого оберіть проєкт або борд, для якого ви хочете налаштувати відображення прогресу.

Коли ви перейдете на потрібний борд, важливо впевнитися, що для кожного завдання є колонка «*Status*». Якщо така колонка відсутня, її можна легко додати, щоб мати змогу визначити статус кожного завдання. Установіть статуси для завдань, вибравши з доступних варіантів, таких як «*To Do*», «*In Progress*» або «*Done*» (рис. 4.9). Це дозволить вам наочно відображати етапи виконання завдань. Також з цим форматом можна працювати використовуючи *Kanban-дошку* (рис. 4.10).

The screenshot shows a 'New Board' interface with a 'Main Table' view. It features a toolbar with options like 'New item', 'Search', 'Person', 'Filter', 'Sort', 'Hide', and 'Group by'. Below the toolbar, there are two expandable groups, each with a 'Group Title' header. Each group contains a table with columns for 'Item', 'Person', 'Status', and 'Date'. The first group has items with statuses 'In Progress', 'To Do', 'In Progress', and 'Done'. The second group has items with statuses 'Done' and 'To Do'. A '+ Add item' button is present at the bottom of each group, and a '+ Add new group' button is at the bottom of the entire board.

Item	Person	Status	Date
New Item		In Progress	
Item 1	VZ	To Do	18 Oct
Item 2		In Progress	18 Oct
Item 3		Done	17 Oct

Item	Person	Status	Date
Item 5		Done	18 Oct
Item 4		To Do	17 Oct

Рис. 4.9 – Робота зі статусами завдань

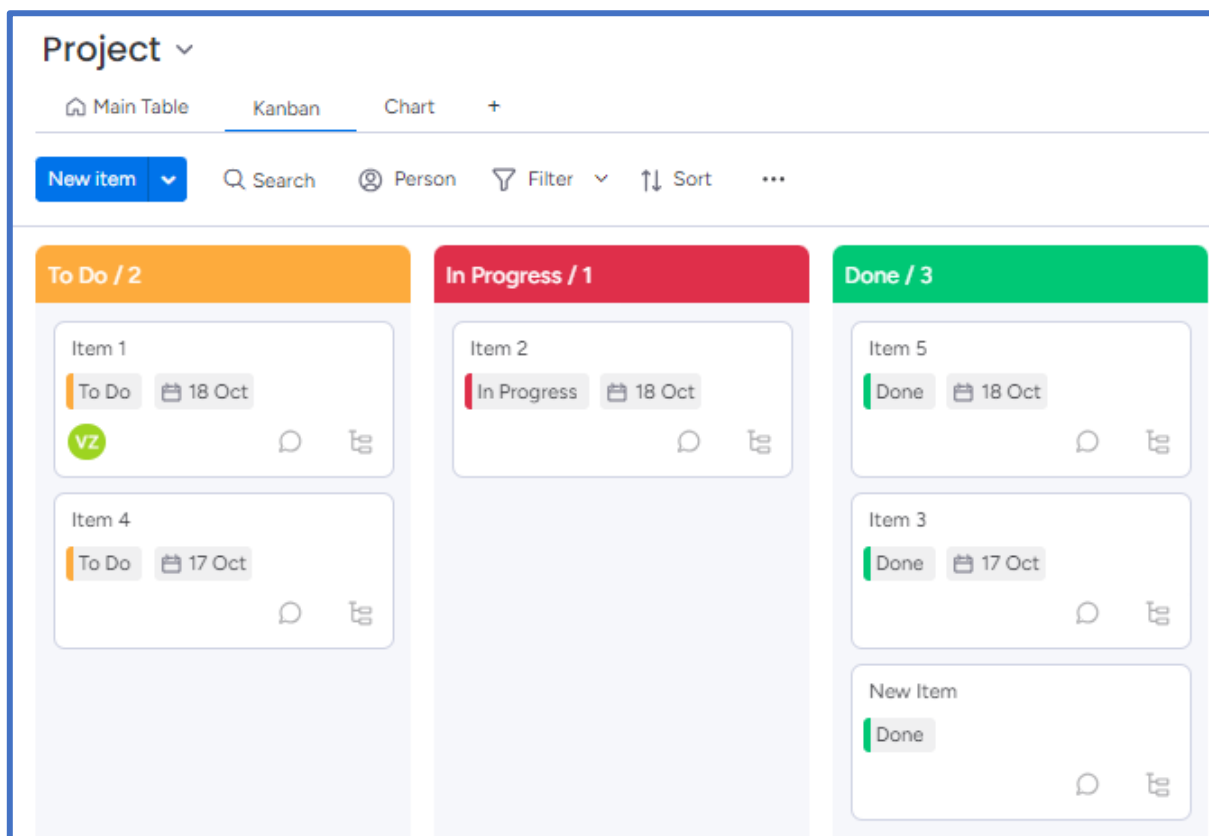


Рис. 4.10 – Використання Kanban-дошки

Для візуалізації загального прогресу проєкту ви можете додати «*Battery Widget*». Для цього натисніть на кнопку «*Dashboards*» у верхній панелі навігації, перейдіть до дашборду або створіть новий. Після цього, у меню налаштувань «*Battery Widget*», вам потрібно обрати колонку зі статусом завдань, за якою буде розраховуватися прогрес. Оберіть відповідний борд і колонку «*Status*», після чого дашборд автоматично відобразить відсотковий прогрес виконання завдань.

Крім цього, можна використовувати «*Chart Widget*» для більш детального аналізу прогресу. Цей віджет дозволяє побачити, скільки завдань перебуває в кожному статусі. Додайте новий віджет на дашборд і під час його налаштування оберіть категорію для візуалізації прогресу, таку як «*Status*», «*People*» або «*Date*». Налаштуйте тип і стиль діаграми, щоб відобразити кількість завдань у кожному статусі.

Для відстеження часових рамок виконання завдань ви також можете додати «*Timeline Widget*». Після натискання на «*Add Widget*» оберіть «*Timeline Widget*», щоб відобразити строки виконання завдань та їхній статус у реальному часі. Це

дозволить вам бачити не тільки стадії виконання, а й терміни, в які завдання повинні бути завершені.

Прогрес завдань оновлюється автоматично після кожної зміни статусу чи часових рамок. Таким чином, коли учасники проекту змінюють статуси завдань, віджет прогресу в реальному часі буде автоматично оновлювати інформацію на дашборді. Крім того, для більшої зручності ви можете налаштувати автоматичні сповіщення про зміни статусу або завершення завдань, натиснувши на іконку автоматизації у верхній панелі борду.

Відображення прогресу в реальному часі в *monday.com* є ефективним способом візуалізувати виконання завдань за допомогою статусів, діаграм та часових віджетів. Наприклад, команда, що займається розробкою, може створити дашборд, на якому відображається прогрес усіх завдань з допомогою «*Battery Widget*», що наочно демонструє, скільки завдань завершено, а скільки ще потребують уваги.

Або команда маркетологів може використати «*Chart Widget*» для відображення кількості завдань у кожному статусі, що допоможе зрозуміти, на якому етапі знаходяться різні кампанії. Команда управління проектами може налаштувати «*Timeline Widget*», щоб контролювати строки виконання важливих завдань, дозволяючи всім учасникам бачити, коли має бути завершено кожне завдання.

Завдяки цим можливостям, *monday.com* стає потужним інструментом для візуалізації даних, що дозволяє командам ефективно відслідковувати прогрес і швидко реагувати на зміни в управлінні проектами.

4.2. Налаштування робочих шаблонів для проєктів

monday.com пропонує потужний інструмент для управління проектами, використовуючи готові робочі шаблони, які можуть значно спростити робочі процеси ваших команд. Для початку вам слід увійти до свого облікового запису на платформі. Після успішного входу, ви потрапите до свого робочого простору, де зможете налаштувати шаблони відповідно до потреб вашого проєкту.

На головній панелі інтерфейсу знайдіть меню «*Templates*» (Шаблони). Клацнувши на цю опцію (рис. 4.11), ви зможете переглянути різноманітні категорії шаблонів, серед яких варто вибрати «*Project Management*» або «*Projects*». Це дозволить вам знайти шаблон, що найкраще відповідає специфіці вашого проекту, будь то розробка програмного забезпечення, впровадження нової системи чи управління командою розробників.

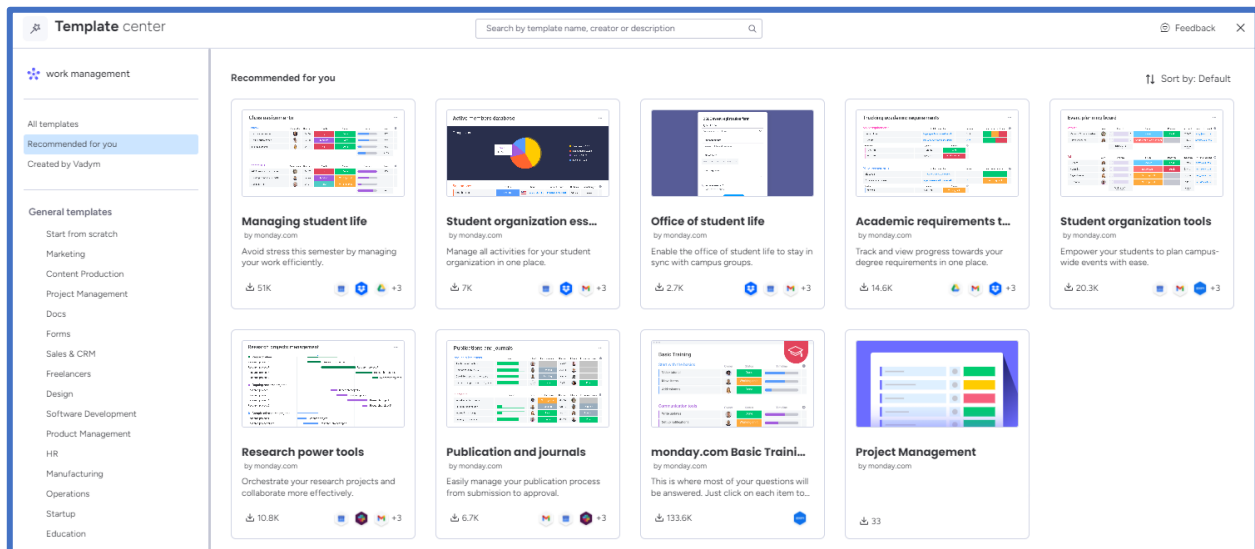


Рис. 4.11 – Робота з шаблонами

Після того, як ви здійсните вибір відповідного шаблону, натисніть на нього, щоб переглянути деталі. Кнопка «*Use Template*» (Використати шаблон) або «*Add to Board*» дозволить вам імплементувати його у ваш робочий простір. Після додавання шаблону, перейдіть до нового борду, що з'явився у вашому робочому просторі, і адаптуйте його під специфіку вашого проекту. Ви можете змінити назву борду, а також налаштувати колонки, завдання та їх статуси.

Важливо також додати чи видалити колонки, які вам потрібні для управління проектом. Наприклад, ви можете створити колонки «*Deadline*», «*Assignee*» (Відповідальний) та «*Priority*». Щоб додати нову колонку, натисніть «*+Add Column*» і виберіть відповідний тип.

Після налаштування колонок настав час створити завдання. Ви можете додавати завдання, використовуючи шаблонні варіанти або створюючи нові. Введіть назву завдання у першому рядку, а потім заповніть відповідні дані у

колонках. Щоб автоматизувати певні процеси, перейдіть до вкладки «*Automate*» (рис. 4.10). Виберіть шаблони автоматизацій, щоб автоматично оновлювати статуси завдань або надсилати сповіщення про зміни.

Окрему увагу слід приділити визначенню ролей і доступу для членів команди, які працюватимуть над проектом. Для цього клацніть на колонку «*Assignee*» для кожного завдання та призначте відповідальних. Коли ви завершите всі зміни, переконайтеся, що шаблон готовий до використання. Ваш борд з шаблоном автоматично зберігається, проте ви можете натиснути кнопку «*Publish*», щоб зробити його доступним для вашої команди.

Наприклад, команда, що займається розробкою програмного забезпечення, може обрати шаблон «*IT Projects*» і адаптувати його під свої потреби. Вони можуть додати колонки для тестування, релізів та помилок, що дозволить відслідковувати кожен етап розробки. Інша команда, що працює над маркетинговою кампанією, може скористатися шаблоном «*Project Management*» для управління завданнями, такими як «Розробка контенту», «Кампанія в соціальних мережах» та «Аналіз результатів».

Крім того, команда з ІТ підтримки може використовувати шаблон для управління запитамі клієнтів, в якому є колонки для пріоритету, статусу запиту та відповідальної особи. Всі ці приклади демонструють, як робочі шаблони в **monday.com** допомагають структурувати процеси та підвищувати ефективність командної роботи. Налаштування робочих шаблонів для проектів стає важливим кроком у оптимізації управлінських процесів, роблячи їх більш структурованими та зручними для команди.

4.3. Управління завданнями та термінами

4.3.1. Створення завдань та контроль за термінами

Управління завданнями в **monday.com** є ключовим елементом, який забезпечує успішне виконання проектів. Для початку, вам потрібно увійти до свого облікового запису на платформі, ввівши свої облікові дані. Після успішного входу виберіть борд «*Name Project*», в якому ви плануєте створити

нові завдання. Ви можете знайти його, клацнувши на назву борду з бокової панелі або на головній сторінці

На борді знайдіть перший пустий рядок або натисніть на кнопку «*New Item*» (Додати елемент або завдання). У першому стовпці, який за замовчуванням може називатися «*Task*» або «*Item*» (рис. 4.12), введіть назву нового завдання. Важливо також визначити термін виконання, тому знайдіть колонку «*Due Date*» (Крайній) або «*Deadline*». Клацнувши на комірку у колонці «*Due Date*», ви зможете вибрати дату з календаря, що дозволить вашій команді чітко дотримуватись встановлених термінів.

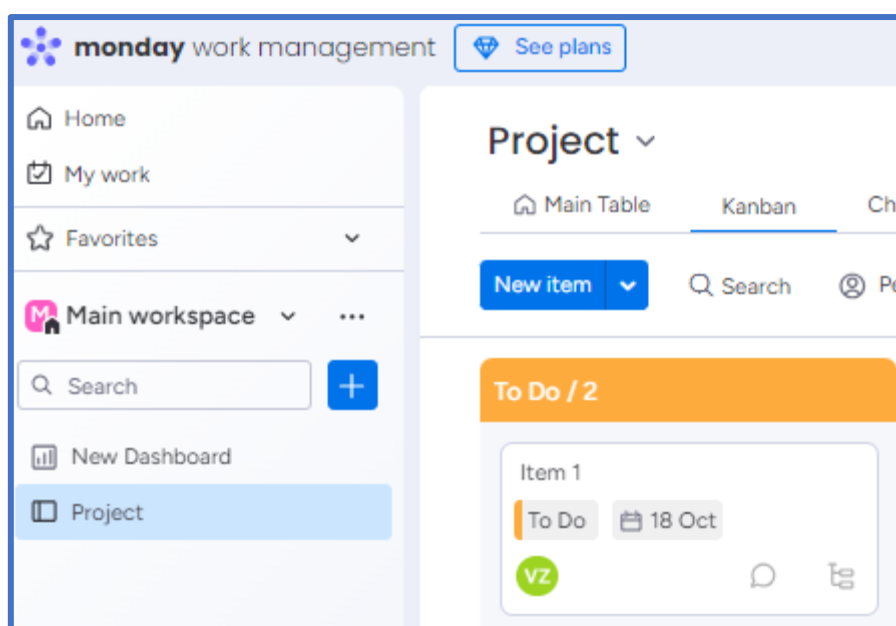


Рис. 4.12 – Додавання завдань

Наступним кроком буде управління пріоритетами завдань. Якщо у вас ще немає колонки для пріоритетів, натисніть «*+Add Column*» і виберіть тип колонки «*Status*» або «*Dropdown*» (Випадний список). У колонці «*Priority*» визначте рівень пріоритету для кожного завдання, вибравши, наприклад, «*High*», «*Medium*» або «*Low*». Це допоможе вашій команді зосередитися на найбільш критичних завданнях у проєкті.

Якщо вам потрібно, ви можете додати додаткові деталі до завдання, такі як опис, відповідальний виконавець та інші колонки. Просто клацніть на відповідні комірки, щоб додати текст або вибрати учасника команди. Під час виконання

завдання важливо оновлювати статус у колонці «*Status*», щоб мати можливість здійснювати моніторинг прогресу. Натисніть на комірку у колонці «*Status*» і виберіть відповідний стан, наприклад, «*In Progress*» або «*Done*».

Для більш ефективного контролю над термінами та пріоритетами, ви можете налаштувати автоматизацію для отримання сповіщень про наближення дедлайнів. Для цього перейдіть до вкладки «*Automation*» у верхній частині борду. Це дозволить вашій команді не пропустити важливі терміни.

Нарешті, для зручного контролю за виконанням завдань використовуйте дашборд. Натисніть на «*Dashboards*» (рис. 4.13) у верхній панелі навігації, щоб створити новий дашборд або перейти до вже існуючого. Це дозволить вам візуально відстежувати прогрес проекту та управляти завданнями більш ефективно.

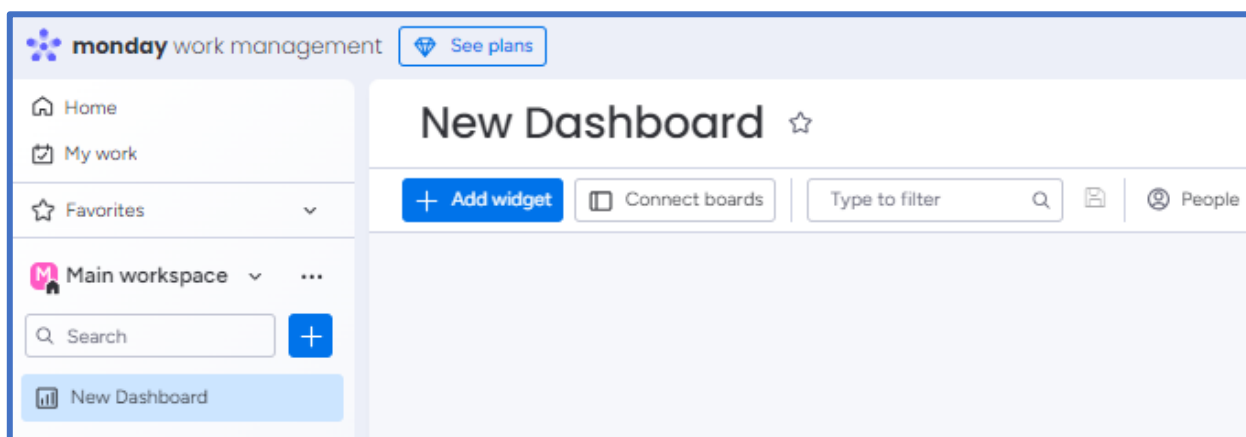


Рис. 4.13 – Робота з Dashboards

Наприклад, команда, що займається розробкою програмного забезпечення, може створити завдання на кшталт «Розробити новий функціонал» з терміном виконання «15 жовтня», пріоритетом «*High*» та відповідальним виконавцем, призначеним на цю задачу. Інша команда, що займається маркетинговими кампаніями, може додати завдання «Запустити рекламну кампанію» з дедлайном на «30 вересня» і середнім пріоритетом. Крім того, команда технічної підтримки може створити завдання на кшталт «Вирішити проблему з клієнтом» з терміном «5 жовтня» та вказати високий пріоритет, щоб уся команда знала, що ця задача є критично важливою.

Таким чином, створення завдань, визначення дедлайнів і управління пріоритетами в *monday.com* дозволяє ефективно контролювати робочі процеси в рамках проєктів, що сприяє успішному завершенню всіх запланованих задач.

4.3.2. Відстеження виконання та результатів

Відстеження виконання завдань і результатів у *monday.com* є важливим етапом для забезпечення успішного управління проєктами. Для початку вам потрібно увійти до свого облікового запису, ввівши свої дані. Після успішного входу ви зможете вибрати борд «*Name Project*», за яким плануєте здійснювати контроль за виконанням завдань. Клацнувши на назву борду з бокової панелі або на головній сторінці, ви перейдете до списку завдань, що є важливою частиною вашої роботи.

На борді уважно розгляньте список завдань, звертаючи особливу увагу на стовпці «*Status*», «*Due Date*» (*Крайній термін*) та інші, щоб оцінити прогрес виконання. Якщо вам потрібно, ви можете відфільтрувати завдання за статусом або пріоритетом. Просто клацніть на фільтр, який розташований у верхньому правому куті, і виберіть відповідні критерії для фільтрації, що дозволить вам швидше знаходити найважливіші завдання.

Для візуалізації прогресу і кращого розуміння результатів ви можете створити або перейти до дашборду «*Dashboards*». Натисніть на «*Dashboards*» у верхній панелі навігації. Тут ви можете додати віджети, які допоможуть вам візуалізувати дані. Клацнувши на «*Add Widget*», ви зможете вибрати відповідні віджети, такі як «*Progress Tracking*» (*Відстеження прогресу*), «*Pie Chart*» (*Кругова діаграма*), або «*Bar Chart*» (*Стовпчаста діаграма*), щоб зручно бачити стан виконання завдань.

Наступним кроком є налаштування віджетів відповідно до потреб вашого проєкту. Виберіть борд, з якого потрібно отримувати дані, та параметри відображення, щоб дашборд був максимально інформативним. Не забувайте здійснювати моніторинг коментарів та оновлень для кожного завдання, щоб залишатися в курсі прогресу. Для цього просто клацніть на завдання, щоб

відкрити його деталі, і перегляньте вкладку «*Updates*» (рис. 4.14).

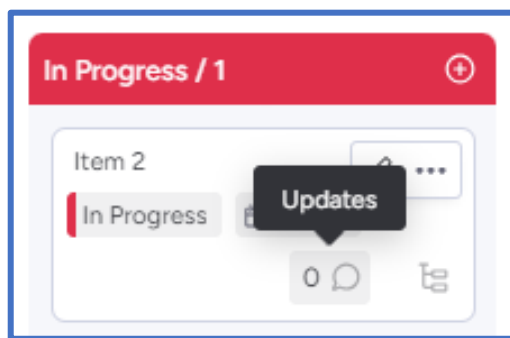


Рис. 4.14 – Робота з функцією «*Updates*»

Оцінка виконання завдань здійснюється на основі їхнього статусу. Перегляньте стовпець «*Status*» для кожного завдання на борді, щоб визначити результати виконання. Чи виконано завдання? Чи ще триває його реалізація? Це дозволить вам оперативно реагувати на будь-які затримки або проблеми.

Для глибшого аналізу результатів створіть звіти, використовуючи дані завдань. Натисніть на «*Reports*» у верхній панелі навігації для створення звітів, що допоможе вам проаналізувати виконання завдань у рамках проєкту.

Наприклад, команда, яка займається розробкою нового програмного забезпечення, може мати завдання «Завершити тестування версії 1.0», яке буде мати статус «*In Progress*» з дедлайном «18 жовтня». Інша команда, що відповідає за маркетингові кампанії, може слідкувати за завданням «Запустити рекламну кампанію», яке має статус «*Done*» і було завершено на тиждень раніше запланованого терміну. Команда технічної підтримки може контролювати прогрес завдання «Виправити помилку у системі», яке ще знаходиться на стадії «*In Progress*», що вказує на активну роботу над ним. Це дозволяє усім учасникам проєкту бути в курсі виконання завдань і вчасно реагувати на можливі затримки або проблеми.

Таким чином, використання цього алгоритму для відстеження виконання завдань і результатів у *monday.com* забезпечує можливість ефективно контролювати прогрес, аналізувати результати та приймати обґрунтовані рішення для покращення процесу виконання завдань.

4.4. Спільна робота та комунікація в команді

У сучасному світі управління проєктами спільна робота та комунікація в команді стали ключовими чинниками для досягнення успішних результатів. Платформа **monday.com** пропонує потужні інструменти, що сприяють ефективному спілкуванню та співпраці, що, у свою чергу, підвищує продуктивність і забезпечує злагоджену роботу команд.

Основні функції для спільної роботи наступні:

1. Оперативна комунікація. **monday.com** дозволяє команді спілкуватися в реальному часі. Кожне завдання має вкладку «*Updates*» (рис. 4.15), де учасники можуть залишати коментарі, ділитися думками та обговорювати деталі проєкту. Це усуває необхідність в електронній пошті і надає всім учасникам доступ до актуальної інформації.

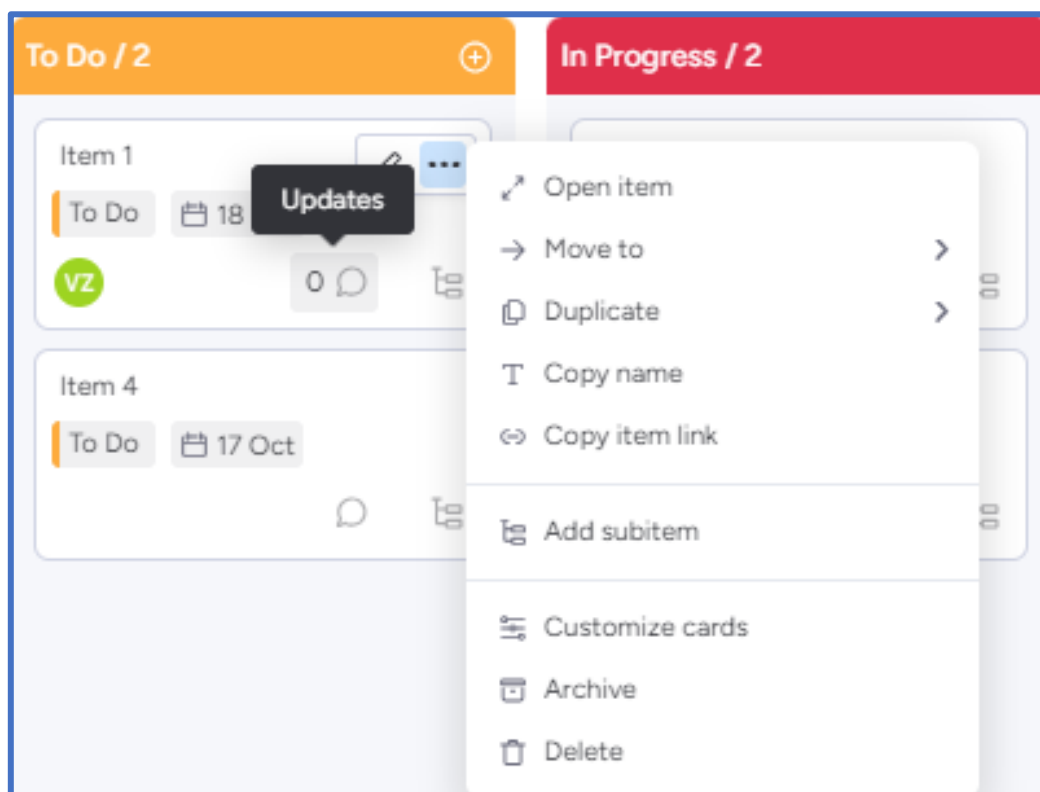


Рис. 4.15 – Розширений функціонал можливостей роботи з функцією «*Updates*»

2. Призначення завдань. Кожен учасник команди може бути призначений на конкретне завдання, що забезпечує чіткість відповідальності. Керівники проєктів

можуть легко відслідковувати, хто за що відповідає, що сприяє підвищенню прозорості в роботі.

3. Інтеграція з іншими сервісами. Платформа підтримує інтеграцію з безліччю сторонніх додатків і сервісів, таких як *Google Drive*, *Zoom*, *Miro*, *Outlook* та інші (рис. 4.16). Це дозволяє команді зручно працювати з різними інструментами, зберігаючи всю інформацію в одному місці.

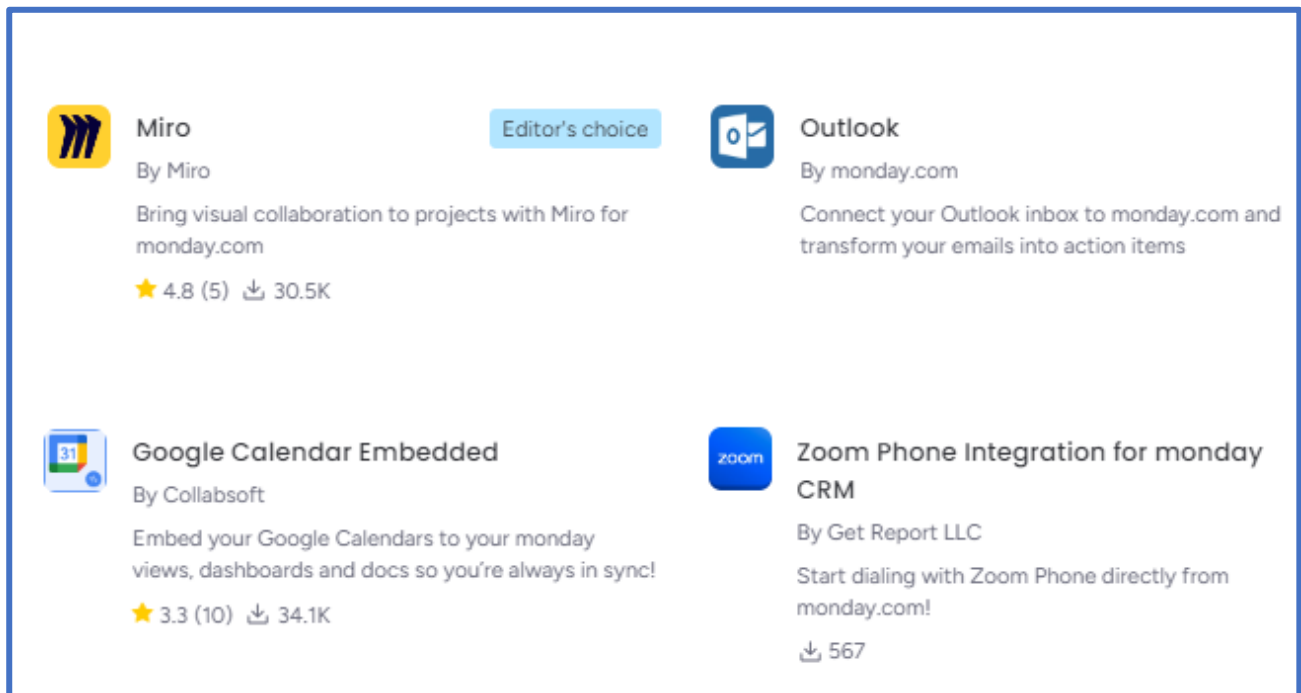


Рис. 4.16 – Приклади інтеграції та можливостей роботи зі сторонніми додатками та сервісами

4. Візуалізація прогресу. *monday.com* пропонує різноманітні візуалізації, такі як дашборди, графіки та таблиці, які наочно демонструють прогрес виконання завдань. Це допомагає командам швидко оцінити ситуацію і вчасно коригувати свої дії.

Переваги спільної роботи

Однією з основних переваг використання *monday.com* для спільної роботи є можливість залучення всіх членів команди до процесу прийняття рішень. Завдяки доступу до актуальної інформації та можливості обговорення важливих

питань у режимі реального часу, команди можуть швидше реагувати на зміни та адаптуватися до нових умов.

Крім того, відсутність бар'єрів у спілкуванні підвищує рівень залученості працівників, що, в свою чергу, позитивно впливає на якість роботи та задоволеність команди.

Спільна робота та комунікація в команді в *monday.com* є невід'ємною частиною управління проектами. Інструменти для оперативного спілкування, призначення завдань, інтеграції з іншими сервісами та візуалізації прогресу створюють середовище, яке сприяє продуктивності, прозорості та злагодженості в роботі команди. В результаті, платформа стає незамінним помічником для сучасних професіоналів у сфері управління проектами.

4.4.1. Використання коментарів та тегів

У *monday.com* коментарі та теги є надзвичайно важливими інструментами, які значно покращують комунікацію між членами команди, відстежують прогрес завдань і організують роботу в цілому. Давайте розглянемо, як ефективно використовувати ці функції, починаючи з входу до системи.

Спочатку вам потрібно увійти до свого облікового запису, ввівши свої дані для входу. Після успішного входу виберіть борд (*Project*), в якому плануєте використовувати коментарі та теги. Клацнувши на назву борду, ви потрапите до його детальної інформації, де зможете активно взаємодіяти зі своєю командою.

Знайдіть завдання, до якого ви хочете додати коментар. Клацнувши на завдання, відкриється його детальний опис. У полі для коментаря можна ввести своє повідомлення (рис. 4.17). Наприклад, ви можете написати: «Звертаю вашу увагу на зміни в специфікаціях. Будь ласка, перевірте!» Це повідомлення сигналізує про важливість інформації і стимулює команду звернути увагу на деталі.



Рис. 4.17 – Можливості роботи з коментарями

Для підвищення інформативності коментаря ви можете додати теги. Використовуйте символ «@» для позначення учасників команди або символ «#» для категорій. Наприклад, якщо ви хочете залучити до обговорення Вадима, напишіть: «@Vadym, чи можеш переглянути новий прототип і дати свої коментарі?». У такий спосіб ви не тільки звертаєтеся до конкретної особи, а й закликаєте її до дії, що допомагає зберегти чіткість у комунікації.

Коли вам потрібно відповісти на вже існуючий коментар, просто натисніть кнопку «Reply» (Відповісти) під коментарем. Це дозволяє підтримувати безперервний діалог. Наприклад, якщо ваша команда обговорює терміни виконання завдання, ви можете відповісти: «Дякую за уточнення, але чи можемо ми перенести термін до кінця тижня? Це дозволить всім учасникам завершити свої частини проєкту». Це забезпечує можливість для конструктивного обговорення і уточнення деталей.

Для перегляду всіх коментарів на борді перейдіть до вкладки «Updates» для конкретного завдання. Тут ви зможете ознайомитися з усіма обговореннями, що стосуються цього завдання, і бачити, як прогресує його

Щоб швидко знайти коментарі з певними тегами, скористайтеся функцією фільтрації. У верхній частині коментарів клацніть на поле фільтру та введіть тег. Наприклад, ви можете ввести «#urgent», щоб знайти всі термінові запити, що можуть потребувати негайної уваги.

Не забувайте налаштувати сповіщення про нові коментарі, щоб бути в курсі всіх оновлень. Для цього перейдіть до налаштувань облікового запису (*Account Settings*), де можна вибрати відповідні опції сповіщень (рис. 4.18).

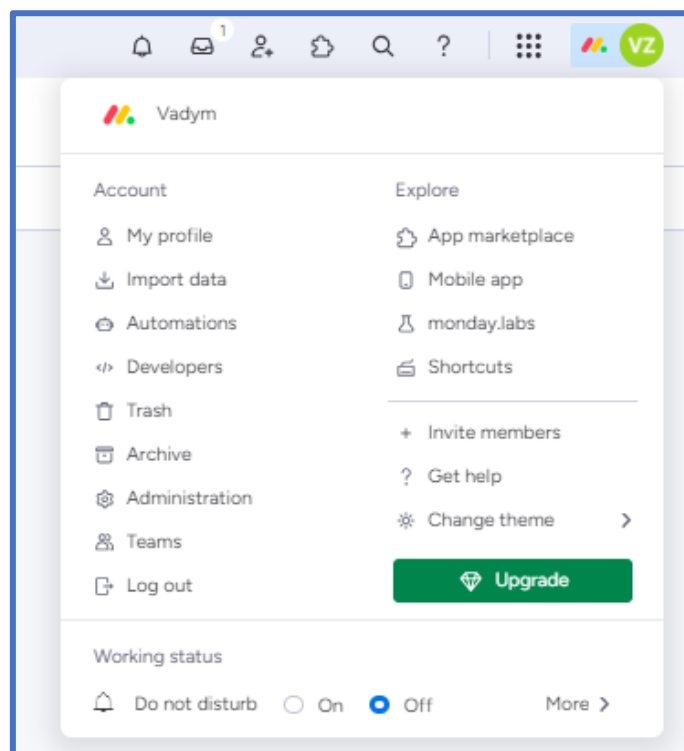


Рис. 4.18 – Налаштування акаунту

Наприклад, користувачу можна обрати отримувати сповіщення на електронну пошту або в додатку, щоб нічого не пропустити.

У разі необхідності редагування або видалення свого коментаря, наведіть курсор на коментар, клацніть на значок трьох крапок «⋮» поруч із ним і виберіть «Edit» або «Delete». Наприклад, якщо ви виявили помилку в коментарі, що стосується термінів, ви можете швидко його відредагувати, написавши: «Вибачте, термін на виконання проєкту – 15 число поточного місяця, а не 20».

Таким чином, завдяки використанню коментарів і тегів у *monday.com*, ви зможете суттєво полегшити комунікацію та організувати роботу в команді. **Наприклад**, у проєкті, що стосується розробки нового програмного забезпечення, коментарі можуть виглядати так: «Нам потрібно вирішити питання з інтеграцією API. @Anastasiia, чи можеш підготувати технічне завдання до середи?» або «Можливо, нам слід провести ще один тест перед запуском. #QA, я чекаю ваших коментарів!». Використання таких коментарів сприяє більш продуктивній взаємодії в команді, дозволяє всім бути в курсі всіх важливих оновлень і приймати обґрунтовані рішення на всіх етапах проєкту.

4.4.2. Функція обговорень у рамках завдань

Функція обговорень у *monday.com* є важливим інструментом для ефективної комунікації та обміну думками між членами команди безпосередньо в рамках завдань. Це сприяє покращенню координації та ясності в процесі виконання проєктів, що в свою чергу підвищує продуктивність та забезпечує досягнення спільних цілей.

Спочатку, щоб скористатися функцією обговорень, увійдіть до свого облікового запису на *monday.com*, ввівши свої дані для доступу. Після цього оберіть борд «*Project*», в якому плануєте використовувати обговорення. Клацнувши на назву борду, ви зможете перейти до його детальної інформації, де і відбудеться подальша взаємодія.

Знайдіть завдання, до якого ви хочете додати обговорення, і клацніть на нього, щоб відкрити його деталі. Після цього перейдіть до вкладки «*Updates*», де ви зможете додавати нові коментарі. У полі для коментаря введіть своє повідомлення, наприклад: «Маємо терміновий проєкт на розгляд. Які пропозиції щодо плану дій?».

Щоб залучити інших учасників до обговорення, використовуйте функцію тегування. Напишіть «@» та виберіть учасника з випадного списку, наприклад: «@Anastasiia, я знаю, що ти маєш досвід у цьому. Можеш поділитися своїми думками?» Це не тільки забезпечить вашу команду важливою інформацією, але й заохотить інших до активної участі.

Коли ваш коментар отримає відповідь, щоб долучитися до діалогу, натисніть кнопку «*Reply*» під коментарем, і введіть свою відповідь у відкритому полі. Наприклад, «Дякую за твою думку, @ Anastasiia! Я вважаю, що можемо спробувати це зробити до кінця тижня». Це створює конструктивну атмосферу, де кожен має можливість висловити свою думку.

Для перегляду всіх коментарів і відповідей залишайтеся на вкладці «*Updates*». Прогорніть вниз, щоб ознайомитися з усіма обговореннями, які стосуються даного завдання. У разі необхідності редагування або видалення свого коментаря, наведіть курсор на коментар і натисніть на значок трьох крапок

«:» для доступу до меню, де ви зможете обрати «*Edit*» або «*Delete*».

Не менш важливо налаштувати сповіщення про нові обговорення, щоб бути в курсі всіх оновлень. Перейдіть до налаштувань облікового запису «*Account Settings*» та виберіть відповідні опції сповіщень. Це дозволить вам отримувати оновлення в режимі реального часу і не пропустити важливі коментарі.

Функція обговорень у рамках завдань у *monday.com* є потужним інструментом для підтримки ефективної комунікації в команді. Наприклад, у проекті, пов'язаному з запуском нового продукту, ваш коментар може звучати так: «Завтра ми плануємо провести нараду. @Vadym, чи можеш підготувати звіт про стан справ?». А у відповідь ви можете побачити: «Звісно, @Anastasiia, готовий представити результати, дякую!». Таке обговорення підкреслює взаємодію та підтримує процес прийняття рішень, що є критично важливим для успішного завершення проєкту.

4.5. Автоматизація процесів в *monday.com*

Автоматизація процесів у *monday.com* є важливим інструментом, що дозволяє зекономити час, зменшити кількість рутинних завдань і підвищити продуктивність команди. За допомогою автоматизації ви можете налаштувати дії, які будуть виконуватися автоматично, коли відбудуться певні події, що допоможе вашій команді зосередитися на важливих аспектах проєкту.

Спершу вам потрібно увійти до свого облікового запису на *monday.com*, ввівши свої облікові дані. Після успішного входу виберіть борд «*Project*», в якому ви плануєте налаштувати автоматизацію. Клацнувши на назву борду, ви потрапите на його головну сторінку, де і буде розпочато налаштування.

На верхній панелі борду знайдіть кнопку «*Automate*» (рис. 4.19), яка відкриє налаштування автоматизації «*Automation center*» (рис. 4.20). Тут ви зможете обрати один із наявних шаблонів або створити нову автоматизацію. Ви можете переглядати категорії шаблонів або скористатися функцією пошуку, щоб знайти найбільш підходящий варіант.

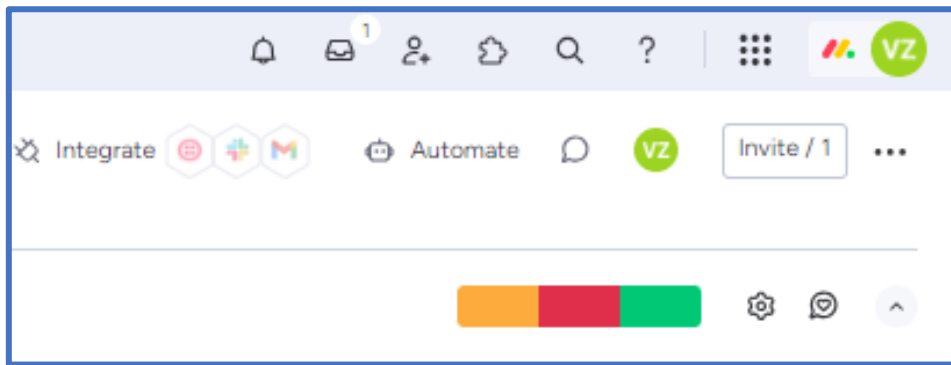


Рис. 4.19 – Розташування кнопки «Automate»

При виборі шаблону автоматизації ви зможете задати тригери (умови), за яких повинна діяти автоматизація, та дії, які вона повинна виконати. **Наприклад**, ви можете обрати тригер «*When a status changes*» (Коли статус змінюється), а далі визначити, що саме потрібно зробити після цього. Це може бути автоматичне сповіщення команди про зміни в статусі завдання або оновлення термінів виконання.

Наступним кроком буде визначення конкретних умов для тригера. **Наприклад**, якщо ви виберете тригер «*When a status changes*», вам потрібно буде вказати, який саме статус має змінитися («*In Progress*» або «*Done*»). Введіть або виберіть значення з меню, щоб налаштувати автоматизацію відповідно до потреб вашої команди.

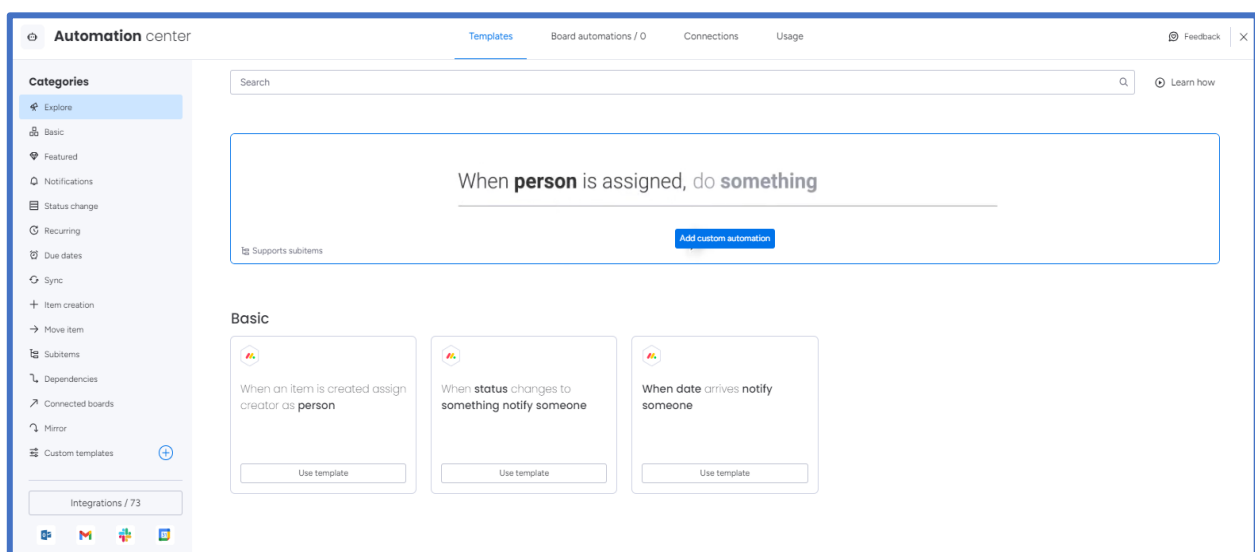


Рис. 4.20 – Робота з центром автоматизації процесів

Після налаштування автоматизації обов'язково протестуйте її, щоб переконатися, що вона працює як слід. Змініть статус завдання або виконайте дію, яка повинна активувати автоматизацію, і перевірте, чи відбувається запланована дія.

Після успішного тестування не забудьте зберегти автоматизацію, натиснувши відповідну кнопку для збереження змін. Щоб переглянути всі активні автоматизації, поверніться до меню «*Automation center*». Тут ви побачите список усіх налаштованих автоматизацій і зможете за потреби редагувати або видаляти їх.

Наприклад, ви можете налаштувати автоматизацію, яка автоматично надсилатиме нагадування команді про терміни виконання завдань. Встановивши тригер «*When due date arrives*» (*Коли настає термін*), ви зможете автоматично надсилати електронні листи або сповіщення у ***monday.com***.

Ще одним **прикладом** може бути автоматизація, яка оновлює статус завдання на «***Completed***» автоматично, коли всі підзавдання виконані. Це дозволить вашій команді швидше отримувати зворотний зв'язок про виконання проекту без необхідності вручну перевіряти кожне завдання.

Крім того, ви можете налаштувати автоматизацію для створення нових завдань на основі певних умов. **Наприклад**, коли новий клієнт додається до бази даних, автоматично створюється завдання для підготовки первинного звіту. Це допоможе вашій команді бути завжди в курсі нових завдань і не пропускати важливі етапи в роботі.

Таким чином, автоматизація процесів у ***monday.com*** дозволяє зменшити рутинні завдання і підвищити ефективність роботи команди. Використовуючи цей алгоритм, ви зможете налаштувати автоматизації, які адаптуються до ваших потреб і сприяють досягненню успішних результатів у вашому проекті.

Завдання для самостійного опрацювання

Завдання 1. Створення проєкту та управління завданнями у monday.com

Хід виконання:

Етап 1. Створення нової дошки проєкту.

Крок 1. Увійдіть у monday.com та створіть нову дошку з назвою «Розробка інформаційної системи...».

Крок 2. Оберіть шаблон «Управління проєктом» або створіть власну структуру з відповідними колонками (наприклад): «Завдання», «Відповідальний», «Статус», «Старт», «Фініш», «Пріоритет», «Терміни».

Етап 2. Додавання завдань.

Крок 1. Додайте основні етапи проєкту (наприклад):

- «Планування».
- «Дизайн».
- «Розробка».
- «Тестування».
- «Реліз».

Крок 2. Для завдання «Розробка» додайте підзадачі, використовуючи функцію підзавдань:

- «Front-end розробка».
- «Back-end розробка».
- «Інтеграція з базою даних».

Крок 3. У колонках «Старт» і «Фініш» встановіть дати початку та закінчення для кожного завдання.

Етап 3. Встановлення залежностей.

Крок 1. Налаштуйте залежності між завданнями, щоб одне завдання не могло розпочатися, поки не буде завершено інше (наприклад):

- «Розробка» може початися тільки після завершення «Дизайну».
- «Тестування» розпочинається після завершення «Розробки».

Крок 2. Переконайтеся, що залежності правильно відображаються у колонці

«Залежності».

Етап 4. Призначення відповідальних.

Крок 1. У колонці «Відповідальний» призначте членів команди для кожного завдання:

- «Планування» – Ім'я 1.
- «Дизайн» – Ім'я 2.
- «Front-end розробка» – Ім'я 3.
- «Back-end розробка» – Ім'я 4.
- «Тестування» – Ім'я 5.

Крок 2. Використайте автоматичні нагадування, щоб інформувати відповідальних про зміни у завданнях.

Етап 5. Сформувати короткий висновок за результатами виконання завдання 1.

Завдання 2. Відстеження прогресу та використання автоматизацій у monday.com

Хід виконання:

Етап 1. Оновлення статусу завдань.

Крок 1. Переконайтеся, що кожен член команди оновив статус своїх завдань у колонці «Статус» (наприклад):

- «Планування» завершено.
- «Дизайн» виконується (50%).
- «Back-end розробка» виконується (60%).

Крок 2. Відкоригуйте статуси для тих завдань, які зазнали змін у процесі роботи.

Етап 2. Використання автоматизацій для процесів.

Крок 1. Налаштуйте автоматизацію, яка буде змінювати статус завдання на «Виконано», коли його підзадачі будуть завершені.

Крок 2. Створіть автоматизацію для відправки сповіщень членам команди, якщо завдання стає простроченим (перевищує встановлений дедлайн).

Етап 3. Використання дашборду для моніторингу проекту.

Крок 1. Створіть дашборд, щоб відстежувати прогрес проекту:

- додайте графік, що відобразить статус усіх завдань за категоріями (завершені, в процесі, прострочені).
- налаштуйте показники для відстеження ключових етапів проекту.

Крок 2. Переконайтеся, що всі дані на дашборді автоматично оновлюються після змін у статусах завдань.

Етап 4. Аналіз ризиків та створення звіту.

Крок 1. Оцініть основні ризики на основі відображення прострочених та затриманих завдань.

Крок 2. Згенеруйте звіт, що включає аналіз прогресу, статусів завдань і ризиків для проекту.

Крок 3. Експортуйте звіт у PDF або Excel для обговорення на наступній командній нараді.

Етап 5. Сформулювати короткий висновок за результатами виконання завдання.

Контрольні запитання до розділу

1. Що таке monday.com і як його використовують для управління проектами?
2. Які основні функції платформи monday.com для управління?
3. Як створювати та налаштовувати нові проекти в monday.com?
4. Що таке «робочі простори» у monday.com і як вони допомагають організувати роботу над проектами?
5. Як у monday.com налаштовувати та використовувати проектні дошки (boards)?
6. Як Monday.com підтримує методології управління проектами, такі як Agile або Scrum?
7. Як відстежувати виконання завдань та підзадач у monday.com?
8. Як у monday.com налаштовується автоматизація робочих процесів?
9. Як використовувати monday.com для управління командою та розподілу завдань?
10. Які типи віджетів доступні на дашбордах у monday.com для візуалізації проектів?
11. Як проводиться пріоритизація завдань у monday.com?
12. Як у monday.com відбувається інтеграція з іншими інструментами (Slack, Zoom тощо)?
13. Які можливості для співпраці та комунікації команди пропонує monday.com?
14. Як відстежувати прогрес у виконанні проекту за допомогою таймлайну?
15. Як налаштовувати сповіщення та нагадування в monday.com для команди?
16. Як проводити аналіз продуктивності проекту за допомогою звітів у monday.com?
17. Що таке «статуси завдань» у monday.com і як їх використовувати для контролю виконання?
18. Як використовувати monday.com для керування ресурсами та планування часу?
19. Які переваги дає використання шаблонів для управління проектами?
20. Як платформа monday.com забезпечує безпеку даних і контроль доступу до інформації?

РОЗДІЛ 5. УПРАВЛІННЯ ПРОЄКТАМИ ЗА ДОПОМОГОЮ ВЕБ-ПЛАТФОРМИ SMARTSHEET

5.1. Основи Smartsheet для управління проєктами

Smartsheet – це потужний інструмент для управління проєктами, який поєднує в собі функціональність електронних таблиць із можливостями колективної роботи та управління завданнями. З моменту свого заснування, *Smartsheet* завоював популярність серед проєктних команд завдяки своїй простоті у використанні, гнучкості та можливостям автоматизації. Нижче розглянемо особливості даної веб-платформи для управління проєктами.

Гнучка структура. *Smartsheet* дозволяє користувачам створювати проєкти в вигляді таблиць, що надає зручний і зрозумілий інтерфейс для управління завданнями, ресурсами та термінами. Завдяки цій структурі, користувачі можуть легко налаштовувати таблиці відповідно до своїх потреб, додаючи кастомізовані поля, формули, та умови форматування.

Координація команди. Однією з основних переваг *Smartsheet* є можливість спільної роботи в реальному часі. Користувачі можуть коментувати завдання, обмінюватися файлами та отримувати сповіщення про зміни в проєкті. Це дозволяє командам ефективно координувати свої дії, забезпечуючи прозорість та зворотний зв'язок на всіх етапах виконання проєкту.

Автоматизація. *Smartsheet* пропонує функції автоматизації, які дозволяють користувачам створювати тригери та дії на основі певних умов. *Наприклад*, можна налаштувати автоматичні сповіщення про зміни статусу завдань, що значно підвищує ефективність управління проєктами та знижує ризик пропуску важливих термінів.

Візуалізація даних. *Smartsheet* надає різноманітні можливості для візуалізації даних, такі як діаграми Ганта, *Kanban-дошки* та дашборди. Це допомагає командам швидко оцінити статус проєкту, виявити потенційні проблеми та планувати наступні кроки на основі реальних даних.

Інтеграції. Завдяки великій кількості інтеграцій з іншими популярними сервісами (такими як *Google Workspace*, *Microsoft Office 365*, *Salesforce* та багато інших), **Smartsheet** забезпечує користувачам можливість легко поєднувати дані та автоматизувати процеси в межах різних платформ. Це робить його універсальним рішенням для управління проектами.

Безпека та контроль доступу. **Smartsheet** забезпечує високий рівень безпеки даних, пропонуючи різноманітні опції контролю доступу. Користувачі можуть налаштовувати права доступу для членів команди, що дозволяє захищати чутливу інформацію та гарантувати, що тільки уповноважені особи можуть вносити зміни в проєкт.

Стартова сторінка при запуску у веб-браузері буде виглядати приблизно так, як наведено на рис. 5.1.

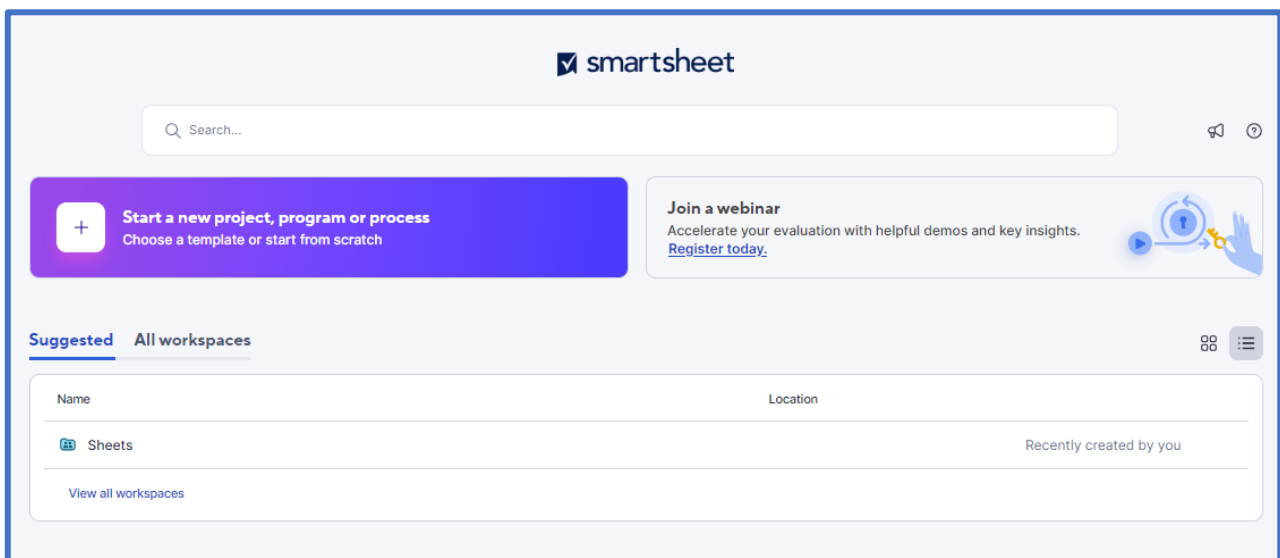


Рис. 5.1 – Початок роботи з сервісом для управління проєктами Smartsheet

Завдяки своїй здатності до спільної роботи, візуалізації даних та інтеграції з іншими сервісами, **Smartsheet** допомагає командам зосередитися на досягненні своїх цілей, підвищуючи продуктивність та ефективність управління проєктами.

5.1.1. Гнучке керування таблицями для проєктів

Smartsheet надає гнучкі можливості для налаштування форматування таблиць, що дозволяє користувачам персоналізувати вигляд своїх проєктів і зробити інформацію більш зрозумілою та доступною. Це сприяє підвищенню продуктивності команди та полегшує управління проєктами. Використовуючи різноманітні функції, ви можете створити таблицю, яка відповідає вашим вимогам і потребам.

Щоб почати, користувач може натиснути кнопку «*Guided Project Setup*» (рис. 5.2), щоб створити нову таблицю, в якій буде зібрана вся необхідна інформація про проєкт. Це перший крок до організації даних, які допоможуть вашій команді з легкістю виконувати свої завдання.

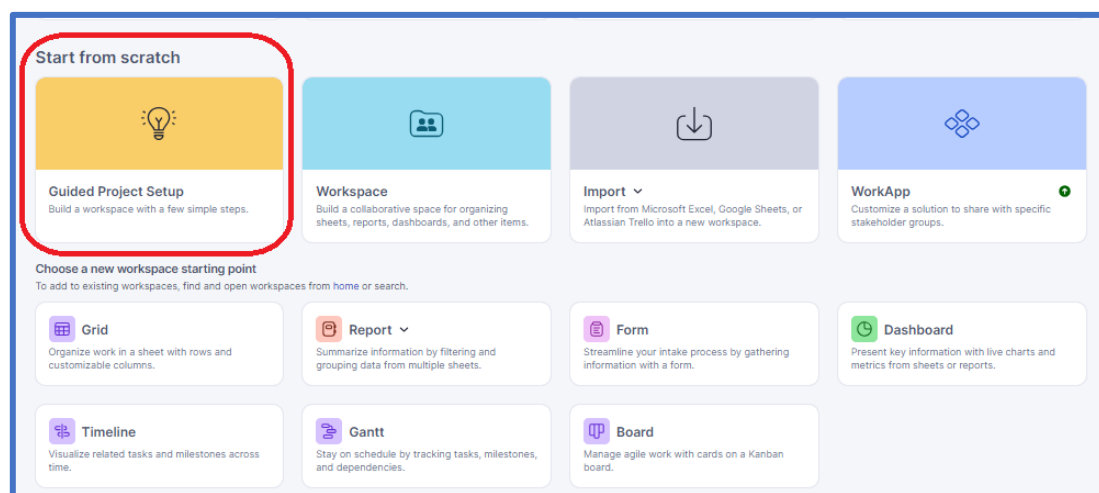


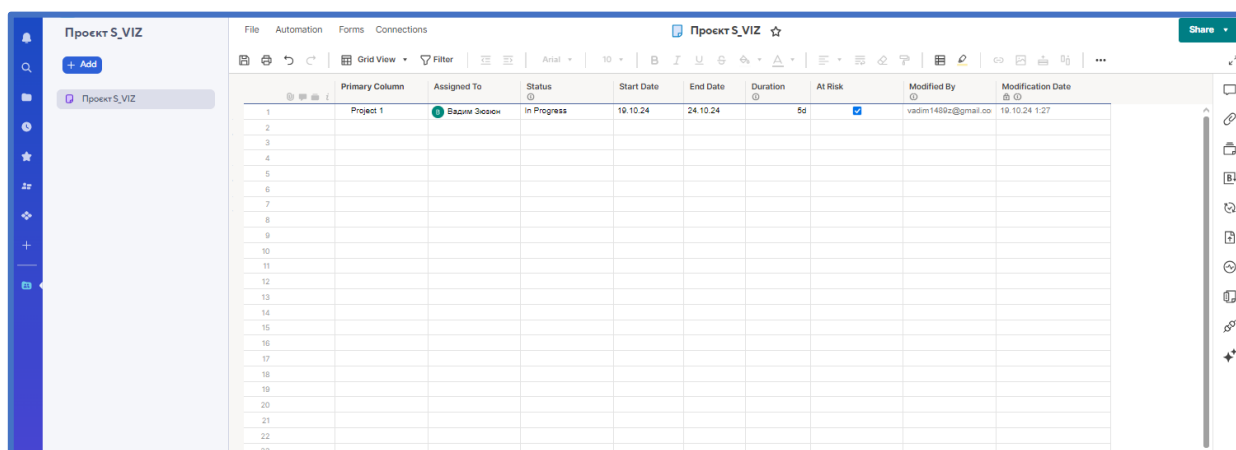
Рис. 5.2 – Створення первинного робочого середовища

Далі важливо вибрати шрифти та кольори. Дослідивши відповідне меню, розташоване над полотном таблиці, користувач може вибрати стиль, розмір та колір шрифту, що дозволяє створити естетичний і зручний для читання вигляд таблиці. **Наприклад**, можна обрати більш яскраві кольори для заголовків, щоб вони виділялися на фоні таблиці, що зробить їх легшими для сприйняття.

Форматування стовпців також є важливим етапом. Розглянувши представлене полотно таблиці, користувач може змінити ширину стовпців, вирівнювання тексту та додати заголовки, щоб організувати дані у зручному для сприйняття вигляді. **Наприклад**, у таблиці з завданнями менеджер проєкту може

збільшити ширину стовпця з описом завдання, щоб забезпечити його повне відображення.

Додавання умовного форматування дозволяє створювати правила, які змінюють форматування клітинок залежно від значень. Це дає можливість швидше візуалізувати критично важливу інформацію, таку як терміни або статус завдань. **Наприклад**, можна налаштувати так, щоб клітинки із термінами, що наближаються, автоматично підсвічувалися червоним кольором, сповіщаючи команду про пріоритетність виконання (рис. 5.3).



The screenshot shows a web-based project management tool. The main area is a grid with columns for 'Primary Column', 'Assigned To', 'Status', 'Start Date', 'End Date', 'Duration', 'At Risk', 'Modified By', and 'Modification Date'. The first row contains data for 'Project 1', assigned to 'Вадим Зосенко', with a status of 'In Progress', start and end dates of '19.10.24', a duration of '5d', and a checked 'At Risk' box. The interface includes a sidebar on the left with navigation icons and a top menu with options like 'File', 'Automation', 'Forms', and 'Connections'.

	Primary Column	Assigned To	Status	Start Date	End Date	Duration	At Risk	Modified By	Modification Date
1	Project 1	Вадим Зосенко	In Progress	19.10.24	24.10.24	5d	<input checked="" type="checkbox"/>	vadim1489z@gmail.co	19.10.24 1:27
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									
16									
17									
18									
19									
20									
21									
22									
23									

Рис. 5.3 – Первинне форматування робочого середовища

Користувач також може натиснути кнопку «Add Comment», щоб надати додаткову інформацію про конкретні дані в клітинках. Це підвищує зрозумілість інформації і дозволяє команді легше здійснювати комунікацію. **Наприклад**, у клітинці з інформацією про бюджет можна залишити коментар, пояснюючи, чому були внесені зміни (рис. 5.4).

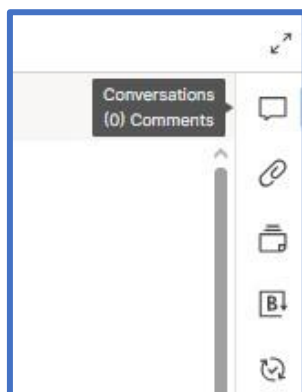


Рис. 5.4 – Робота з меню для додавання коментарів

Для полегшення роботи з таблицею, користувач може натиснути кнопку «*Filters*», щоб налаштувати фільтри і сортування даних. Це дозволяє зосередитися на найбільш актуальних даних і підвищує ефективність роботи з таблицею. **Наприклад**, можна створити фільтр для відображення лише активних завдань або тих, які потребують термінового виконання (рис. 5.5).

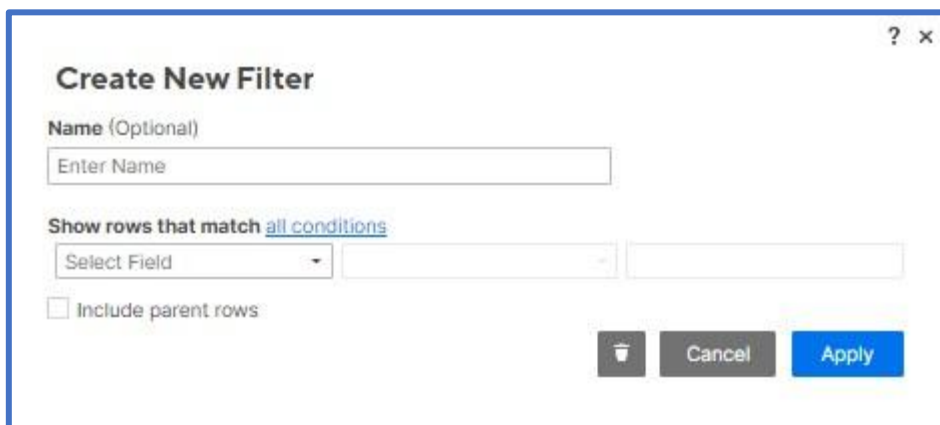


Рис. 5.5 – Створення нового фільтру

Після завершення налаштувань, користувач може натиснути кнопку «*Save as Template*» (Зберегти як шаблон), щоб зберегти налаштування форматування таблиці. Це дозволяє використовувати їх у майбутніх проєктах, заощаджуючи час на налаштування.

Завершивши налаштування, користувач може експортувати таблицю, натиснувши кнопку «*Export*», для збереження таблиці в різних форматах, таких як *Excel* або *PDF*, для подальшого використання. Це зручно для представлення інформації іншим членам команди або стейкхолдерам.

Розглянемо декілька **прикладів** щодо використання форматування в **Smartsheet**:

1. Проєкт «Організація конференції». Команда створює таблицю для відстеження завдань, де менеджер налаштовує шрифт, вибираючи корпоративний стиль. Він також додає умовне форматування для клітинок з термінами, які підсвічуються жовтим, якщо залишилося менше тижня до виконання.

2. Проєкт «Запуск нового продукту». У таблиці команда налаштовує ширину стовпців для опису завдань, щоб кожен міг прочитати деталі без необхідності прокручувати. Користувачі додають коментарі до клітинок із бюджетом, пояснюючи, чому були змінені цифри.
3. Проєкт «Модернізація IT-інфраструктури». Команда використовує фільтри для відображення лише тих завдань, які завершені або в процесі виконання. Менеджер налаштовує шаблон форматування, щоб легко повторно використовувати його в наступних проєктах.
4. Проєкт «Проведення маркетингової кампанії». Користувачі створюють таблицю, де у клітинках статусу завдань використовують умовне форматування, яке змінює колір тексту в залежності від статусу («Виконано» – зелений, «В процесі» – жовтий, «Не виконано» – червоний). Це дозволяє швидко оцінити прогрес кампанії.

Таким чином, налаштування форматування таблиць у *Smartsheet* допомагає не тільки візуалізувати дані, а й покращити комунікацію в команді, підвищуючи загальну ефективність управління проєктами.

Візуалізація даних через діаграми

Також *Smartsheet* надає можливості щодо візуалізації даних через діаграми, що дозволяє командам наочно представити інформацію, спростити аналіз і прийняття рішень. Використання діаграм допомагає краще розуміти стан проєкту, аналізувати прогрес та виявляти тенденції. Завдяки цьому ви зможете швидше реагувати на зміни та оптимізувати свої процеси.

Щоб розпочати, користувач може натиснути кнопку «*Dashboard*», що дозволяє створити нову діаграму на основі даних з таблиць. Це швидкий спосіб візуалізувати інформацію та вибрати тип діаграми, що найкраще відображає дані вашого проєкту.

Після натискання кнопки «*Add Widget*», користувач може вибрати з різних типів діаграм (графіки, стовпчикові, кругові тощо), щоб найбільш ефективно представити дані. **Наприклад**, для аналізу витрат можна обрати кругову діаграму, що чітко демонструє, як розподілені ресурси між різними категоріями (рис. 5.6).

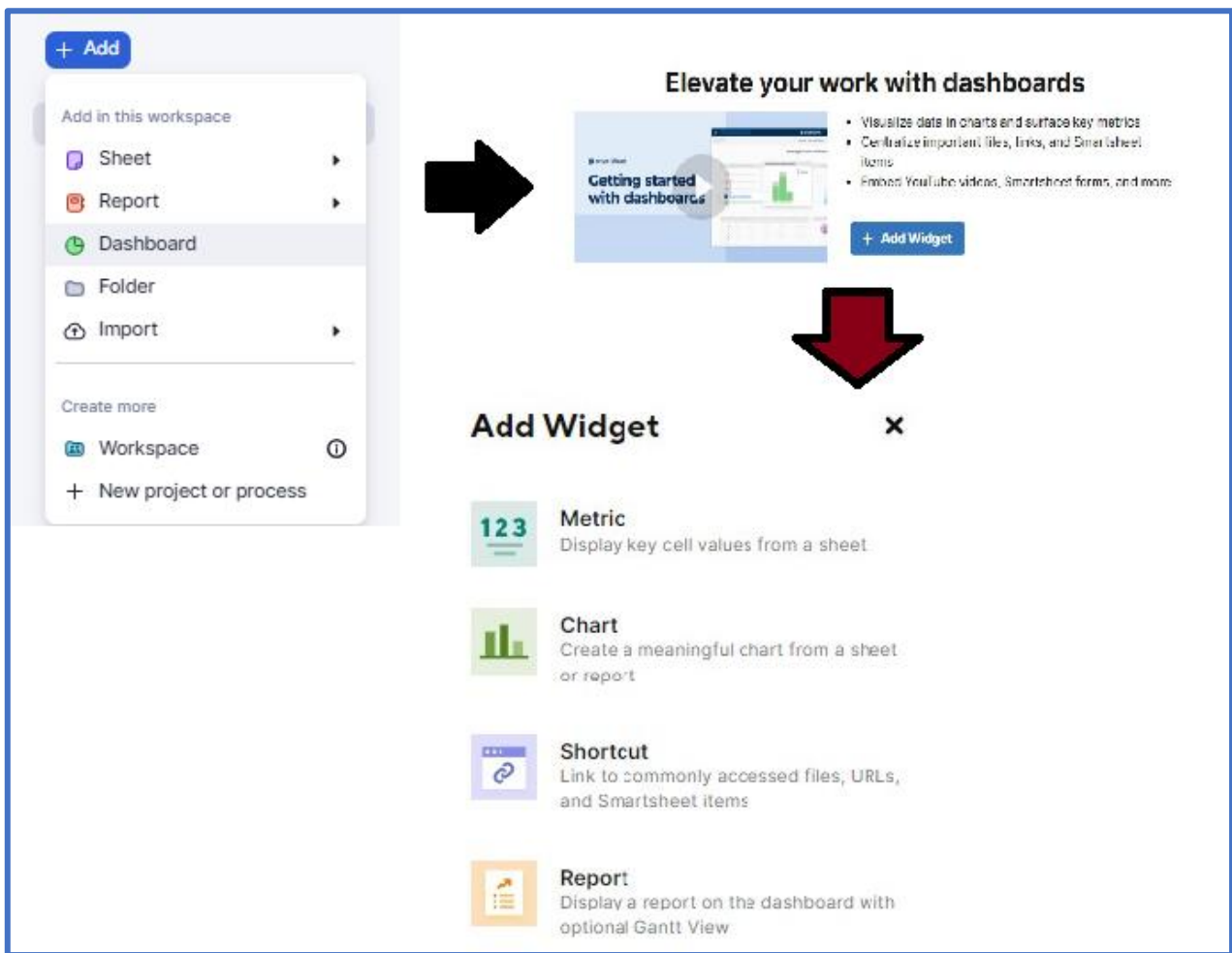


Рис. 5.6 – Робота з віджетами

Далі, натиснувши кнопку «*Customize Data*», користувач може вибрати конкретні полів і рядків, які потрібно відобразити на діаграмі. Це дозволяє налаштувати діаграму під специфічні потреби проєкту. **Наприклад**, у проєкті з продажу продуктів можна вибрати дані лише за останній квартал, щоб детально проаналізувати результати.

Після створення діаграми, користувач може натиснути кнопку «*Format Chart*» (Форматувати діаграму), щоб змінити кольори, шрифти та інші елементи дизайну, роблячи діаграму більш читабельною та привабливою. **Наприклад**, зміна кольору стовпців залежно від статусу завдань («зелені» – виконані, «жовті» – в процесі, «червоні» – не виконані) може допомогти швидше візуалізувати ситуацію.

Користувач також може натиснути кнопку «*Add Comment*», щоб надати додаткову інформацію або контекст до діаграми, що допомагає членам команди краще зрозуміти дані. Наприклад, коментар про зменшення витрат у певній категорії може пояснити, чому ця інформація важлива для поточного проєкту.

Після завершення налаштувань, користувач може експортувати діаграму, натиснувши кнопку «*Export Chart*», для збереження діаграми в форматі зображення або PDF для подальшого використання в презентаціях або звітах. Це зручно для надання інформації клієнтам або для обговорення на засіданнях.

Крім того, користувач може натиснути кнопку «*Share Chart*», щоб надати доступ іншим членам команди або партнерам. Це забезпечує спільну роботу і обмін інформацією, що важливо для ефективного управління проєктом.

Розглянемо декілька **прикладів** щодо використання діаграм у **Smartsheet**:

1. Проєкт «Аналіз витрат». Команда використовує кругову діаграму для візуалізації витрат на різні категорії: маркетинг, дослідження та розробка, операційні витрати. Завдяки такій діаграмі менеджер може швидко виявити, в яких сферах витрати перевищують заплановані.
2. Проєкт «Моніторинг виконання завдань». Менеджер проєкту обирає стовпчикову діаграму для відображення прогресу виконання завдань за квартал. Він налаштовує дані так, щоб відобразити кількість виконаних і незавершених завдань, що дозволяє команді оцінити загальний стан проєкту.
3. Проєкт «Розробка маркетингової кампанії». Команда використовує лінійну діаграму, щоб відстежувати щотижневі результати кампанії. Зміна трендів продажів та взаємодії з клієнтами наочно показується у діаграмі, що допомагає адаптувати стратегію в реальному часі.
4. Проєкт «Забезпечення якості». Використовуючи стовпчикову діаграму, команда візуалізує дані про кількість знайдених дефектів у продуктах за місяць. Це дозволяє швидко визначити, в яких продуктах чи процесах потрібно провести додаткову перевірку.

Таким чином, візуалізація даних через діаграми в *Smartsheet* є важливим інструментом для аналізу інформації, що допомагає командам приймати обґрунтовані рішення та зберігати фокус на досягненні спільних цілей у проєкті.

5.1.2. Шаблони проєктів

Щодо налаштування шаблонів проєктів *Smartsheet* надає якісний функціонал, що дозволяє командам швидко та ефективно розпочати нові проєкти, забезпечуючи структурованість і організованість з самого початку. Шаблони зменшують час на налаштування та допомагають уникнути помилок, завантажуючи стандартні елементи управління проєктами. Це зручний інструмент для полегшення старту нових ініціатив та швидкої адаптації під конкретні вимоги проєкту.

Основний функціонал для роботи з шаблонами проєктів наступний:

Вибір шаблону. Користувач може натиснути кнопку «*Choose Template*» (*Обрати шаблон*), щоб переглянути наявні шаблони проєктів. Це дозволяє швидко знайти підходящий варіант відповідно до типу проєкту, будь то розробка програмного забезпечення, маркетингова кампанія або управління подіями.

Налаштування шаблону. Після вибору шаблону користувач може натиснути кнопку «*Customize Template*» (*Налаштувати шаблон*), щоб додати зміни. Наприклад, можна додати або видалити завдання, поля, змінити терміни виконання чи відповідальних осіб. Це дає змогу адаптувати шаблон до специфічних потреб вашого проєкту (рис. 5.7).

Копіювання шаблону. Для створення нового проєкту на основі існуючого шаблону користувач може натиснути кнопку «*Copy Template*» (*Копіювати шаблон*). Це дозволяє зберегти всі налаштування, структуру та основні елементи керування, що прискорює початок роботи над новими завданнями, зберігаючи спадковість проєктів.

Імпорт даних. Користувач може натиснути кнопку «*Import Data*» (*Імпортувати дані*), щоб перенести інформацію з інших джерел, таких як Excel або *Google Sheets*, у шаблон. Це спрощує процес початку нового проєкту,

дозволяючи швидко додати існуючі дані та інтегрувати їх у *Smartsheet*.

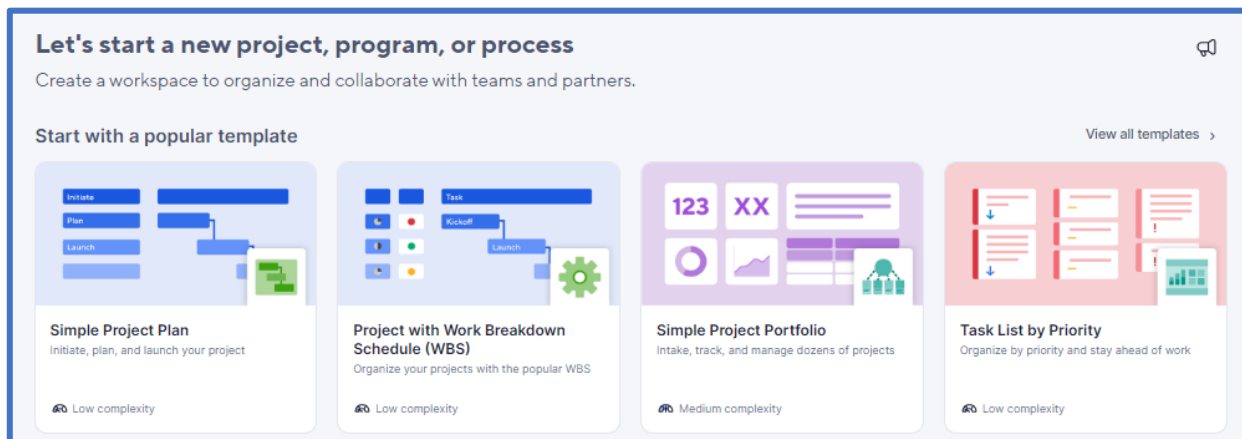


Рис. 5.7 – Вибір та налаштування шаблону

Створення документації. Для покращення комунікації та управління проектом користувач може натиснути кнопку «*Add Documentation*», щоб додати важливі файли, інструкції або звіти, що стосуються проекту. Це допомагає підтримувати порядок і легкий доступ до необхідної інформації всередині команди.

Збереження як шаблону. Після налаштування проекту користувач може натиснути кнопку «*Save as Template*» (*Зберегти як шаблон*), щоб зберегти створений проект для подальшого використання. Це дозволяє використовувати налаштований шаблон знову або ділитися ним з іншими командами для спільної роботи.

Спільний доступ до шаблонів. Користувач може натиснути кнопку «*Share Template*» (*Поділитися шаблоном*), щоб надати доступ іншим членам команди або партнерам. Це забезпечує спільну роботу, даючи можливість використовувати однакові шаблони для стандартизації проектів у компанії.

Розглянемо декілька *прикладів* щодо використання шаблонів у *Smartsheet*:

1. Команда використовує шаблон *Smartsheet* для управління проектом запуску нового продукту. Менеджер вибирає шаблон «Запуск ІТ продукту» через кнопку «Обрати шаблон», а потім налаштовує його, додаючи нові завдання, пов'язані з маркетинговими дослідженнями та виробництвом. Це допомагає команді зосередитися на кожному етапі запуску та дотримуватися термінів.

2. Для проведення стратегічного планування компанія використовує шаблон «Планування та моніторинг стратегії». Після імпорту даних зі старих таблиць, менеджер налаштовує шаблон, додаючи нові цілі та ключові показники ефективності для кожного кварталу. Шаблон спрощує відстеження прогресу виконання стратегії і підтримує команду в досягненні довгострокових цілей.
3. Менеджер подій використовує шаблон «Організація ІТ заходу» для планування щорічної конференції. Шаблон містить стандартні завдання з логістики, маркетингу та зв'язків з громадськістю. Після налаштування деталей, таких як дати виступів та контакти постачальників, команда може зосередитися на конкретних аспектах події без втрати часу на налаштування процесів.

Завдяки використанню шаблонів у *Smartsheet*, команди можуть заощадити час на підготовку та забезпечити структурованість роботи, дозволяючи сконцентруватися на досягненні цілей проєкту.

5.2. Створення та відстеження завдань

Smartsheet пропонує розширені можливості для створення та відстеження завдань, що дозволяє командам ефективно управляти проєктами, контролювати прогрес та досягати поставлених цілей у встановлені терміни. Завдяки інтуїтивному інтерфейсу та широкому функціоналу, користувачі можуть організувати свою роботу, мінімізувати ризики пропущених дедлайнів і забезпечити прозорість виконання завдань.

Основні можливості для створення та відстеження завдань представлено нижче.

Створення завдання. Процес створення завдань у *Smartsheet* починається з додавання завдань в колонку «*Primary Column*». Варто зазначити, що функціонал *Smartsheet* надзвичайно гнучкий, що в свою чергу дозволяє перейменовувати навіть базові колонки. Це дозволяє командам чітко визначати ключові етапи проєкту, деталізувати кожне завдання та встановлювати його пріоритети. Таке

завдання стає будівельним блоком загального успіху проєкту, а прозора структура допомагає усім членам команди зрозуміти, які саме дії необхідно виконати (рис. 5.8).

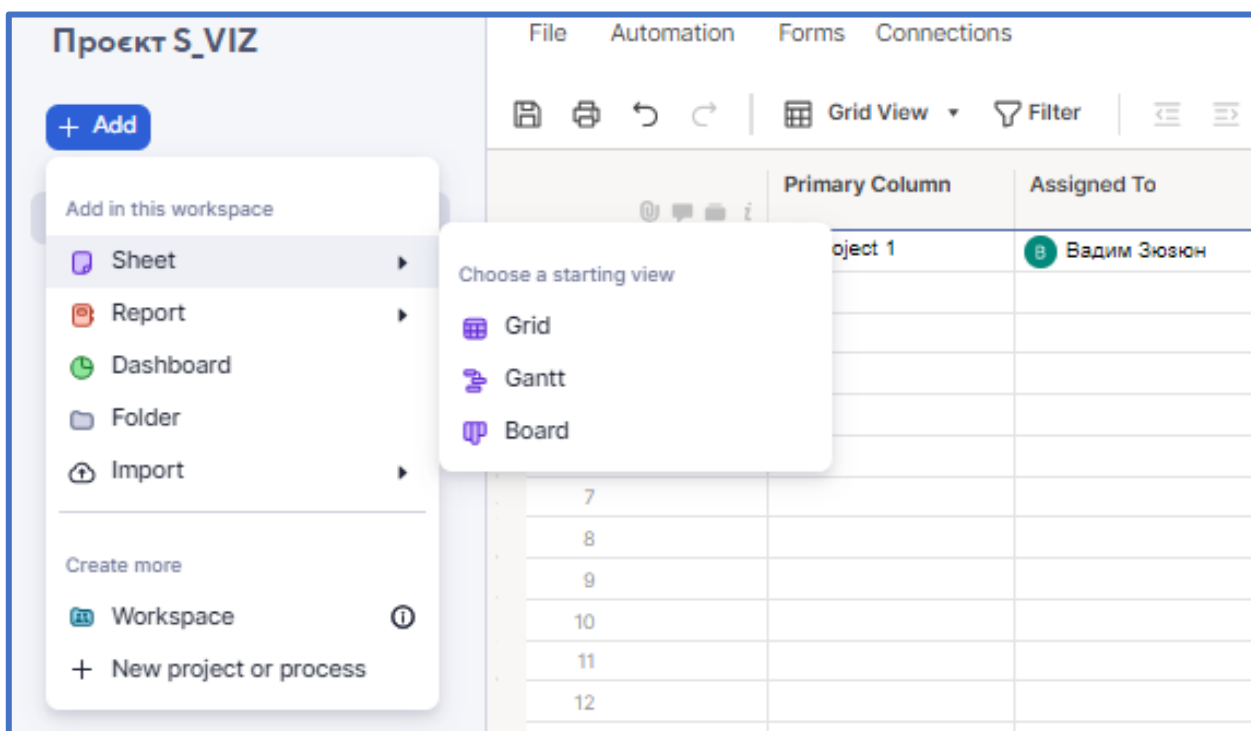


Рис. 5.8 – Вибір форми відображення «Grid» та додавання завдань в колонку *Primary Column*

Призначення відповідальних осіб. **Smartsheet** дозволяє легко призначати завдання конкретним членам команди. Додавши інформацію до колонки «*Assigned To*», менеджер проєкту може швидко закріпити кожне завдання за відповідною особою. Це чітко визначає обов'язки та гарантує, що ніхто не залишиться без конкретних задач, а робота буде розподілена рівномірно.

Відстеження прогресу. Щоб завжди бути в курсі стану проєкту, **Smartsheet** надає можливість відстежувати прогрес у реальному часі. Натискання кнопки «*Status*» дозволяє користувачам позначати, на якій стадії виконання знаходиться завдання – воно може бути в процесі, завершене чи потребувати додаткової уваги. Це допомагає команді швидко реагувати на зміни та тримати проєкт під контролем.

Визначення термінів виконання. Для успішного завершення будь-якого проєкту надзвичайно важливо встановити чіткі терміни виконання. Користувач, додавши інформацію до колонок «*Start Date*» та «*End Date*», може точно визначити дату початку та завершення завдання. Це дозволяє команді мати зрозумілу часову шкалу для кожного етапу роботи, підвищуючи контроль за дотриманням графіка та уникаючи невинуватих затримок.

Візуалізація завдань. **Smartsheet** пропонує зручну функцію візуалізації завдань, яка дозволяє легко оглянути всі задачі в проєкті на єдиній панелі. Натискання кнопки «*Grid View*» (рис. 5.9) відкриває повну картину того, що вже зроблено, що ще в роботі, а також допомагає виявляти завдання, що потребують негайної уваги.

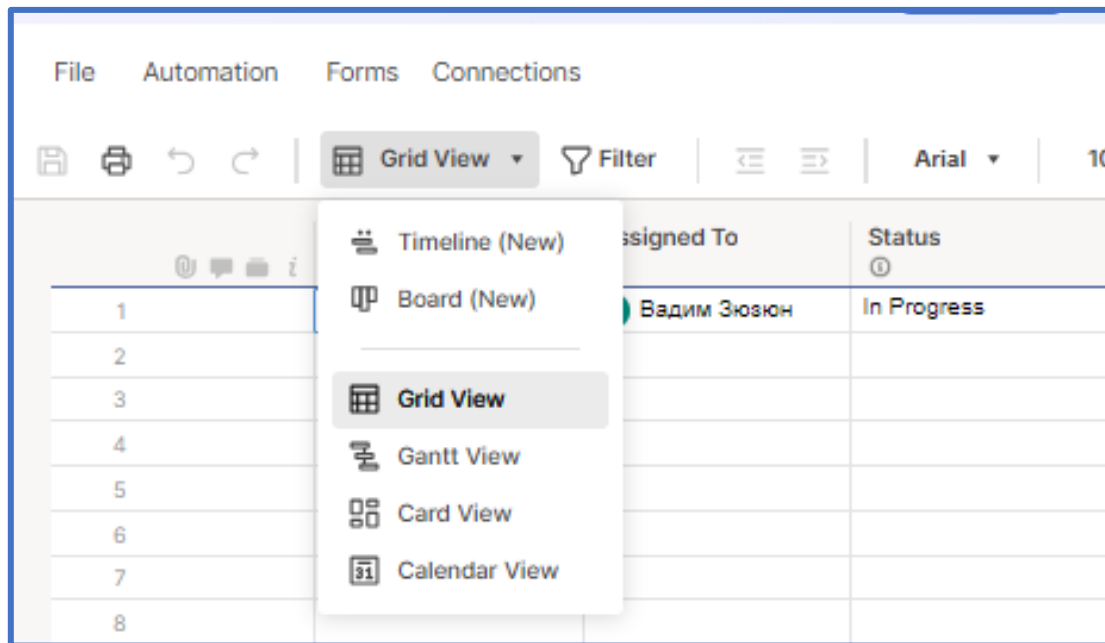


Рис. 5.9 – Вибір форми відображення «*Grid View*»

Коментарі та співпраця. Кожне завдання у **Smartsheet** може супроводжуватися детальними коментарями від членів команди. Натискання кнопки «*Add Comment*» дозволяє учасникам спілкуватися безпосередньо в контексті конкретного завдання, обмінюватися думками, додавати примітки та пропозиції. Це покращує співпрацю та забезпечує відкритий канал комунікації всередині проєкту.

Фільтрація та сортування. **Smartsheet** дозволяє налаштовувати відображення завдань за допомогою фільтрів, що спрощує управління великими проектами. Натискання кнопки «*Filters*» дозволяє сортувати завдання за різними критеріями – статусом, відповідальними особами, термінами тощо. Це допомагає командам зосередитися на найбільш важливих завданнях і швидко знаходити потрібну інформацію.

Розглянемо декілька *прикладів*:

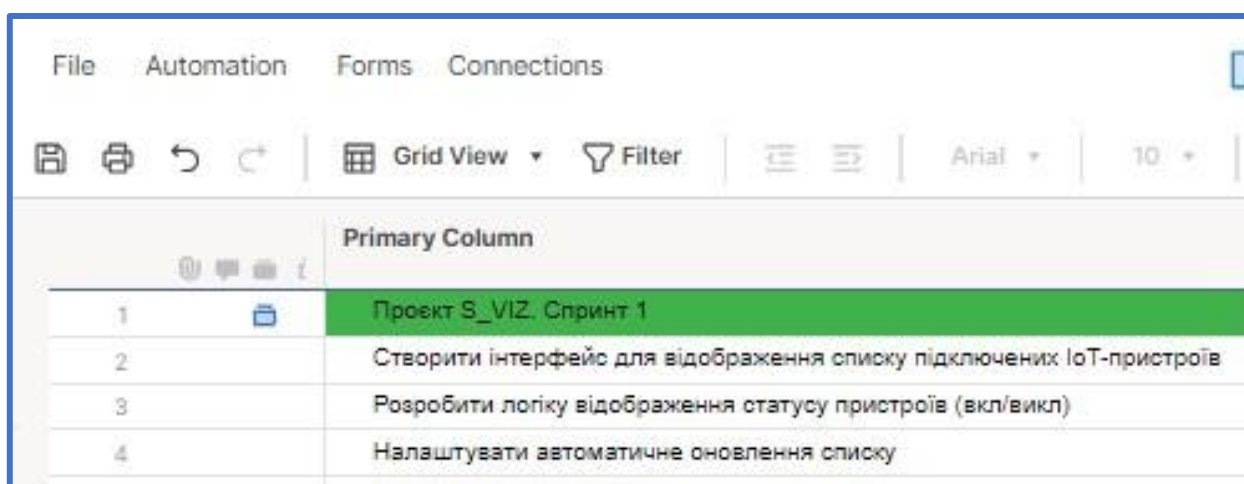
1. В проєкті зі створення мобільного додатку команда **Smartsheet** активно використовує можливість створення завдань. Менеджер проєкту додає завдання для кожного етапу розробки – від проєктування дизайну до тестування. Завдяки функції «*Assigned To*», він швидко розподіляє обов'язки між програмістами, дизайнерами та спеціалістами з тестування. Всі члени команди вчасно оновлюють статус своїх завдань, що дозволяє ефективно відстежувати прогрес і уникати збоїв.
2. У проєкті з розробки веб-платформи менеджер використовує **Smartsheet** для встановлення чітких термінів виконання кожного етапу. Завдяки можливості задавати дати, команда точно знає, коли потрібно завершити дизайн, інтеграцію баз даних та тестування. Візуалізація завдань дозволяє менеджеру відразу бачити, які завдання виконані, а які потребують додаткових ресурсів.
3. В рамках модернізації IT-інфраструктури компанії **Smartsheet** використовується для організації завдань із заміни серверного обладнання. Кожен етап, від демонтажу старих серверів до налаштування нових, документується у вигляді окремих завдань. Менеджер призначає відповідальних інженерів, фіксує крайні терміни і, використовуючи коментарі, забезпечує швидкий обмін інформацією щодо проблем та прогресу.

Таким чином, **Smartsheet** надає всі необхідні інструменти для ефективного управління завданнями в рамках будь-якого проєкту, підвищуючи прозорість, організованість і ефективність командної роботи.

5.2.1. Розподіл завдань та їх пріоритизація

Smartsheet пропонує потужні інструменти для розподілу завдань та встановлення пріоритету кожного з них, що дозволяє командам ефективно управляти робочим навантаженням та забезпечувати вчасне виконання проєктів з максимальним ефектом. Завдяки гнучкому функціоналу, менеджери проєктів можуть легко створювати завдання, призначати відповідальних осіб, встановлювати терміни виконання, визначати пріоритети та візуалізувати навантаження кожного члена команди. Ці можливості допомагають уникати перевантаження працівників та забезпечують належне виконання ключових завдань у зазначені терміни.

Процес управління завданнями у *Smartsheet* розпочинається зі створення нового завдання, що здійснюється за допомогою додавання його у відповідний рядок колонки «*Primary Column*». Це дозволяє менеджерам одразу визначати чіткі цілі та терміни виконання, а також задавати структуру робочого процесу. Кожне завдання може бути детально описане, включаючи всі необхідні деталі для його успішного виконання (рис. 5.10).



The screenshot shows the Smartsheet interface with a menu bar (File, Automation, Forms, Connections) and a toolbar (Grid View, Filter, Arial, 10). Below the toolbar is a table with a header row labeled 'Primary Column'. The table contains four rows of tasks:

	Primary Column
1	Проект S_VIZ. Спринт 1
2	Створити інтерфейс для відображення списку підключених IoT-пристроїв
3	Розробити логіку відображення статусу пристроїв (вкл/викл)
4	Налаштувати автоматичне оновлення списку

Рис. 5.10 – Формування переліку завдань які потрібно буде виконати в межах першого спринту

Наступним кроком є призначення відповідальних осіб, яке здійснюється додаванням інформації до відповідної комірки колонки «*Assigned To*». Ця функція допомагає забезпечити чіткість відповідальності за виконання

конкретного завдання, що дозволяє уникати непорозумінь у команді та підвищує загальну ефективність роботи. Менеджери можуть призначати завдання різним членам команди залежно від їх компетенцій і доступності (рис. 5.11).

Після створення завдання важливо встановити конкретні терміни виконання, що робиться шляхом додавання інформації до відповідної комірки колонок «*Start Date*» та «*End Date*» (рис. 5.11). Це дозволяє чітко визначити часові рамки для кожного завдання, допомагаючи командам дотримуватись графіку і контролювати прогрес. Встановлення термінів є ключовим фактором у забезпеченні своєчасного завершення проєкту.

Далі, для того щоб забезпечити максимальну ефективність роботи, кожне завдання можна пріоритетувати, використовуючи колонку «*Set Priority*» (дуже часто ця колонка не є в базовому варіанті таблиці, тому її варто просто додати а відредагувати від власні потреби). Це допомагає команді зосередитися на найважливіших завданнях першочергово, що особливо важливо при обмежених ресурсах чи стиснених термінах. Встановлення пріоритетів дозволяє краще організувати робочий процес і уникнути відволікання на другорядні завдання (рис. 5.11).

	Primary Column	Assigned To	Status	Set Priority	Start Date	End Date	Duration
1	Проект S_VIZ. Спринт 1	Вадим Зюсюк	In Progress		13.10.24	24.10.24	11d
2	Створити інтерфейс для відображення списку підключених IoT-пристроїв	Розробник	Complete		13.10.24	16.10.24	3d
3	Розробити логіку відображення статусу пристроїв (вкл/викл)	Розробник	In Progress		16.10.24	17.10.24	1d
4	Налаштувати автоматичне оновлення списку	Розробник	Not Started		18.10.24	21.10.24	3d
5	Протестувати результат	Тестувальник	Not Started		22.10.24	24.10.24	2d

Рис. 5.11 – Відображення форматування завдань, відповідно додаючи виконавця, терміни виконання, статус та пріоритет

Smartsheet також пропонує можливість групування завдань, що здійснюється за допомогою кнопки «*Group Tasks*». Це дозволяє організувати завдання за категоріями або етапами проєкту, що спрощує управління великими проєктами та допомагає краще орієнтуватися у структурі робочого процесу.

Групування завдань допомагає командам тримати все під контролем і забезпечує легкий доступ до пов'язаних завдань, що підвищує ефективність командної роботи.

Щоб забезпечити рівномірний розподіл роботи серед членів команди, **Smartsheet** надає функцію візуалізації робочого навантаження (рис. 5.12), яка активується через кнопки «*Resource Management*» → «*View Resource Allocation*» → «*Workload View*» (*Переглянути навантаження*). Це дозволяє менеджерам отримати загальний огляд навантаження кожного члена команди, виявити можливі перевантаження або недовантаження, і відповідно відкоригувати розподіл завдань. Такий підхід допомагає уникнути вигорання працівників та забезпечити ефективну роботу всієї команди.

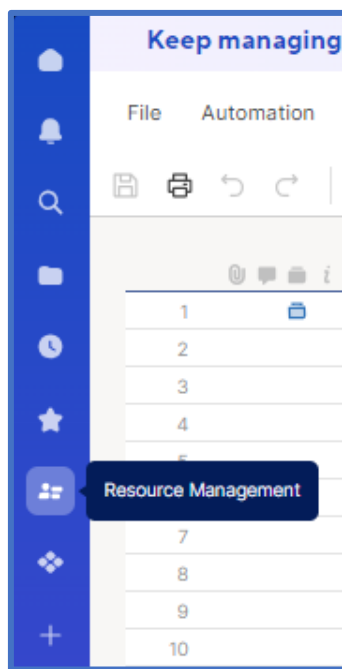


Рис. 5.12 – Доступ до функціоналу «*Resource Management*»

Комунікація є важливим елементом успішного управління проектами, тому **Smartsheet** надає можливість залишати коментарі до кожного завдання за допомогою кнопки «*Comment*». Це дозволяє учасникам обговорювати деталі, ставити запитання або надавати додаткову інформацію щодо виконання завдання. Така функція значно покращує комунікацію в команді, забезпечуючи постійний зворотний зв'язок.

Наприклад, у проєкті «Розробка мобільного додатку» команда активно використовує можливості *Smartsheet* для ефективного розподілу завдань, в тому числі і встановлення пріоритетності цих завдань. Менеджер проєкту створює завдання для кожного етапу розробки за допомогою кнопки «Додати завдання», призначає відповідальних осіб, натискаючи «Призначити», та встановлює терміни виконання через «Встановити терміни». Встановивши пріоритет кожного завдання за допомогою «Встановити пріоритет», команда може зосередитися на критичних завданнях. Окрім цього, використовуючи функцію «Переглянути навантаження», менеджер оцінює робоче навантаження всіх учасників і переконується, що немає перевантаження.

Іншим прикладом може слугувати проєкт з модернізації ІТ-інфраструктури в якому менеджер використовує *Smartsheet* для створення завдань з демонтажу старого обладнання та встановлення нових серверів. Розстановка пріоритетів для завдань дозволяє команді почати роботу з найбільш критичних частин інфраструктури, що потребують термінового оновлення. Завдяки функції перегляду робочого навантаження команда вчасно коригує свої плани, уникаючи перевантаження інженерів.

Відповідно, *Smartsheet* надає командам повний контроль над завданнями, дозволяючи ефективно організовувати робочі процеси, уникати перевантаження та забезпечувати своєчасне виконання проєктів.

5.2.2. Моніторинг прогресу виконання завдань

Щодо інструментів для моніторингу прогресу виконання завдань *Smartsheet* дозволяє командам ефективно контролювати стан проєктів у режимі реального часу. Завдяки цьому можна оперативно виявляти потенційні затримки та приймати своєчасні рішення для усунення проблем, забезпечуючи тим самим успішне завершення проєкту вчасно і в межах бюджету.

Одним з основних інструментів моніторингу є можливість відстеження статусу завдань. За допомогою колонки «*Status*» може легко змінювати статус виконання будь-якого завдання, що дозволяє підтримувати актуальність

інформації про хід проєкту. Це надзвичайно важливо для забезпечення прозорості робочого процесу та своєчасного інформування всіх учасників команди про виконання певних етапів.

Smartsheet також надає можливість візуалізувати прогрес за допомогою кнопки «*Add Widget*» (див. рис. 5.6). Візуалізація може бути представлена у вигляді графіків, діаграм або інших візуальних засобів, що дозволяють миттєво отримати загальний огляд стану проєкту. Такий підхід сприяє більш легкому сприйняттю великого обсягу даних, допомагаючи швидко оцінювати, які завдання виконані, а які знаходяться в процесі або відстають від графіка.

Користувачі можуть налаштувати сповіщення про зміни статусу завдань за допомогою кнопки «*Notifications*» (*Налаштувати сповіщення*). Це дозволяє отримувати автоматичні повідомлення про будь-які важливі оновлення у проєкті, що значно підвищує оперативність комунікації в команді. Завдяки цій функції, керівники та учасники команди завжди залишаються в курсі ключових подій та змін у проєкті (рис. 5.13).

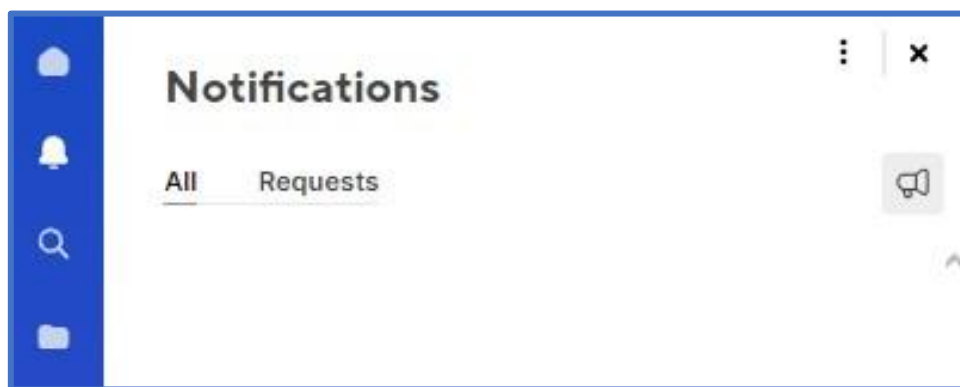


Рис. 5.13 – Доступ до функціоналу «*Notifications*»

Для виявлення потенційних затримок *Smartsheet* пропонує інструмент аналізу, який активується кнопкою «*Analyze Delays*» (*Аналізувати затримки*). Ця функція дозволяє детально дослідити, які завдання відстають від запланованого графіка, та вчасно вживати заходів для виправлення ситуації. Аналіз затримок допомагає керівникам проєктів зосередитися на проблемних ділянках і швидко приймати рішення, щоб запобігти серйозним порушенням у виконанні проєкту.

Завершуючи етап або весь проєкт, користувач може згенерувати детальний звіт про прогрес за допомогою кнопок «*Row Report*» або «*Sheet Summary Report*» (рис. 5.14). Це дозволяє отримати глибокий аналіз виконання завдань, включаючи ключові показники та досягнення. Такий звіт є важливим інструментом для оцінки успіхів проєкту та використовується для подальшого планування наступних етапів або для звіту перед керівництвом.

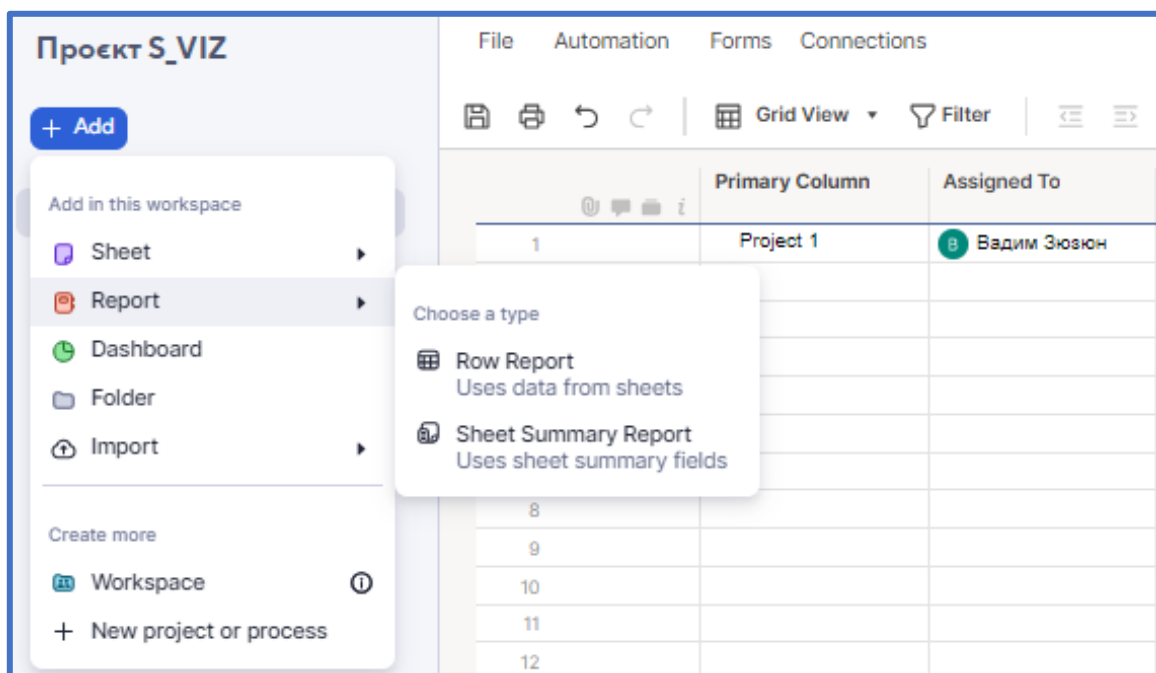


Рис. 5.14 – Доступ до функціоналу «Report»

Окрім цього, *Smartsheet* дозволяє обговорювати прогрес виконання завдань за допомогою функції коментування. Використовуючи кнопку «*Comment on Task*», члени команди можуть залишати коментарі щодо стану завдання або обговорювати необхідні зміни. Це забезпечує належний рівень зворотного зв'язку та сприяє кращій командній співпраці.

Щоб отримати загальний огляд всього проєкту, користувач може скористатися функцією візуалізації дорожньої карти, натискаючи кнопку «*Card View*». Ця функція дозволяє бачити всі завдання та їхні терміни виконання в єдиній системі, що полегшує контроль за виконанням проєкту і дозволяє ефективно планувати наступні кроки (рис. 5.15).

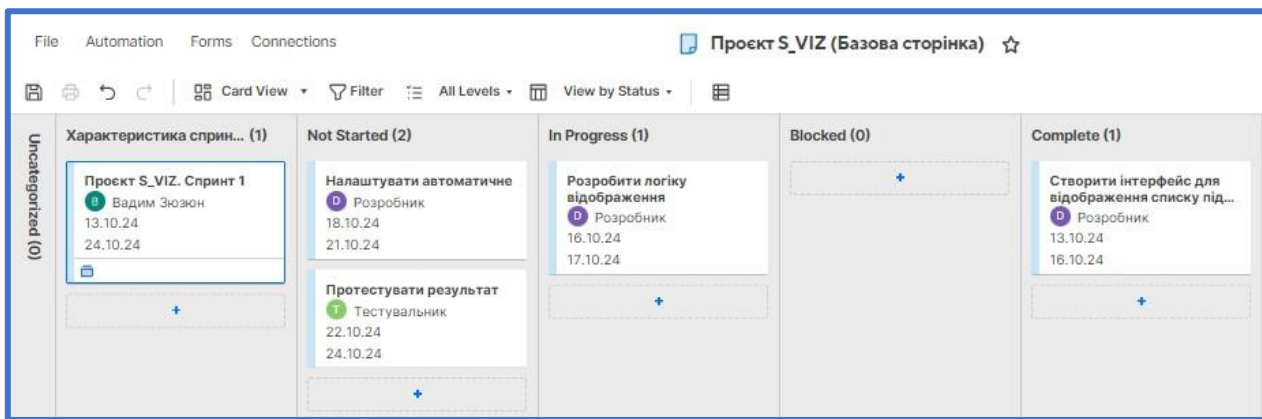


Рис. 5.15 – Перегляд завдань за допомогою функціоналу «Card View»

Розглянемо декілька **прикладів**:

1. У проєкті з розробки програмного забезпечення команда **Smartsheet** активно використовує всі інструменти для моніторингу прогресу. Після завершення етапу тестування, розробник оновлює статус завдання, натискаючи кнопку «Оновити статус». Менеджер проєкту відстежує загальний прогрес за допомогою кнопки «Переглянути прогрес» і швидко помічає, що деякі завдання відстають від графіка. Використовуючи функцію «Аналізувати затримки», він проводить аналіз причин відставання і організовує командну нараду, де обговорює можливі рішення через функцію «Коментувати завдання». Після завершення етапу менеджер генерує звіт про прогрес, натискаючи кнопку «Генерувати звіт про прогрес», щоб підсумувати досягнення команди та окреслити подальші плани.

2. У іншому проєкті зі створення маркетингової стратегії менеджер використовує **Smartsheet** для контролю за виконанням завдань на всіх етапах. Відділ створення контенту оновлює статус завдань через «Оновити статус», а менеджер щодня переглядає загальний прогрес через «Переглянути прогрес». При виявленні затримок у розробці рекламних матеріалів, він швидко визначає джерело проблеми за допомогою «Аналізувати затримки» і спрямовує зусилля на вирішення. По завершенню кампанії звіт про досягнуті результати формується через «Генерувати звіт про прогрес».

5.3. Управління ресурсами та часовими рамками

Smartsheet надає потужні інструменти для управління ресурсами (*Resource Allocation*) та часовими рамками, що дозволяє командам ефективно планувати та контролювати виконання проєктів, забезпечуючи баланс між робочим навантаженням співробітників та термінами виконання завдань. Використовуючи ці функції, керівники проєктів можуть спрощувати управління, підвищувати ефективність та уникати перевантаження ресурсів.

Одним з ключових елементів управління у *Smartsheet* є створення проєктних графіків. Користувач може легко створити новий графік, натискаючи кнопку «*Timeline*» (5.16), де будуть відображені всі ключові етапи та терміни виконання проєкту. Це дозволяє чітко бачити, як просувається проєкт і які завдання необхідно виконати в конкретні дати. Завдяки цій візуалізації команда отримує краще уявлення про загальний план і терміни.

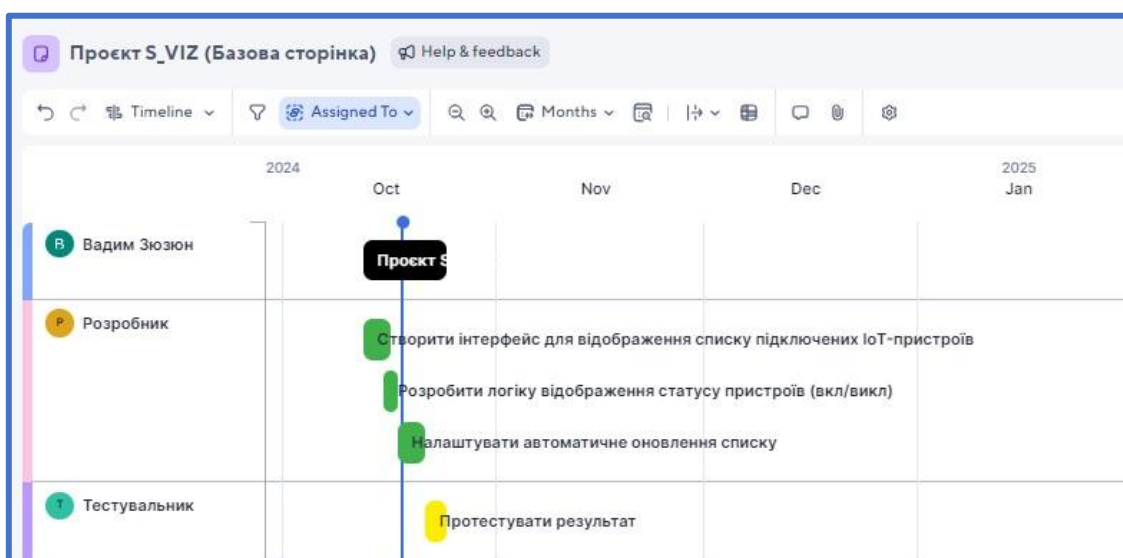


Рис. 5.16 – Візуальне відображення роботи функціоналу «*Timeline*»

Щоб визначити конкретні дедлайни для завдань, користувач може скористатися функцією призначення термінів, редагуючи відповідні комірки в колонках «*Start Date*» та «*End Date*». Це дає змогу точно окреслити часові рамки для кожного завдання, що є важливим аспектом планування. *Smartsheet* допомагає не тільки структурувати робочий процес, а й уникати непорозумінь щодо кінцевих дат.

Для управління ресурсами проєкту користувач може додавати необхідних членів команди через кнопку «Add User» використовуючи функцію «Project setting», або використавши можливість додати учасника команди безпосередньо у потрібну комірку колонки «Assigned To» (рис. 5.17).

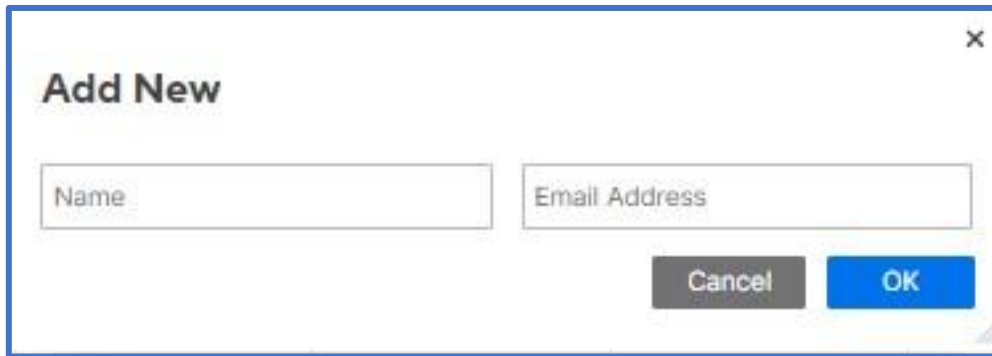


Рис. 5.17 – Додавання учасника команди безпосередньо у потрібну комірку колонки «Assigned To»

Також додати нових учасників можливо шляхом використання функціоналу, доступ до якого можна отримати шляхом наступного покрокового алгоритму, а саме «Share» → «Workspace» (рис. 5.18).

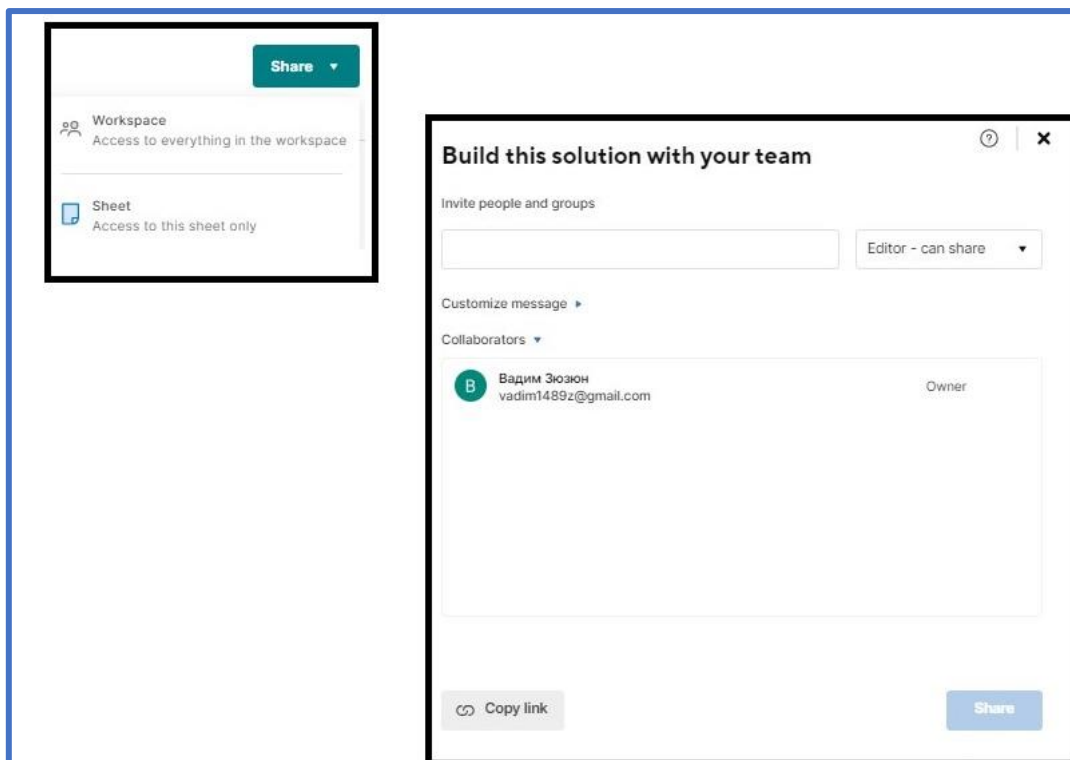


Рис. 5.18 – Додавання учасника команди

Ця функція дозволяє чітко вказати, хто з команди буде відповідальний за виконання того чи іншого завдання, а також ефективно розподіляти робочі навантаження. Після цього *Smartsheet* надає можливість переглянути навантаження на кожного члена команди через кнопки «*Resource Management*» → «*View Resource Allocation*» → «*Workload View*». Ця функція особливо корисна для того, щоб виявити можливі перевантаження або недостатнє використання ресурсів, що дозволяє збалансувати робочий процес.

Крім того, *Smartsheet* надає інструменти для контролю термінів виконання завдань у реальному часі. За допомогою кнопки «*Calendar View*» (керівник проекту може оперативнo відстежувати прогрес і своєчасно реагувати на можливі затримки). Приклад як виглядає даний функціонал на практиці наведено на рис. 5.19. У поєднанні з функцією автоматизації нагадувань, активованою через кнопку «*Set a Reminders*» (*Налаштувати нагадування*), система надсилає повідомлення про наближення дедлайнів, що допомагає команді уникати порушень термінів та затримок у роботі.

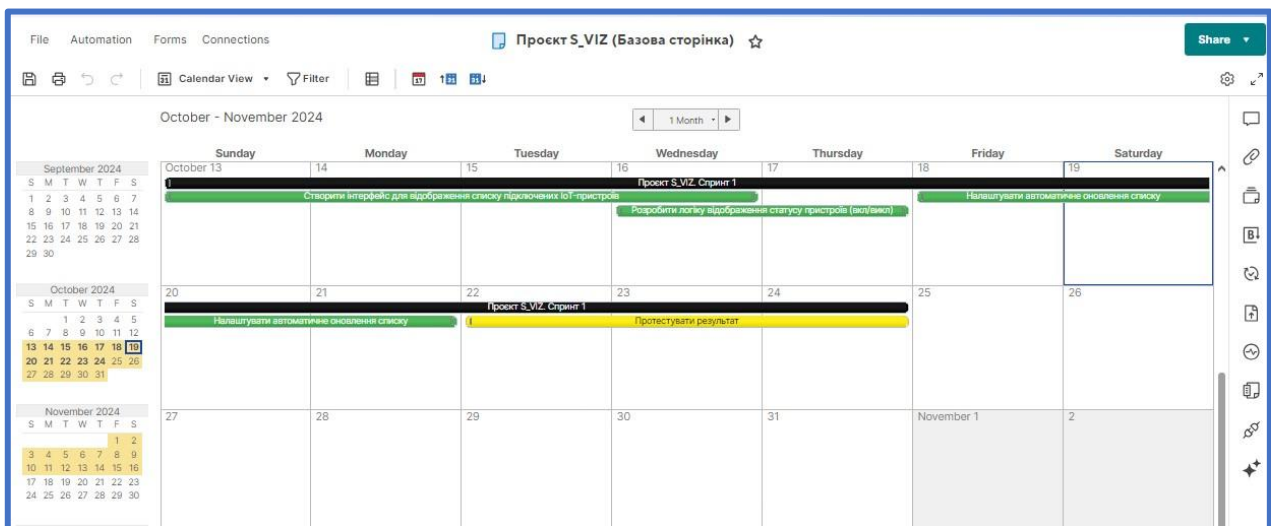


Рис. 5.19 – Візуальне відображення роботи функціоналу «*Calendar View*»

Для оцінки виконання проекту Smartsheet пропонує можливість створювати детальні звіти про виконання завдань за допомогою кнопок «*Report*». Ці звіти містять усю необхідну інформацію для аналізу досягнень, включаючи терміни завершення завдань, рівень залучення ресурсів та ефективність виконання.

Розглянемо декілька *прикладів*:

1. У проєкті «Розробка мобільного додатку» команда активно використовує можливості *Smartsheet* для управління ресурсами та часовими рамками. Менеджер проєкту спершу створює графік виконання завдань, натискаючи кнопку «*Timeline*», і далі призначає терміни для основних етапів за допомогою редагування інформації у відповідних комірках в колонках «*Start Date*» та «*End Date*». Щоб забезпечити належний розподіл ресурсів, він додає до проєкту членів команди, використовуючи кнопку «*Add User*» і переглядає навантаження на кожного учасника за допомогою кнопки «*Workload View*». Виявивши, що один з розробників має перевантаження, він коригує розподіл завдань. Завдяки автоматичним нагадуванням про дедлайни, налаштованим через кнопку «*Set a Reminders*», команда отримує повідомлення про наближення термінів, що дозволяє уникнути затримок. Наприкінці проєкту менеджер генерує звіт про виконання, натискаючи кнопку «*Report*», і підсумовує досягнення команди, оцінюючи ефективність роботи та вчасне завершення проєкту.

У проєкті з розробки ІТ-інфраструктури для корпоративного клієнта менеджер використовує *Smartsheet* для управління ресурсами і часовими рамками на кожному етапі. Він створює графік, додає членів команди, стежить за їх навантаженням і контролює дедлайни через відповідні інструменти.

5.3.1. Визначення і контроль ресурсів

Smartsheet забезпечує командам потужні інструменти для визначення та контролю ресурсів, що дозволяє оптимізувати управління проєктами та зменшити ризик перевантаження або недостатнє завантаження співробітників. Використовуючи ці функції, менеджери можуть чітко планувати залучення ресурсів, щоб досягти максимального рівня ефективності в рамках проєкту.

Однією з ключових можливостей *Smartsheet* є додавання ресурсів до проєкту. Користувач може включити нового члена команди, редагуючи інформацію в комірках колонки «*Assigned To*», що дозволяє зазначити, хто відповідатиме за виконання певних завдань. Це дає змогу уникнути

непорозуміннь щодо ролей та відповідальності учасників команди у проєкті. Після того, як ресурси додані, користувач може призначити їх до конкретних завдань, скориставшись функцією «*Assigned To*». Ця дія забезпечує чітке розуміння, хто за що відповідає, що є важливим для злагодженої роботи команди.

Наступним кроком є визначення доступності ресурсів. За допомогою кнопки «*Set Availability*» (*Налаштувати доступність*) користувач може задати години роботи кожного члена команди, враховуючи графіки їхньої роботи під час планування завдань. Це особливо корисно, коли команда працює в гнучкому графіку, адже дозволяє уникнути ситуацій, коли завдання призначені на години, коли ресурс недоступний.

Для перевірки навантаження на ресурси користувач може натиснути кнопку «*View Workload*». Ця функція надає змогу виявити, чи є співробітники перевантаженими або, навпаки, недостатньо завантаженими. Наприклад, якщо керівник проєкту виявляє, що один з розробників має занадто багато завдань, він може скористатися функцією «*Edit Plans*», щоб перенести частину завдань або додати зміни до розподілу ресурсів, що забезпечить оптимальну роботу команди без затримок.

Після завершення етапу проєкту користувач може створити звіт про використання ресурсів, натискаючи кнопок «*Row Report*» або «*Sheet Summary Report*». Цей звіт міститиме детальну інформацію про ефективність використання ресурсів, що допоможе проаналізувати, наскільки правильно були розподілені завдання та який рівень залучення кожного члена команди.

Розглянемо декілька **прикладів**:

1. У проєкті «Розробка веб-платформи» команда активно використовує **Smartsheet** для визначення та контролю ресурсів. Менеджер проєкту додає співробітників, натискаючи кнопку «*Add User*», і призначає їх до завдань за допомогою функції «*Assigned To*». Після цього він перевіряє навантаження, натискаючи кнопку «*Workload View*», і виявляє, що один з розробників перевантажений. Завдяки функції «*Edit Plans*» менеджер коригує графік, перенісши частину завдань на інший день, що забезпечує оптимальну

роботу команди без затримок. Наприкінці проєкту він генерує звіт про використання ресурсів, натискаючи кнопку «*Row Report*» або «*Sheet Summary Report*», щоб оцінити ефективність розподілу ресурсів.

5.3.2. Відстеження термінів виконання

Smartsheet пропонує командам зручні інструменти для відстеження термінів виконання завдань в рамках проєктів. Це дозволяє ефективно контролювати хід роботи та вчасно реагувати на можливі затримки, що є критично важливим для успішного завершення проєктів.

Для зручного перегляду термінів виконання у графічному форматі, *Smartsheet* надає можливість використовувати функцію «*Gantt View*» (*Переглянути Діаграму Ганта*). Ця візуалізація надає наочне уявлення про тривалість завдань, їх послідовність та взаємозв'язки, що сприяє кращому плануванню та координації роботи (рис. 5.20).

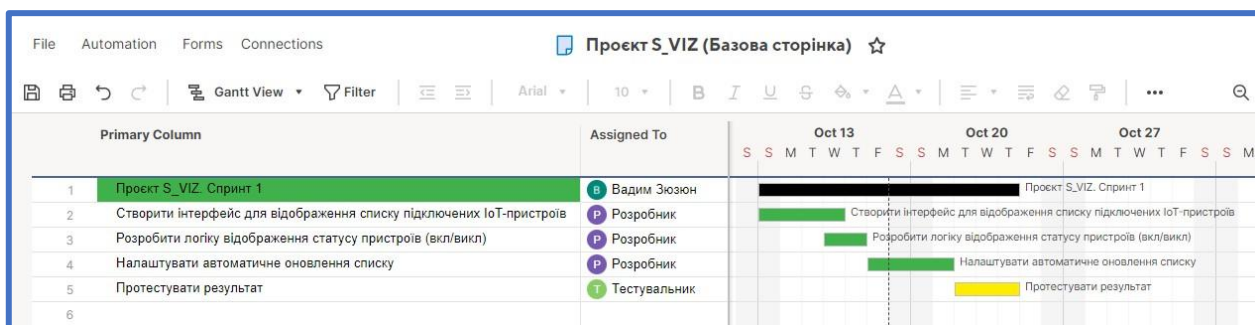


Рис. 5.20 – Діаграма Ганта в Smartsheet

Також для відстеження прогресу виконання проєкту його можна переглядати у форматі «*Board*». Приклад наведено на рис. 5.21.

Щоб не пропустити важливі дати, користувачі можуть налаштувати сповіщення про наближення термінів, натискаючи кнопку «*Set Notifications*» (*Налаштувати сповіщення*). Це дозволяє отримувати автоматичні нагадування про терміни виконання, що підвищує рівень відповідальності команди та зменшує ризик затримок.

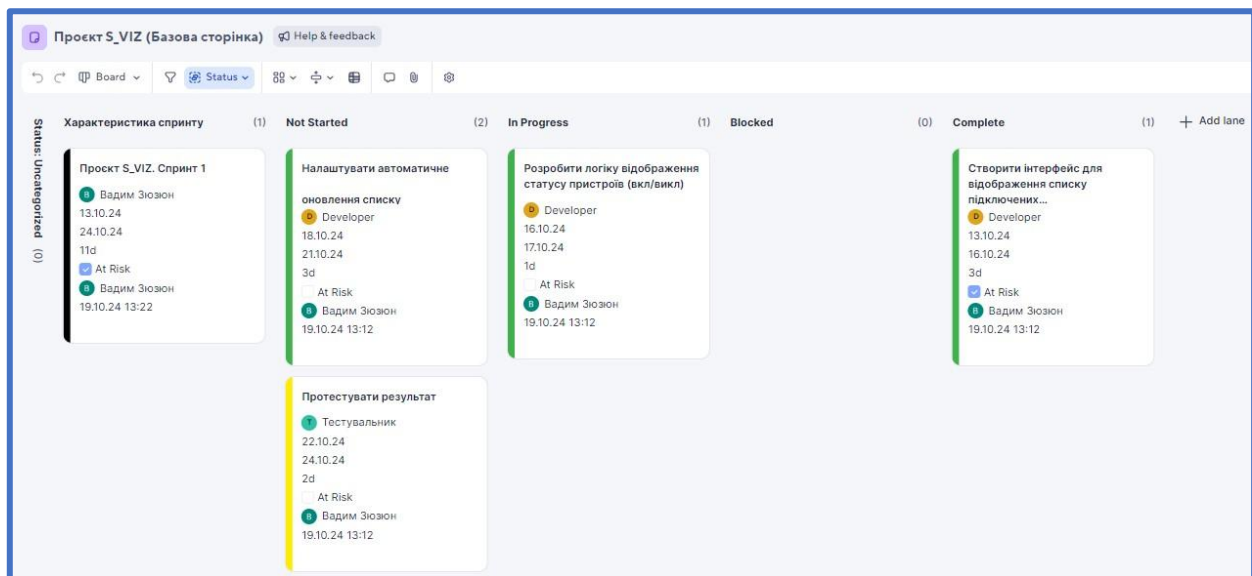


Рис. 5.21 – Відображення прогресу виконання проєкту у форматі «Board»

Наприкінці проєкту, користувач може генерувати звіт, натискаючи кнопку «Report». Цей звіт міститиме інформацію про виконання завдань відповідно до термінів, що є корисним для аналізу ефективності роботи команди.

Розглянемо декілька *прикладів*:

1. У проєкті «Створення мобільного додатку» команда використовує *Smartsheet* для відстеження термінів виконання завдань. Менеджер проєкту додає терміни для кожного завдання, працюючи з колонками «Start Date», «End Date» та «Duration», і переглядає їх у графіку через функцію «Gantt View». Коли терміни наближаються, система автоматично надсилає сповіщення про терміни, що дає змогу команді своєчасно реагувати на можливі затримки. Після завершення проєкту менеджер генерує звіт, натискаючи кнопку «Report», щоб оцінити виконання завдань відповідно до запланованих термінів.
2. У проєкті «Створення корпоративної системи» менеджер активно використовує *Smartsheet* для контролю за термінами. Він регулярно перевіряє дотримання термінів за допомогою відслідковування інформації в колонках таблиці «Start Date», «End Date» та «Duration». Коли система виявляє затримки, менеджер реагує на ситуацію, коригуючи графік виконання завдань.

5.4. Спільна робота та звітування. Автоматизація звітів про виконання проєкту

За допомогою функцій автоматизації роботи над звітністю команди можуть ефективно співпрацювати, обмінюватися інформацією та контролювати прогрес проєктів. Однією з ключових можливостей є спільний доступ до проєктів. Користувач може запросити членів команди, натискаючи кнопку «*User Management*» в категорії «*Project Setting*» (рис. 5.22), що дозволяє всім учасникам отримати доступ до актуальної інформації та спільно працювати над завданнями. Це створює умови для активного обговорення проєкту та полегшує командну взаємодію.

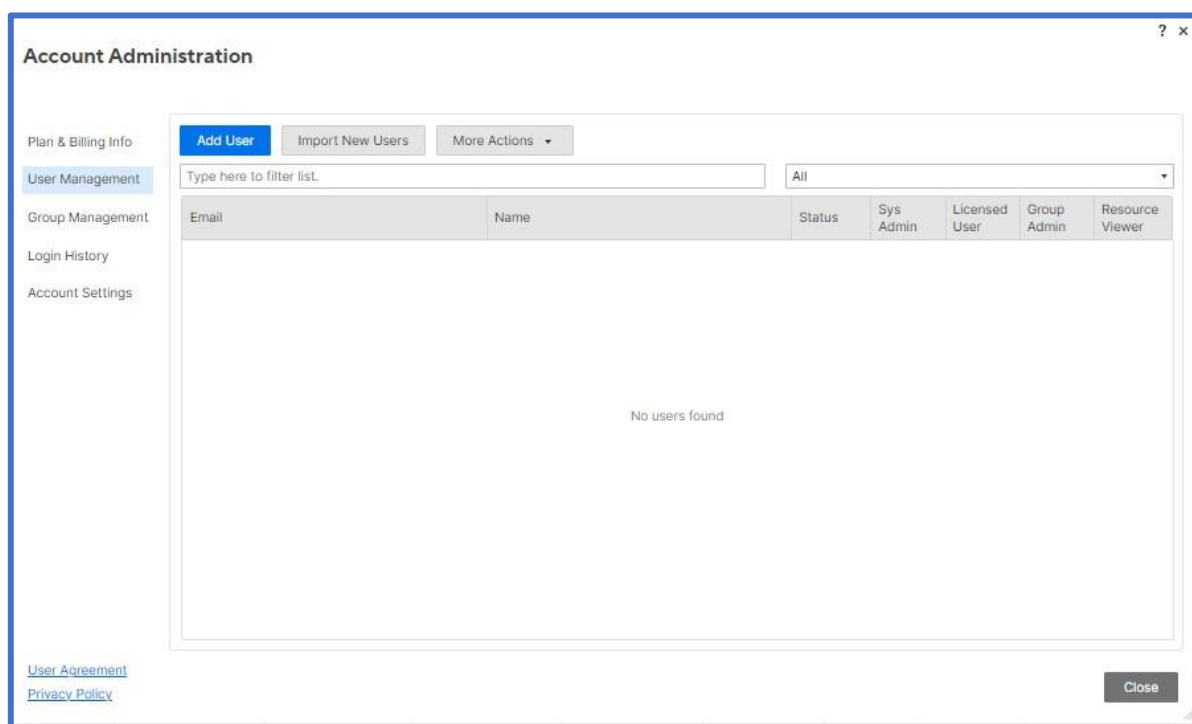


Рис. 5.22 – Налаштування спільного доступу до проєкту для команди

Крім того, *Smartsheet* дозволяє користувачам залишати коментарі в рамках завдань, натискаючи кнопку «*Comment*». Це сприяє активному обговоренню завдань та швидкому вирішенню питань, що виникають у процесі роботи. Завдяки можливості коментування учасники команди можуть ділитися своїми думками, ставити запитання та отримувати відповіді в реальному часі.

Для оптимізації процесу звітування *Smartsheet* пропонує функцію автоматизації звітів. Користувачі можуть налаштувати автоматичні звіти, натискаючи кнопку «Automation» (рис. 5.23).

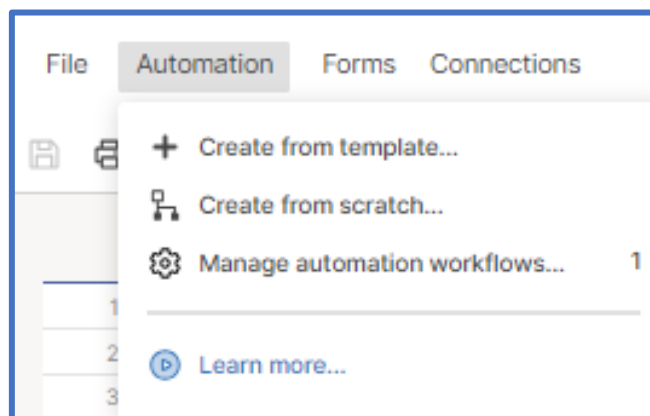


Рис. 5.23 – Використання функціоналу «Automation»

Даний функціонал дозволяє встановити графік, за яким звіти будуть надсилатися членам команди. Це забезпечує своєчасну інформацію про виконання завдань та допомагає уникнути непорозумінь.

Щоб отримувати оновлення про зміни в проекті, користувачі можуть налаштувати сповіщення. Це дає змогу тримати всіх учасників у курсі останніх подій та своєчасно реагувати на будь-які зміни.

Користувач також може завжди перевіряти актуальність інформації на дашборді, натискаючи кнопку «Refresh». Це дозволяє командам бачити всі зміни в режимі реального часу та адаптуватися до нових обставин.

Після створення звіту, користувач може експортувати його, натискаючи кнопку «Export». Це дозволяє зберігати документи у форматах *PDF* або *Excel*, що спрощує аналіз та обмін інформацією між членами команди.

Розглянемо декілька **прикладів**:

1. У проекті «Розробка нового програмного забезпечення» команда використовує *Smartsheet* для спільної роботи над завданнями. Менеджер проекту запрошує колег, натискаючи кнопки «Share» → «Workspace», і налаштовує автоматичні звіти про виконання завдань, використовуючи кнопку «Automation». Члени команди обговорюють статус завдань через

коментарі, натискаючи кнопку «*Comment*», що дозволяє оперативно вирішувати питання. Завдяки цим функціям, команда підтримує високий рівень співпраці та контролю над проектом, що сприяє успішному завершенню розробки.

2. У проекті «Впровадження CRM-системи» менеджер запрошує учасників команди, використовуючи кнопки «*Share*» → «*Workspace*», і налаштовує автоматичні звіти, щоб члени команди завжди мали доступ до актуальної інформації. Команда активно використовує коментарі для обговорення змін, що відбуваються у проекті, та проводить зустрічі для оперативного реагування на питання.
3. У проекті «Розробка мобільного додатку» команда налаштовує автоматизацію звітів, натискаючи кнопку «*Automation*», щоб щотижня отримувати оновлення про виконання завдань. Завдяки сповіщенням, налаштованим через функціонал «*Notification*», члени команди постійно залишаються в курсі останніх подій і можуть швидко реагувати на зміни.

Таким чином, ***Smartsheet*** дозволяє командам не лише ефективно працювати разом, але й забезпечує прозорість у виконанні завдань, що є запорукою успішного завершення будь-якого проекту.

Завдання для самостійного опрацювання

Завдання 1. Створення плану проєкту та налаштування завдань у Smartsheet

Хід виконання:

Етап 1. Створення аркуша проєкту.

Крок 1. Створіть новий аркуш у Smartsheet і назвіть його «Розробка інформаційної системи...».

Крок 2. Увімкніть функцію діаграми Ганта, щоб відстежувати прогрес у візуальній формі.

Крок 3. Додайте основні колонки (наприклад): «Завдання», «Відповідальний», «Стартова дата», «Кінцева дата», «Статус», «Прогрес (%)».

Етап 2. Додавання завдань та ієрархії.

Крок 1. Додайте основні завдання проєкту (наприклад):

- «Планування функціоналу».
- «Дизайн інтерфейсу».
- «Розробка».
- «Тестування».
- «Реліз».

Крок 2. Для завдання «Розробка» створіть підзавдання, використовуючи ієрархічну структуру (вкладеність):

- «Back-end розробка».
- «Front-end розробка».
- «Інтеграція API».

Крок 3. Встановіть для кожного завдання стартову та кінцеву дати.

Етап 3. Визначення залежностей між завданнями.

Крок 1. Встановіть залежності між завданнями (наприклад):

- «Розробка» може початися тільки після завершення «Планування функціоналу».
- «Тестування» може початися після завершення завдання «Розробка».

Крок 2. Переконайтеся, що всі залежності відображаються в діаграмі Ганта.

Етап 4. Призначення відповідальних.

Крок 1. У колонці «Відповідальний» додайте членів команди:

- «Планування функціоналу» – Ім'я 1.
- «Дизайн інтерфейсу» – Ім'я 2.
- «Back-end розробка» – Ім'я 3.
- «Front-end розробка» – Ім'я 4.
- «Тестування» – Ім'я 5.

Крок 2. Налаштуйте автоматичні сповіщення для членів команди про зміни в їх завданнях.

Етап 5. Сформувані короткий висновок за результатами виконання завдання 1.

Завдання 2. Відстеження прогресу та створення звітів у Smartsheet

Хід виконання:

Етап 1. Оновлення статусу та прогресу завдань.

Крок 1. Переконайтеся, що кожен член команди оновив статус своїх завдань у колонці «Статус»:

- «Планування функціоналу» – завершено.
- «Дизайн інтерфейсу» – виконується (50%).
- «Back-end розробка» – виконано на 70%.

Крок 2. Оновіть колонку «Прогрес (%)» відповідно до виконання кожного завдання.

Етап 2. Використання дашбордів для моніторингу проєкту.

Крок 1. Створіть дашборд для моніторингу прогресу проєкту:

- Додайте графік для відображення відсотка виконання основних завдань.
- Створіть віджет для відстеження прострочених завдань.

Крок 2. Додайте показники прогресу для кожного ключового етапу проєкту (планування, розробка, тестування).

Етап 3. Створення звіту.

Крок 1. Створіть звіт, який включатиме:

- Завдання, які завершені.
- Завдання, що знаходяться у процесі виконання.
- Прострочені завдання (якщо є).

Крок 2. Експортуйте звіт у форматі PDF або Excel для подальшого аналізу.

Етап 4. Сформувати короткий висновок за результатами виконання завдання 2.

Контрольні запитання до розділу

1. Що таке Smartsheet, і як його використовують для управління проектами?
2. Які основні функції Smartsheet для управління проектами?
3. Як створювати і налаштовувати нові проекти в Smartsheet?
4. Що таке робочий аркуш (sheet) і як він використовується для організації завдань?
5. Як у Smartsheet використовуються проектні шаблони для прискорення роботи?
6. Як Smartsheet підтримує методології управління проектами, такі як Agile або Scrum?
7. Які можливості пропонуються для управління командами та ролями в проектах?
8. Як відслідковується прогрес проекту за допомогою таймлайну (Gantt chart) у Smartsheet?
9. Як використовувати автоматизацію у Smartsheet для оптимізації процесів?
10. Як налаштовуються сповіщення та нагадування у Smartsheet для команди?
11. Як відслідковувати та керувати ризиками в проектах в Smartsheet?
12. Як Smartsheet можна використовувати для планування ресурсів?
13. Як у Smartsheet реалізується взаємодія між членами команди та зацікавленими сторонами?
14. Які можливості інтеграції з іншими інструментами пропонує Smartsheet (наприклад, з Google Drive, Microsoft Teams)?
15. Як використовувати дашборди в Smartsheet для відображення ключових метрик проекту?
16. Як проводиться аналіз продуктивності за допомогою звітів у Smartsheet?
17. Що таке картковий вигляд (Card View) у Smartsheet, і як його можна використовувати для управління завданнями?
18. Як у Smartsheet проводити управління бюджетами і витратами на проект?
19. Як Smartsheet забезпечує безпеку даних і конфіденційність інформації в проектах?
20. Які інструменти Smartsheet допомагають у прогнозуванні термінів виконання завдань та управлінні дедлайнами?

РОЗДІЛ 6. ВЕБ-ПЛАТФОРМА ONLINE GANTT ДЛЯ УПРАВЛІННЯ ПРОЄКТАМИ: ПЛАНУВАННЯ ТА ВІДСТЕЖЕННЯ

6.1. Основні функції Online Gantt для управління проєктами

Online Gantt надає можливість будувати графіки проєктів у формі діаграми Ганта, що дозволяє візуально представляти всі етапи виконання проєкту на часовій шкалі. Під час створення завдань, як показано на рис. 6.1, можна задати назви завдань, визначити їх тривалість та встановити дати початку і завершення.

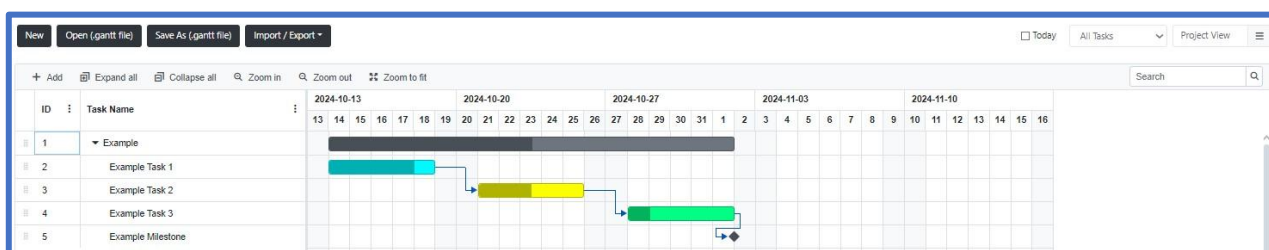


Рис. 6.1 – Базовий вигляд стартового робочого середовища в Online Gantt

Важливою перевагою є можливість динамічно змінювати строки виконання задач за допомогою перетягування смужок на графіку. Це дозволяє керівнику проєкту швидко вносити корективи, адаптуючи план під зміни умов. **Наприклад**, якщо завдання затримується, ви можете просто пересунути його смужку на пізніший термін, автоматично коригуючи залежні завдання.

Окрім цього, є можливість додавання ключових етапів (*Milestones*), які допомагають відстежувати важливі точки у проєкті. Ці етапи можуть бути зображені на діаграмі у вигляді ромбів (див. рис. 6.1), що дозволяє чітко бачити досягнення основних віх.

На рис. 6.2 наведено інші формати представлення робочого середовища та відповідно даних з якими можна працювати в **Online Gantt**, а саме: крім основної вкладки «*Project View*», це «*Project View*», «*Resource View*», «*Edit Resources*», «*Edit Settings*», «*Share*», «*Revisions*».

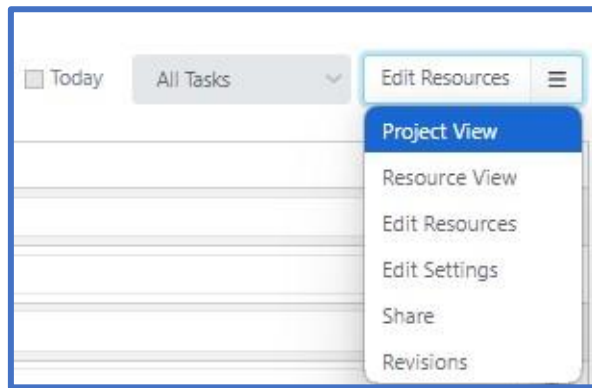


Рис. 6.2 – Робочі середовища в Online Gantt

Також зручною та важливою особливістю використання даного онлайн ресурсу є те, що користувач має можливість в кінці своєї роботи вивантажити сформований документ на ПК, натиснувши кнопку «Save As», а потім відкрити документ за допомогою кнопки «Open». Файл зберігається в розширенні (.gantt).

Відстеження виконання проєктів та контроль термінів

Для ефективного управління проєктами важливо мати інструменти контролю виконання завдань та термінів. В **Online Gantt** це реалізовано за допомогою стовпця «Progress», де для кожного завдання можна вказати процент виконання. Як видно на рис. 6.3, блакитний колір завдання «Example Task 1» показує, що воно виконане на 100%, в той час як для інших завдань прогрес може бути неповним (це помітно від градієнту насиченості кольорового забарвлення). Відповідно завдання, яке забарвлено жовтим кольором «Example Task 2», виконано на 50 %. Ця функція дозволяє швидко і наочно побачити, які завдання виконані, а які потребують додаткової уваги.

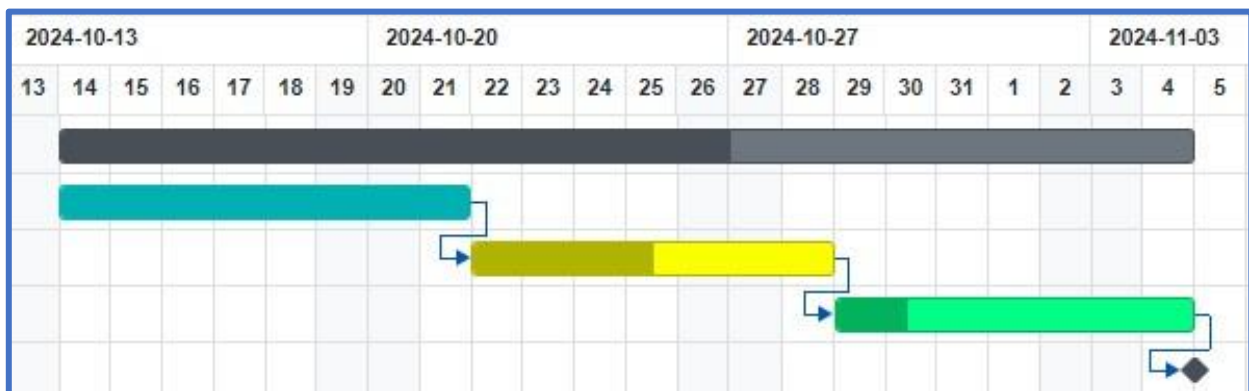


Рис. 6.3 – Відстеження завдань за терміном виконання

Крім того, ви можете включити опцію відображення критичного шляху проекту, що показує найбільш критичні завдання, затримка яких призведе до затримки всього проекту. Це допомагає сфокусувати увагу команди на ключових задачах, щоб уникнути перевищення термінів.

6.2. Інструменти для управління завданнями

6.2.1. Додавання завдань

Щоб створити нове завдання в *Online Gantt*, необхідно виконати кілька простих кроків. Платформа забезпечує зручний інтерфейс для додавання завдань, налаштування залежностей та визначення відповідальних осіб.

Додавання нового завдання. Для початку необхідно натиснути кнопку «New» у верхній частині інтерфейсу таблиці завдань (див. рис. 6.1). З'явиться вікно створення нового завдання, яке містить кілька вкладок для введення необхідної інформації:

Вкладка «General» (рис. 6.4). У цій вкладці вказуються основні параметри завдання:

- «Name» – введіть назву завдання (наприклад, «Create API Documentation»).
- «Start Date» та «End Date» – встановіть дати, що визначають час виконання завдання.
- «Duration» – автоматично розраховується або може бути вручну встановлена користувачем.

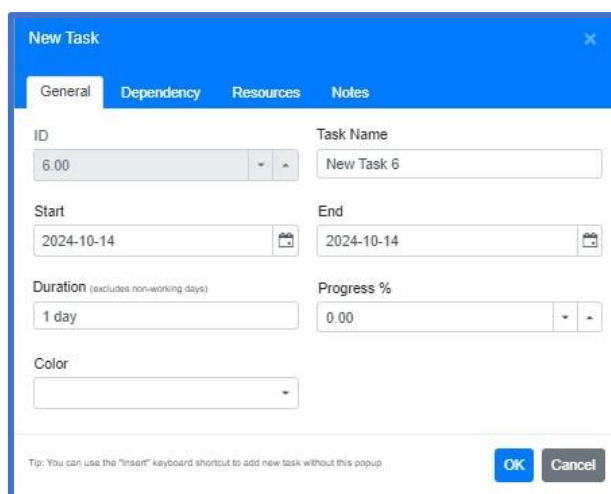
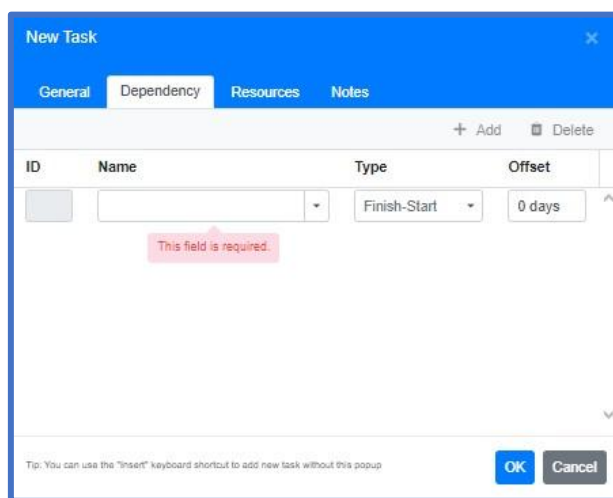


Рис. 6.4 – Додавання нового завдання

6.2.2. Спільне управління завданнями

Важливою функцією для управління проєктами є створення залежностей між завданнями. Це дозволяє ефективно синхронізувати виконання задач, запобігаючи можливим конфліктам в термінах. *Наприклад*, як показано на рис. 6.5, у вкладці «*Dependency*» (*Залежності між завданнями*) можна встановити різні типи залежностей, такі як «*Finish-Start*», що означає, що одне завдання повинне завершитися, перш ніж почнеться наступне. Інші можливі типи залежностей включають «*Start-Start*», «*Finish-Finish*» та «*Start-Finish*».



ID	Name	Type	Offset
		Finish-Start	0 days

Рис. 6.5 – Встановлення залежностей між завданнями

Ця гнучкість дозволяє адаптувати управління завданнями під специфіку проєкту. *Наприклад*, при розробці нового програмного продукту можна визначити, що тестування не може початися до завершення написання коду, і система автоматично коригуватиме строки виконання в залежності від прогресу.

6.2.2. Управління робочим навантаженням і ресурсами

Ефективне управління ресурсами є одним з ключових аспектів для успішного виконання проєкту. У *Online Gantt* можна призначати кожному завданню відповідальних осіб через вкладку «*Resources*», як показано на рис. 6.6. Це дозволяє бачити розподіл завдань між членами команди та запобігати їх перевантаженню.

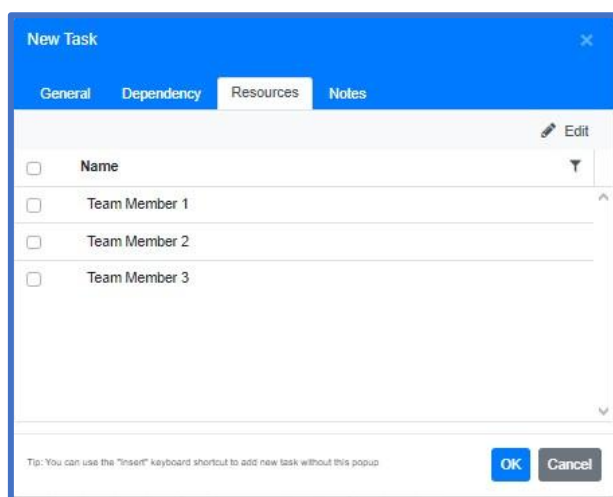


Рис. 6.6 – Призначення відповідальних осіб для виконання завдань

Додатково, система надає можливість відстежувати загальне робоче навантаження кожного співробітника, що дозволяє більш рівномірно розподіляти задачі. У випадку, якщо хтось із членів команди перевантажений, завдання можна швидко перерозподілити, щоб уникнути затримок.

6.3. Спільна робота в межах проєкту

6.3.1. Спільне редагування проєктного плану

Online Gantt підтримує функції спільної роботи в режимі реального часу, що дозволяє декільком користувачам одночасно працювати над одним і тим самим проєктом. Це особливо важливо для проєктів, де команда може бути розподілена по різних містах або країнах. Кожен користувач може редагувати завдання, додавати нові або змінювати строки, і всі зміни негайно відображаються для інших учасників проєкту (рис. 6.7).

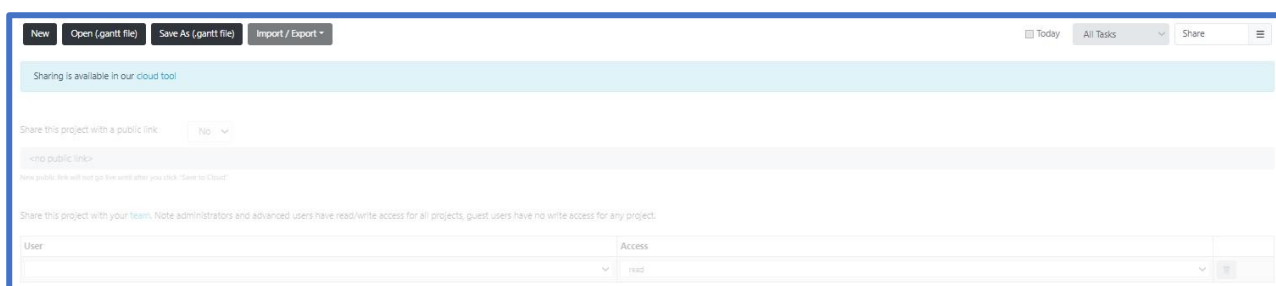


Рис. 6.7 – Налаштування спільного доступу в Online Gantt

Крім цього, існує можливість налаштовувати різні рівні доступу для користувачів. *Наприклад*, менеджери проєкту можуть мати повний доступ до всіх функцій, тоді як розробники можуть мати обмежений доступ лише до редагування певних задач.

6.3.2. Спільне управління завданнями

Як видно з рис. 6.8, кожне завдання в *Online Gantt* має вкладку «Notes», де можна залишати коментарі. Це дозволяє команді обговорювати деталі конкретних задач без необхідності використовувати зовнішні інструменти, такі як електронна пошта або месенджери. *Наприклад*, менеджер може залишити коментар щодо пріоритетності певного завдання або уточнити вимоги до виконання, що допоможе уникнути непорозумінь.

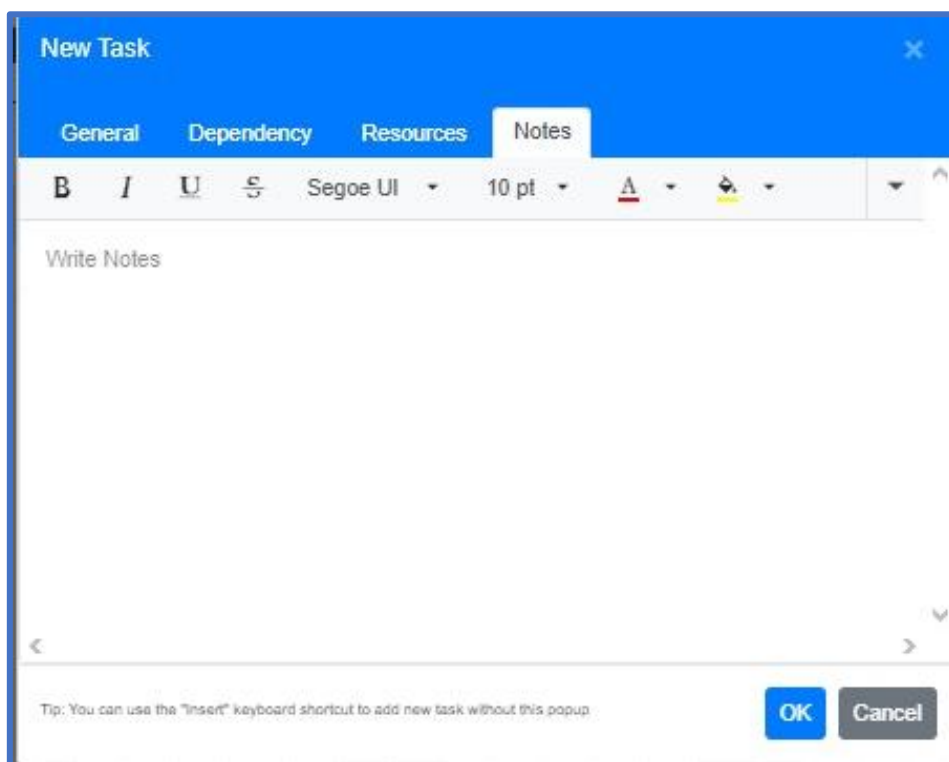


Рис. 6.8 – Робота з коментарями до завдань в *Online Gantt*

Ця функція також забезпечує прозорість комунікацій у команді, оскільки всі коментарі доступні для перегляду всім учасникам проєкту.

6.4. Звіти та аналітика

Однією з важливих функцій *Online Gantt* є можливість створення звітів про хід виконання проєкту. Ці звіти можуть містити інформацію про виконані задачі, строки, затримки та завдання, що знаходяться в процесі виконання. Звіти можна генерувати у різних форматах, наприклад у *PDF* або *Excel*, що дозволяє легко ділитися ними з командою або замовниками.

Ці звіти є корисними для регулярних зустрічей щодо статусу проєкту (*Status Meetings*), де можна детально аналізувати прогрес виконання проєкту та обговорювати необхідні коригування.

Online Gantt пропонує широкі можливості для візуалізації даних, що є важливим аспектом контролю та аналізу ефективності виконання проєктів. Ці інструменти дозволяють проєктним менеджерам не тільки моніторити поточний стан завдань, але й оцінювати прогрес у довгостроковій перспективі, виявляти ризики та приймати більш обґрунтовані рішення.

Графіки прогресу завдань. Одним із ключових інструментів візуалізації є графіки, які відображають прогрес виконання завдань. *Наприклад*, при аналізі темпів виконання завдань, система може відображати процентне співвідношення завершення кожного завдання порівняно з плановими строками. Цей графік дозволяє одразу побачити, які завдання виконуються за планом, а які – ні.

Користувачі можуть налаштовувати кольорову індикацію для різних етапів прогресу. *Наприклад*: зелений колір вказує на завдання, що йдуть за планом; червоний сигналізує про відставання; жовтий використовується для завдань, що знаходяться на межі перевищення терміну.

За допомогою таких графіків менеджери можуть легко відслідковувати, скільки часу залишилось до завершення проєкту або спринту або окремих його етапів, та коригувати дії у разі відхилень.

Графіки завантаженості ресурсів. Ще одним корисним візуальним інструментом є графіки завантаженості ресурсів. Вони показують, наскільки ефективно розподілені ресурси між завданнями. На такому графіку можна побачити, які члени команди мають занадто велике або недостатнє

навантаження, що дозволяє оптимізувати розподіл роботи. *Наприклад*, можна налаштувати перегляд завантаженості для окремих виконавців або груп і бачити, де є потенційні перевантаження. Це допоможе уникнути збоїв у роботі команди через неправильний розподіл обов'язків.

Аналіз виконання планів з часовою шкалою. Важливим аспектом візуалізації є можливість бачити історію змін у проєкті. *Online Gantt* дозволяє відстежувати, як змінювалися строки виконання завдань з часом. Це корисно для аналізу минулих етапів проєкту та виявлення причин відхилень від початкового плану. *Наприклад*, за допомогою такої візуалізації можна простежити, як і чому було змінено кінцевий термін певного завдання. Виявивши причину затримки, менеджер може запобігти повторенню цієї помилки у майбутньому.

Фільтри та сортування для аналізу даних. *Online Gantt* надає можливість застосовувати фільтри для більш детального аналізу. *Наприклад*, можна відфільтрувати завдання за статусом, виконавцем або пріоритетом, щоб краще зрозуміти загальну картину ефективності. Крім того, функція сортування дозволяє швидко знайти найбільш критичні завдання або ті, що потребують негайної уваги. Візуалізація таких даних на часовій шкалі надає зрозуміле уявлення про те, які завдання впливають на загальний термін завершення проєкту.

Історія змін у проєктах. *Online Gantt* дозволяє зберігати всю історію змін, яка відображає, як змінювались строки виконання завдань або ресурси, що залучені до проєкту. Це дозволяє аналізувати минулі етапи і приймати більш обґрунтовані рішення для майбутніх проєктів. *Наприклад*, якщо певне завдання багаторазово відкладалось через відсутність ресурсів, цей аналіз дозволяє уникнути таких ситуацій у подальших проєктах.

Експорт звітів. Візуалізовано представлені дані та аналітичні звіти можуть бути експортовані у різні формати, зокрема *PDF* або *Excel*, що дозволяє їх зберігати або передавати керівництву для подальшого аналізу. Це дає можливість легко ділитися результатами проєкту з іншими зацікавленими сторонами або замовниками.

Завдання для самостійного опрацювання

Завдання 1. Створення проєкту та планування завдань у Online Gantt

Хід виконання:

Етап 1. Створення проєкту.

Крок 1. Увійдіть до Online Gantt і створіть новий проєкт з назвою «Розробка інформаційної системи, веб-сайту, гри...».

Крок 2. Вкажіть дату початку проєкту на перший день наступного місяця і заплануйте його тривалість на 4 місяці.

Крок 3. Додайте короткий опис проєкту: «Проєкт з розробки багатофункціонального корпоративного сайту для клієнта».

Етап 2. Додавання основних завдань.

Крок 1. Додайте основні етапи проєкту як завдання (наприклад):

- «Збір вимог».
- «Проектування UX/UI».
- «Розробка та програмування».
- «Тестування та валідація».
- «Запуск сайту».

Крок 3. Для завдання «Розробка та програмування» створіть підзавдання (наприклад):

- «Front-end розробка».
- «Back-end розробка».
- «Інтеграція з базою даних».

Крок 4. Вкажіть тривалість кожного завдання та встановіть дедлайни, щоб вони відповідали загальним термінам проєкту.

Етап 3. Налаштування залежностей.

Крок 1. Налаштуйте залежності між завданнями (наприклад):

- «Проектування UX/UI» має бути завершене перед початком «Розробка та програмування».
- «Тестування та валідація» розпочинається після завершення всіх завдань із

розробки.

Крок 2. Використовуйте функцію перетягування на діаграмі Ганта, щоб побачити побудовані залежності візуально.

Етап 4. Призначення ресурсів.

Крок 1. Призначте членів команди до кожного завдання:

- «Збір вимог» – Ім'я 1.
- «Проектування UX/UI» – Ім'я 2.
- «Front-end розробка» – Ім'я 3.
- «Back-end розробка» – Ім'я 4.
- «Інтеграція з базою даних» – Ім'я 5.

Крок 2. Налаштуйте завантаження кожного члена команди, щоб уникнути перевантаження.

Етап 5. Сформувані короткий висновок за результатами виконання завдання 1.

Завдання 2. Відстеження прогресу та аналіз у Online Gantt

Хід виконання:

Етап 1. Оновлення статусу завдань.

Крок.1. Переконайтеся, що кожен член команди оновив статус своїх завдань (наприклад):

- «Збір вимог» завершено на 100%.
- «Проектування UX/UI» виконано на 70%.
- «Front-end розробка» розпочата, статус – 30%.

Крок 2. Оновіть тривалість і терміни для завдань, якщо виникли затримки.

Етап 2. Аналіз завантаження ресурсів.

Крок 1. Перейдіть до вкладки «Ресурси» та перевірте, чи кожен член команди має оптимальне завантаження.

Крок 2. Якщо один із членів команди перевантажений (зайнятість перевищує 100%), змініть розподіл роботи між іншими членами команди або відкоригуйте дедлайни завдань.

Крок 3. Визначте, чи є потенційні ризики для завершення проєкту через навантаження або затримки.

Етап 3. Налаштування звіту.

Крок 1. Створіть звіт по проєкту, в якому відобразіть:

- Завдання, що виконані вчасно.
- Проблеми зі строками (причини затримок).
- Поточний прогрес по кожному завданню та по всьому проєкту.

Крок 2. Експортуйте звіт у PDF або Excel для подальшого аналізу.

Етап 4. Сформувавши короткий висновок за результатами виконання завдання 2.

Контрольні запитання до розділу

1. Що таке Online Gantt і як він використовується для управління проектами?
2. Які основні функції Online Gantt для управління проектами?
3. Як створювати та налаштовувати нові проекти в Online Gantt?
4. Як побудувати і налаштувати діаграму Ганта у Online Gantt?
5. Які можливості Online Gantt надає для планування та управління ресурсами?
6. Як у Online Gantt відслідковується прогрес завдань і проекту?
7. Як у Online Gantt організовується командна співпраця та розподіл завдань?
8. Що таке віхи у Online Gantt і як вони допомагають у плануванні проектів?
9. Як використовувати залежності між завданнями у Online Gantt для кращого планування?
10. Як налаштувати повідомлення та нагадування в Online Gantt для управління термінами?
11. Які можливості пропонує Online Gantt для роботи з віддаленими командами?
12. Як використовувати шаблони в Online Gantt для швидкого старту нових проектів?
13. Як автоматизувати процеси в Online Gantt, щоб підвищити ефективність управління проектами?
14. Як TeamGantt допомагає в управлінні ризиками та змінами у проектах?
15. Як інтегрувати Online Gantt для роботи з іншими інструментами?
16. Як проводити аналіз продуктивності команди за допомогою звітів?
17. Як у Online Gantt управляти бюджетами та фінансовими аспектами проектів?
18. Як діаграма Ганта в Online Gantt допомагає прогнозувати терміни виконання завдань?
19. Які є опції для експорту та спільного використання даних проекту в Online Gantt?
20. Як Online Gantt забезпечує безпеку даних та управління правами доступу до проектів?

ТЕСТОВІ ЗАВДАННЯ ДЛЯ САМОКОНТРОЛЮ

1. Яку роль Jira відіграє в управлінні проєктами?

- а) система для автоматизації бухгалтерських операцій
- б) система для управління базами даних
- в) інструмент для управління проєктами та відстеження завдань
- г) програма для розробки мобільних додатків

2. Які основні функції Jira для управління проєктами?

- а) створення баз даних і звітів
- б) організація завдань, планування спринтів, відстеження прогресу
- в) оптимізація серверів та мережевого трафіку
- г) розробка мобільних додатків

3. Що таке Scrum і Kanban дошки в Jira, і в чому їх відмінність?

- а) Scrum орієнтований на тривалі проєкти, а Kanban – на короткострокові
- б) Scrum використовує спринти, а Kanban – візуалізує поточну роботу без спринтів
- в) Scrum використовується тільки для розробки програмного забезпечення, а Kanban – для будь-яких проєктів
- г) відмінностей між ними немає

4. Як у Jira організована структура проєктів, епік, завдань і підзадач?

- а) проєкти містять підпроєкти, які містять задачі та епіки
- б) завдання створюються з підзадач, які об'єднуються в епік
- в) епіки містять завдання, які можуть бути розділені на підзадачі
- г) підзадачі є головними елементами в проєкті

5. Як створювати і налаштовувати проєкти в Jira?

- а) вибирати відповідний шаблон, налаштовувати ролі і права, додавати учасників
- б) налаштовувати вручну через консольний інтерфейс
- в) створювати проєкт тільки через інтеграцію з іншими інструментами
- г) додати епіки і завдання без можливості їхньої зміни

6. Що таке Workflow у Jira, і як можна його налаштувати для команди?

- а) алгоритм автоматизації розробки
- б) послідовність статусів завдань, яку можна налаштувати для контролю процесів
- в) інструмент для моніторингу часу роботи
- г) звіт про виконані завдання

7. Як відслідковувати прогрес команди за допомогою дашбордів у Jira?

- а) використовуючи аналітичні модулі Jira для підрахунку часу
- б) створюючи дашборди з різними віджетами для візуалізації завдань і їхнього статусу
- в) відслідковуючи вручну через електронні таблиці
- г) перевіряючи кожне завдання окремо

8. Як у Jira використовуються спринти для управління проєктами в Scrum?

- а) для розподілу завдань на короткі відрізки часу (спринти) з метою швидкого виконання
- б) для створення довгострокових планів проєкту
- в) для складання річного звіту про роботу команди
- г) для оцінки ризиків у проєкті

9. Що таке backlog у Jira і як його управляти?

- а) це список завершених завдань
- б) це архів проєктів
- в) це список завдань, які ще не були виконані, їх можна пріоритетувати і планувати
- г) це набір звітів команди

10. Як у Jira проводиться оцінка завдань за допомогою Story Points?

- а) завдання оцінюються за часом, витраченим на виконання
- б) Story Points використовуються для оцінки складності завдань відносно одне одного
- в) кожне завдання автоматично отримує один бал
- г) оцінки завдань задаються випадковим чином

11. Для яких цілей Uspace використовується в управлінні проєктами?

- а) платформа для управління фінансами
- б) платформа для автоматизації маркетингових процесів
- в) інструмент для управління проєктами та організації співпраці в команді
- г) програма для налаштування мережевого обладнання

12. Які основні функції платформи Uspace для управління проєктами?

- а) пошук та обробка великих обсягів даних
- б) організація завдань, управління часом, відстеження прогресу
- в) моніторинг серверів та кібербезпека
- г) управління базами даних

13. Як у Uspasy організуються проєкти, завдання та підзадачі?

- а) завдання не підтримують підзадачі, лише етапи
- б) проєкти містять завдання, які можна розділяти на підзадачі
- в) підзадачі автоматично створюються для кожного завдання
- г) завдання можуть бути об'єднані в проєкти без можливості їх поділу

14. Як налаштовується робоча панель (дашборд) в Uspasy для відстеження прогресу?

- а) налаштовується автоматично, і не потребує змін
- б) користувачі можуть додавати віджети для відображення інформації про проєкти та завдання
- в) дашборд використовується тільки для перегляду завдань без можливості налаштувань
- г) панель відображає лише виконані завдання, інші опції відсутні

15. Як Uspasy дозволяє керувати командою та розподіляти завдання між учасниками?

- а) кожен учасник самостійно обирає завдання
- б) завдання автоматично розподіляються між учасниками системою
- в) адміністратор може призначати завдання конкретним учасникам команди
- г) завдання можна ділити лише між двома учасниками

16. Що таке таймлайн у Uspasy, і як він допомагає планувати проєкт?

- а) графік із зазначеними дедлайнами для проєктів і завдань
- б) час роботи кожного учасника команди
- в) список контактів команди для швидкого зв'язку
- г) журнал історії змін у проєкті

17. Як використовуються теги та категорії для організації завдань в Uspasy?

- а) теги використовуються для позначення завершених завдань
- б) теги і категорії допомагають групувати і фільтрувати завдання за пріоритетами або темами
- в) теги не використовуються в Uspasy
- г) категорії автоматично генеруються системою для кожного завдання

18. Як проводиться контроль дедлайнів і термінів у Uspasy?

- а) система не підтримує функцію відстеження термінів
- б) дедлайни можна налаштувати для кожного завдання, і система сповіщає про їх наближення
- в) дедлайни автоматично додаються системою без можливості їх зміни
- г) термінів у Uspasy немає

19. Яким чином monday.com використовують для управління проектами?

- а) система для управління базами даних
- б) інструмент для управління проектами, завданнями і командами
- в) платформа для створення веб-сайтів
- г) програма для налаштування серверів

20. Які основні функції платформи monday.com для управління проектами?

- а) управління кодовими базами
- б) створення і налаштування завдань, відстеження прогресу, комунікація в команді
- в) розробка програмного забезпечення
- г) моніторинг безпеки мереж

21. Як створювати та налаштовувати нові проекти в monday.com?

- а) платформа автоматично створює всі необхідні проекти
- б) користувачі можуть створювати нові проекти, вибираючи шаблони або налаштовуючи їх вручну
- в) проекти створюються лише через інтеграцію з іншими сервісами
- г) нові проекти не можна створювати, лише підзадачі

22. Як у monday.com налаштовувати та використовувати проектні дошки (boards)?

- а) дошки автоматично створюються системою без можливості налаштувань
- б) дошки можна налаштовувати для відображення завдань, статусів і етапів проекту
- в) дошки використовуються тільки для відстеження завершених завдань
- г) дошки не підтримуються в monday.com

23. Як monday.com підтримує методології управління проектами, такі як Agile або Scrum?

- а) платформа має шаблони для управління Agile і Scrum, включаючи спринти та дошки Kanban
- б) monday.com не підтримує ці методології
- в) лише через інтеграцію з іншими інструментами
- г) підтримується лише традиційний підхід до управління проектами

24. Як у monday.com налаштовується автоматизація робочих процесів?

- а) автоматизація можлива тільки через сторонні сервіси
- б) можна налаштовувати автоматичні правила для виконання повторюваних завдань та сповіщень
- в) автоматизація виконується вручну через консольний інтерфейс
- г) робочі процеси в monday.com не можна автоматизувати

25. Як використовувати monday.com для управління командою та розподілу завдань?

- а) завдання автоматично розподіляються системою між усіма учасниками команди
- б) адміністратор або керівник проєкту може призначати завдання конкретним учасникам або командам
- в) кожен учасник самостійно обирає завдання
- г) завдання можна призначати лише двом учасникам одночасно

26. Які типи віджетів доступні на дашбордах у monday.com для візуалізації проєктів?

- а) віджети для фінансового аналізу, але не для завдань
- б) віджети для відстеження статусу завдань, таймлайну, прогресу команди
- в) лише віджети для відображення календаря
- г) віджети не підтримуються у monday.com

27. Яким чином Smartsheet використовують для управління проєктами?

- а) програма для графічного дизайну
- б) платформа для управління проєктами, завданнями, автоматизацією і колаборацією команд
- в) інструмент для моніторингу безпеки серверів
- г) система для розробки програмного забезпечення

28. Які основні функції Smartsheet для управління проєктами?

- а) управління базами даних та розробка ПЗ
- б) створення проєктів, управління завданнями, автоматизація процесів, відстеження прогресу
- в) обробка зображень та відеомонтаж
- г) виключно фінансове управління

29. Як створювати і налаштовувати нові проєкти в Smartsheet?

- а) проєкти автоматично генеруються з зовнішніх даних
- б) проєкти створюються вручну або за допомогою шаблонів із можливістю гнучкого налаштування
- в) можна створювати тільки одне завдання на проєкт
- г) створення проєктів можливе лише через інтеграцію з іншими сервісами

30. Що таке робочий аркуш (sheet) у Smartsheet, і як він використовується для організації завдань?

- а) лист для відображення фінансових операцій
- б) таблиця, яка використовується для планування завдань, термінів тощо

- в) лист для відстеження комунікації з клієнтами
- г) інструмент для створення баз даних

31. Як у Smartsheet використовуються проєктні шаблони для прискорення роботи?

- а) Smartsheet не підтримує шаблони для проєктів
- б) проєктні шаблони дають змогу швидко налаштувати типові проєкти з попередньо визначеними структурами
- в) шаблони використовуються лише для управління фінансами
- г) шаблони автоматично створюють завдання з випадковим вмістом

32. Як Smartsheet підтримує методології управління проєктами, такі як Agile або Scrum?

- а) Smartsheet не підтримує ці методології
- б) платформа підтримує шаблони та функціонал для роботи з Agile і Scrum, такі як спринти та борди
- в) ці методології можливі лише через зовнішні плагіни
- г) підтримується тільки традиційний метод управління проєктами

33. Як відслідковується прогрес проєкту за допомогою таймлайну (Gantt chart) у Smartsheet?

- а) таймлайн використовується тільки для завершених завдань
- б) Smartsheet пропонує Gantt chart для відстеження термінів завдань, зв'язків між ними та прогресу
- в) Gantt chart використовується лише для фінансового планування
- г) Gantt chart не підтримується Smartsheet

34. Яким чином Online Gantt використовується для управління проєктами?

- а) платформа для графічного дизайну
- б) система для розробки мобільних додатків
- в) інструмент для управління проєктами з акцентом на діаграми Ганта
- г) програма для керування фінансовими операціями

35. Які основні функції Online Gantt для управління проєктами?

- а) моніторинг серверів та робота з базами даних
- б) планування проєктів, створення діаграм Ганта, управління ресурсами, командна співпраця
- в) обробка зображень та відео
- г) виключно відстеження витрат

36. Як створювати та налаштовувати нові проєкти в Online Gantt?

- а) проєкти генеруються автоматично з бази даних
- б) користувачі можуть створювати нові проєкти вручну або використовувати шаблони для швидкого налаштування
- в) можливо лише редагування існуючих проєктів
- г) нові проєкти створюються через інтеграцію з зовнішніми інструментами

37. Як побудувати і налаштувати діаграму Ганта у Online Gantt?

- а) діаграма Ганта будується автоматично на основі всіх завдань
- б) діаграми Ганта створюються вручну, з можливістю налаштування залежностей, термінів і тривалості завдань
- в) можна створювати діаграми лише для фінансових операцій
- г) діаграма Ганта використовується тільки для звітування

38. Які можливості Online Gantt надає для планування ресурсів та управління ними?

- а) TeamGantt не підтримує планування ресурсів
- б) можливість планувати ресурси та призначати їх на конкретні завдання для уникнення перевантаження
- в) планування ресурсів можливе тільки через сторонні плагіни
- г) ресурси автоматично розподіляються між завданнями

39. Як у Online Gantt відслідковується прогрес завдань і проєкту?

- а) прогрес завдань можна відстежувати тільки за допомогою зовнішніх сервісів
- б) Online Gantt дозволяє візуально відображати прогрес завдань у діаграмах Ганта та переглядати статуси завдань
- в) прогрес проєкту можна відстежувати тільки в текстовому вигляді
- г) прогрес завдань не відслідковується

40. Як використовувати залежності між завданнями у Online Gantt для кращого планування?

- а) залежності між завданнями не підтримуються
- б) залежності можна налаштовувати для відображення послідовності виконання завдань і зв'язків між ними
- в) залежності між завданнями задаються автоматично
- г) можна встановлювати залежності тільки між завершеними завданнями

ГЛОСАРІЙ

А

Activity log (*Журнал активності*). Історія дій, які виконувалися у проєкті або над завданням.

Activity-based scheduling (*Планування на основі діяльності*). Підхід до планування, що фокусується на визначенні діяльності, необхідної для досягнення цілей.

Adaptive planning (*Адаптивне планування*). Метод планування, який дозволяє змінювати план у відповідь на нові умови, обставини чи ризики.

Admin permissions (*Адміністраторські дозволи*). Налаштування прав доступу до різних функцій і даних у межах системи управління проєктами.

Agile board (*Аджайл-дошка*). Візуальний інструмент для управління роботою в рамках гнучкої методології, де завдання представлені у форматі карток.

Agile estimation (*Аджайл-оцінка*). Метод оцінювання завдань в Agile на основі їхньої складності чи обсягу роботи, наприклад, у балах.

Agile methodology (*Аджайл-методологія*). Гнучкий підхід до управління проєктами, що спрямований на швидкі результати через ітеративну розробку.

Agile roadmap (*Аджайл-дорожня карта*). Високорівневий план, що визначає ітерації, етапи та цілі в рамках гнучкої методології.

Agile (*Аджайл*). Гнучка методологія управління проєктами, що орієнтована на ітеративний підхід до розробки, співпрацю команди та постійну адаптацію до змін.

Approval process (*Процес затвердження*). Процедура, яка визначає порядок розгляду та затвердження важливих змін у проєкті.

Approval requests (*Запити на затвердження*). Система, що дозволяє надсилати запити на затвердження змін чи рішень у проєкті.

Approval workflow (*Робочий процес затвердження*). Автоматизований процес для перевірки та затвердження завдань, документів чи змін у проєкті.

Archived projects (*Архівовані проєкти*). Проєкти, завершені або закриті, збережені у системі для довідки чи аналізу.

Archived tasks (*Архівовані завдання*). Завдання, які були завершені або закриті, але залишаються доступними для довідки.

Automated alerts (*Автоматичні сповіщення*). Система повідомлень, яка інформує користувачів про майбутні строки, ризики або зміни в реальному часі.

Automated status updates (*Автоматичні оновлення статусу*). Система, яка автоматично повідомляє про зміни статусу завдань або проєкту.

Automation (Автоматизація). Використання технологій для виконання повторюваних завдань без участі людини.

Automation rules (Правила автоматизації). Попередньо налаштовані сценарії, що виконують завдання автоматично на основі певних умов, наприклад, зміна статусу завдання або надсилання сповіщення.

B

Backlog (Беклог). Список завдань, функцій або вимог, що мають бути реалізовані у межах проекту. Зазвичай використовується в Agile-проектах.

Baseline (Базовий показник). Початковий план, який використовується як основа для порівняння фактичного виконання проекту.

Baseline comparison (Порівняння з базовим показником). Функція для аналізу прогресу проекту шляхом порівняння поточного стану із запланованим.

Baseline metrics (Базові метрики). Початкові показники, використані для порівняння та оцінки прогресу проекту.

Baseline shift (Зміщення базових показників). Відхилення фактичних результатів проекту від початкових планових показників.

Board (Дошка). Візуальне представлення проектних завдань у форматі колонок із картками, які рухаються між статусами (наприклад, «Виконується», «На перевірці», «Завершено»).

Budget allocation (Розподіл бюджету). Процес виділення фінансових ресурсів на конкретні завдання або етапи проекту.

Budget overrun (Перевищення бюджету). Перевищення запланованих витрат на виконання проекту.

Budget tracking (Відстеження бюджету). Функція для моніторингу витрат та фінансового стану проекту.

Bug tracking (Відстеження помилок). Система для реєстрації, моніторингу та виправлення програмних помилок або дефектів.

Burndown chart (Графік вигорання). Графік, що показує, скільки завдань або роботи залишилося до завершення спринту чи проекту.

Burnup chart (Діаграма зростання обсягу роботи). Графік, який відображає, скільки роботи було виконано в проекті за певний період часу.

C

Calendar integration (Інтеграція календаря). Функція, що дозволяє синхронізувати проектні завдання з зовнішніми календарями, наприклад, Google Calendar.

Calendar view (*Перегляд календаря*). Функція для візуалізації завдань та подій у форматі календаря.

Capacity planning (*Планування потужностей*). Процес визначення доступності ресурсів для виконання завдань у межах проєкту.

Capacity utilization rate (*Коефіцієнт використання потужностей*). Показник, що відображає, наскільки ефективно використовуються доступні ресурси.

Change control board (*Рада з контролю змін*). Група, відповідальна за розгляд і затвердження змін у проєкті.

Change requests (*Запити на зміни*). Формальні пропозиції внести зміни до початкового плану проєкту.

Cloud storage (*Хмарне сховище*). Функція для збереження проєктних даних у віддаленому сховищі з доступом через інтернет.

Collaboration (*Співпраця*). Процес взаємодії між членами команди, що спрямований на досягнення спільної мети.

Collaboration dashboard (*Панель співпраці*). Інструмент, що об'єднує ключові функції для командної роботи: завдання, коментарі та спільне використання файлів.

Collaboration tools (*Інструменти співпраці*). Програми чи функції, що дозволяють командам ефективно взаємодіяти, наприклад, чат, коментарі чи обмін файлами.

Collaborative planning (*Спільне планування*). Процес розробки плану проєкту за участю всіх членів команди.

Collaborative task management (*Спільне управління завданнями*). Процес, за допомогою якого команда працює над завданнями одночасно, обмінюючись оновленнями в реальному часі.

Collaborative whiteboard (*Спільна дошка*). Інтерактивний інструмент, що дозволяє членам команди спільно працювати над візуалізацією ідей і даних.

Comments (*Коментарі*). Примітки, що додаються до завдань для полегшення спілкування між членами команди.

Completion percentage (*Відсоток виконання*). Числове значення, яке відображає прогрес виконання завдання або проєкту.

Continuous improvement (*Безперервне вдосконалення*). Філософія управління, спрямована на постійне підвищення якості, ефективності та результатів роботи.

Cost tracking (*Відстеження витрат*). Функція для моніторингу витрат, пов'язаних із виконанням проєкту, у реальному часі.

Critical chain method (*Метод критичного ланцюга*). Підхід до управління проєктами, який фокусується на усуненні перевантажень і оптимізації ресурсів.

Critical path (*Критичний шлях*). Найдовша послідовність залежних завдань, яка визначає мінімальний час для завершення проєкту.

Critical path analysis (*Аналіз критичного шляху*). Процес визначення послідовності завдань, що мають найбільший вплив на строки завершення проєкту.

Critical task (*Критичне завдання*). Завдання, яке безпосередньо впливає на терміни або результати проєкту.

Cross-functional team (*Крос-функціональна команда*). Група, що складається з членів із різними спеціалізаціями, які співпрацюють для досягнення спільної мети.

Cross-functional team. Команда, що складається з учасників із різними навичками, які працюють над досягненням спільної мети.

Cross-project dependencies (*Залежності між проєктами*). Взаємозв'язки між завданнями або етапами різних проєктів, які впливають один на одного.

Cross-project reporting (*Звітування по кількох проєктах*). Функція створення звітів, які охоплюють дані з кількох проєктів.

Cross-team collaboration (*Співпраця між командами*). Процес спільної роботи між різними командами або департаментами для досягнення спільних цілей.

Custom dashboards (*Індивідуальні панелі*). Налаштовані панелі управління, що відображають ключові метрики та дані проєкту відповідно до потреб користувача.

Custom fields (*Користувацькі поля*). Додаткові поля, які можна створювати для налаштування інформації про завдання, наприклад, тип, статус чи категорію.

Custom templates (*Користувацькі шаблони*). Попередньо створені структури завдань, які можна налаштувати для конкретних проєктів чи процесів.

Custom templates. Шаблони придатні до налаштування для проєктів або завдань, які відповідають специфічним потребам організації.

Custom workflows (*Користувацькі робочі процеси*). Налаштовані послідовності завдань та статусів, які відповідають унікальним вимогам проєкту.

Cycle time (*Час циклу*). Проміжок часу, необхідний для завершення одного циклу роботи в процесі.

D

Dashboard (*Інформаційна панель*). Централізована панель, яка відображає ключові показники та дані про проєкт у реальному часі.

Dashboard customization (*Налаштування панелі*). Процес адаптації панелі управління відповідно до потреб користувача чи проєкту.

Data export (*Експорт даних*). Функція, що дозволяє вивантажувати дані з платформи у вигляді файлів для аналізу чи подальшої роботи. Наприклад, в формат Excel чи PDF.

Data import (*Імпорт даних*). Функція, що дозволяє завантажувати зовнішні дані у платформу для подальшої роботи.

Deliverables (*Результати*). Конкретні продукти, послуги чи результати, які створюються в рамках проєкту.

Dependencies (*Залежності*). Взаємозв'язок між завданнями або етапами, що впливає на хронологію проєкту.

Dependencies tracker (*Відстеження залежностей*). Функція, що дозволяє моніторити та керувати залежностями між завданнями у проєкті.

Dependency (*Залежність*). Взаємозв'язок між завданнями, при якому виконання одного завдання залежить від завершення іншого.

Dependency conflicts (*Конфлікти залежностей*). Ситуація, коли виконання одного завдання блокується через незавершеність іншого.

Dependency management (*Управління залежностями*). Процес моніторингу та контролю взаємозв'язків між завданнями.

Dependency mapping (*Карта залежностей*). Візуалізація залежностей між завданнями або компонентами проєкту.

Drag-and-drop gantt Chart (*Діаграма Ганта з функцією перетягування*). Інтерактивний інструмент, що дозволяє змінювати строки завдань простим перетягуванням елементів.

Drag-and-drop interface (*Інтерфейс «перетягни й відпусти»*). Зручний користувацький інтерфейс, що дозволяє змінювати розташування елементів простим перетягуванням.

Drag-and-drop milestones (*Майлстоуни з функцією перетягування*). Функція, що дозволяє змінювати строки важливих етапів шляхом їх перетягування на діаграмі.

Drag-and-drop scheduling (*Планування методом перетягування*). Інтуїтивний спосіб змінювати строки чи розташування завдань у графіку за допомогою миші.

Dynamic filters (*Динамічні фільтри*). Інструменти для швидкого пошуку та сортування завдань за різними критеріями.

E

Epic (*Епік*). Велика задача або мета, яка складається з кількох дрібніших завдань (історій користувача).

Escalation workflow (*Робочий процес ескалації*). Процес передачі складного або критичного завдання на вищий рівень управління.

External stakeholders (Зовнішні зацікавлені сторони). Особи або організації, які мають інтерес у проєкті, але не входять до команди, наприклад, клієнти чи постачальники.

F

Feedback loops (Петлі зворотного зв'язку). Процес постійного збору та впровадження зворотного зв'язку для вдосконалення проєкту.

File attachment (Прикріплення файлів). Функція для додавання документів, зображень чи інших файлів безпосередньо до завдання або проєкту.

G

Gantt Chart (Діаграма Ганта). Графік, що представляє завдання проєкту у вигляді горизонтальних смуг, розташованих відповідно до часової шкали.

Gantt dependency Arrows (Стрілки залежностей у діаграмі Ганта). Графічне зображення взаємозв'язків між завданнями на діаграмі Ганта.

Goal alignment (Вирівнювання цілей). Процес узгодження цілей проєкту із загальною стратегією організації.

H

Historical data (Історичні дані). Дані про минулі проєкти чи завдання, які використовуються для аналізу або прогнозування.

Historical data analysis (Аналіз історичних даних). Оцінка даних із попередніх проєктів для прогнозування або покращення поточних процесів.

I

Integration (Інтеграція). Процес об'єднання окремих компонентів, систем або даних у єдину цілісну структуру, яка дозволяє їм взаємодіяти та функціонувати разом. У контексті інформаційних технологій інтеграція передбачає з'єднання різних програмних систем, баз даних, платформ чи процесів для спрощення обміну даними, підвищення ефективності та зменшення дублювання завдань.

Integration API (Інтеграційний API). Інструмент для підключення сторонніх додатків до систем управління проєктами для автоматизації процесів.

Integration APIs (API для інтеграції). Інтерфейси програмування, що дозволяють підключати зовнішні сервіси або додатки до платформи.

Issue (Проблема, завдання). Будь-яке завдання, помилка чи запит, зареєстроване в системі управління проєктами. У Jira це основний елемент роботи.

Issue tracking (Відстеження проблем). Система для фіксації, моніторингу та вирішення проблем, які виникають у проєкті.

Iteration (Ітерація). Повторення циклу виконання завдань для поступового досягнення результату.

Iteration goals (Цілі ітерації). Конкретні результати, які команда планує досягти до кінця ітерації.

Iteration planning (Планування ітерації). Процес визначення завдань, які будуть виконані під час наступної ітерації в рамках Agile.

Iterative development (Ітеративна розробка). Метод виконання проєктів, який передбачає поступове вдосконалення продукту через повторювані цикли.

Iterative process (Ітеративний процес). Методологія, що передбачає виконання завдань через послідовні цикли з поступовим поліпшенням результату.

Iterative risk assessment (Ітеративна оцінка ризиків). Постійний перегляд ризиків на кожному етапі проєкту для забезпечення актуальності плану управління ризиками.

Iterative task management (Ітеративне управління завданнями). Підхід до управління завданнями, що передбачає їх покращення через повторні цикли.

Iterative workflow (Ітеративний робочий процес). Процес виконання завдань через повторення етапів із поступовим удосконаленням.

К

Kanban (Канбан). Метод управління роботою, що передбачає використання дошки з картками для візуалізації прогресу завдань.

Kanban board (Канбан-дошка). Візуальний інструмент управління завданнями, що використовує картки для позначення етапів виконання завдань.

Kanban swimlanes (Канбан-смуги). Горизонтальні розділи на дошці Kanban, які допомагають групувати завдання за певними критеріями.

Key deliverables (Ключові результати). Конкретні та вимірювані результати, які мають бути досягнуті в межах проєкту.

Key performance indicators (KPI) (Ключові показники ефективності). Метрики, що використовуються для оцінки успіху проєкту або завдань.

KPI dashboard (Панель KPI). Візуальний інструмент для моніторингу ключових показників ефективності в реальному часі.

KPI tracking (Відстеження ключових показників ефективності). Система для моніторингу прогресу щодо ключових метрик проєкту.

L

Lead time (Час виконання). Проміжок часу між створенням завдання і його завершенням.

Live collaboration tools (Інструменти для живої співпраці). Функції, які дозволяють команді працювати над документами чи завданнями одночасно в реальному часі.

Load testing (Тестування навантаження). Процес оцінки продуктивності системи за умов пікового навантаження.

M

Meeting scheduler (Планувальник зустрічей). Інструмент для організації зустрічей чи нарад із врахуванням доступності членів команди.

Milestone (Контрольна точка). Ключова подія або етап у проєкті, що позначає завершення важливого етапу.

Milestone dependency (Залежність від контрольної точки). Зв'язок між етапами чи важливими подіями у проєкті.

Milestone reporting (Звітність за контрольними точками). Форма звітів, яка підсумовує досягнення ключових етапів у проєкті.

Milestone tracking (Відстеження контрольних точок). Функція для моніторингу важливих етапів або подій у проєкті.

Multi-project management (Управління кількома проєктами). Здатність управляти кількома проєктами одночасно в одній системі.

Multi-project scheduling (Планування кількох проєктів). Функція, що дозволяє координувати графіки кількох проєктів одночасно.

Multi-tiered projects (Багаторівневі проєкти). Проєкти, які розділені на кілька рівнів чи фази з різними цілями.

Multi-user access (Доступ для декількох користувачів). Функція, що дозволяє кільком учасникам одночасно працювати над одним проєктом.

N

Notification system (Система сповіщень). Функція, яка інформує користувачів про зміни, нові завдання або важливі події в проєкті.

Notifications (Сповіщення). Автоматичні повідомлення про зміни у проєкті, нові завдання або завершення етапів.

Notifications settings (Налаштування сповіщень). Функція для керування типами й частотою отримуваних сповіщень.

O

Offline mode (Офлайн-режим). Режим роботи платформи, який дозволяє вносити зміни до даних без доступу до інтернету, із подальшою синхронізацією.

Online collaboration (Онлайн-співпраця). Функція, що дозволяє учасникам команди працювати разом у реальному часі через інтернет.

Online document sharing (Онлайн-обмін документами). Функція, що дозволяє командам спільно використовувати документи через платформу.

Outcome-based planning (Планування на основі результатів). Стратегія планування, що фокусується на досягненні конкретних цілей і результатів.

P

Parallel workflows (Паралельні робочі процеси). Механізм, що дозволяє виконувати кілька завдань або процесів одночасно.

Performance benchmarking (Оцінка продуктивності за еталонами). Процес порівняння результатів проєкту із заздалегідь встановленими стандартами.

Performance dashboard (Панель ефективності). Інтерфейс, який надає зведену інформацію про ключові метрики проєкту.

Performance indicators Dashboard (Панель ключових показників ефективності). Інструмент для моніторингу та відображення КРІ у режимі реального часу.

Performance reports (Звіти про ефективність). Аналітичні документи, що відображають продуктивність команди чи прогрес проєкту.

Permissions (Дозволи користувачів). Механізм, який регулює доступ користувачів до різних функцій чи даних системи.

Permissions management (Управління дозволами). Модуль для налаштування доступу до функцій та даних у проєкті.

Personal dashboard (Персональна панель). Індивідуальний інтерфейс для кожного користувача, де відображаються його завдання та ключові показники.

Phase completion (Завершення етапу). Завершення певного етапу або частини проєкту відповідно до запланованого графіку.

Portfolio (Портфель). Збірка кількох проєктів, якими управляють одночасно для досягнення стратегічних цілей.

Portfolio dashboard (Панель управління портфелем). Інструмент для огляду та аналізу кількох проєктів одночасно.

Portfolio management (Управління портфелем проєктів). Процес управління групою проєктів, спрямований на досягнення стратегічних цілей організації.

Predecessor tasks (Попередні завдання). Завдання, які повинні бути завершені перед початком іншого завдання.

Priority (Пріоритет). Рівень важливості завдання в порівнянні з іншими завданнями проєкту.

Priority levels (Рівні пріоритету). Класифікація завдань за ступенем їх важливості, наприклад, «Високий», «Середній», «Низький».

Progress bar (Індикатор прогресу). Графічне відображення, що показує відсоткове виконання завдання або проєкту.

Progress monitoring (Моніторинг прогресу). Процес спостереження за виконанням завдань та досягненням цілей.

Project alignment (Вирівнювання проєкту). Процес забезпечення узгодженості проєкту з бізнес-цілями та потребами зацікавлених сторін.

Project audit (Аудит проєкту). Процес оцінки проєкту на предмет відповідності початковому плану, бюджету та строкам.

Project charter (Паспорт проєкту). Документ, який формально визначає цілі, учасників і межі проєкту.

Project constraints (Обмеження проєкту). Фактори, які обмежують виконання проєкту, такі як строки, бюджет і ресурси.

Project delay analysis (Аналіз затримок у проєкті). Вивчення причин і впливу затримок на загальний план виконання проєкту.

Project dependencies (Залежності проєкту). Взаємозв'язки між завданнями або етапами, які впливають на хронологію виконання проєкту.

Project health metrics (Метрики стану проєкту). Ключові показники, що відображають загальний стан проєкту, наприклад, продуктивність, строки або витрати.

Project manager (Менеджер проєкту). Особа, відповідальна за планування, виконання та завершення проєкту.

Project plan (План проєкту). Документ або структура, яка визначає цілі, етапи, завдання та часові рамки проєкту.

Project prioritization (Пріоритизація проєктів). Процес визначення, які проєкти або завдання мають найвищий пріоритет.

Project repository (Сховище проєкту). Централізоване місце зберігання всіх документів, файлів і даних, пов'язаних із проєктом.

Project scope (Обсяг проєкту). Опис усіх цілей, завдань, результатів і меж, які охоплює проєкт.

Project templates (Шаблони проєктів). Попередньо створені структури для запуску нових проєктів з уже заданими налаштуваннями та завданнями.

Project tracking (Моніторинг проєкту або його відстеження). Процес постійного збору, аналізу та оновлення інформації про стан реалізації проєкту з метою контролю прогресу щодо поставлених цілей, термінів, бюджету та ресурсів.

R

Real-time analytics (Аналітика в реальному часі). Миттєвий аналіз даних проєкту для прийняття рішень без затримок.

Real-time collaboration (Співпраця в реальному часі). Можливість команд працювати над одним завданням або документом одночасно, отримуючи оновлення миттєво.

Real-time task updates (Оновлення завдань у реальному часі). Функція, яка автоматично синхронізує зміни у завданнях між усіма користувачами.

Real-time updates (Оновлення в реальному часі). Функція, яка забезпечує миттєве відображення змін у проєкті для всіх учасників.

Recurring tasks (Повторювані завдання). Завдання, які автоматично створюються у визначений проміжок часу, наприклад, щотижня чи щомісяця.

Reporting (Звітність). Модуль або функція, яка дозволяє створювати звіти про прогрес, продуктивність та виконання завдань.

Reporting templates (Шаблони звітів). Попередньо створені структури для швидкого створення звітів.

Resource allocation (Розподіл ресурсів або управління ресурсами). Процес планування та управління людськими, фінансовими чи матеріальними ресурсами для виконання проєкту.

Resource allocation chart (Діаграма розподілу ресурсів). Графік, що показує рівень завантаження ресурсів у проєкті.

Resource calendar (Календар ресурсів). Інструмент для відстеження доступності ресурсів у визначені дати.

Resource conflict (Конфлікт ресурсів). Ситуація, коли один і той самий ресурс потрібен для виконання кількох завдань одночасно, що може призвести до затримок.

Resource contingency (Резерв ресурсів). Додаткові ресурси, що зберігаються для використання у випадку непередбачених обставин.

Resource forecasting (Прогнозування ресурсів). Оцінка майбутньої потреби в ресурсах для успішного виконання проєкту.

Resource management (Управління ресурсами). Процес планування, організації та моніторингу ресурсів проєкту.

Resource overload (*Перевантаження ресурсів*). Ситуація, коли ресурс (людина чи обладнання) має більше завдань, ніж може виконати за визначений час.

Resource planning (*Планування ресурсів*). Процес прогнозування та розподілу ресурсів для виконання завдань.

Resource pool (*Ресурсний пул*). Сукупність усіх ресурсів, доступних для використання у межах проекту.

Resource sharing (*Спільне використання ресурсів*). Процес розподілу ресурсів між кількома проектами або командами.

Resource utilization rate (*Рівень використання ресурсів*). Показник, який відображає, наскільки ефективно використовуються доступні ресурси у проекті.

Retrospective meeting (*Ретроспективна зустріч*). Зустріч у рамках Agile-методології для оцінки успіхів і визначення шляхів покращення процесів.

Risk assessment (*Оцінка ризиків*). Процес аналізу потенційних ризиків та розробки стратегій для їхнього уникнення чи мінімізації.

Risk contingency plan (*План резервів на ризики*). Резервний план для управління потенційними ризиками, які можуть вплинути на проект.

Risk exposure (*Ризикована вразливість*). Оцінка впливу ризиків на загальний результат проекту.

Risk management (*Управління ризиками*). Процес ідентифікації, оцінки та мінімізації ризиків, які можуть вплинути на успішність проекту.

Risk management plan (*План управління ризиками*). Документ, який визначає стратегії для виявлення, оцінки та управління ризиками у проекті.

Risk mitigation (*Зниження ризиків*). Дії, спрямовані на зменшення негативного впливу потенційних ризиків на проект.

Risk mitigation plan (*План зменшення ризиків*). Стратегія для зменшення впливу або ймовірності настання ризиків.

Risk probability matrix (*Матриця ймовірності ризику*). Інструмент для оцінки ризиків за їхньою ймовірністю та впливом на проект.

Risk register (*Реєстр ризиків*). Документ, що містить ідентифіковані ризики проекту, їхню оцінку та стратегії управління.

Risk scoring (*Оцінка ризиків*). Процес присвоєння числового значення ризикам для визначення їхньої важливості та пріоритетності.

Roadmap (*Дорожня карта*). Візуальний план, що відображає основні етапи та завдання проекту на часовій шкалі.

Roadmap planning (*Планування дорожньої карти*). Процес створення стратегічного плану, що визначає ключові етапи та строки досягнення цілей.

Role-based access control (Управління доступом на основі ролей). Модель, яка дозволяє обмежувати доступ до функцій платформи залежно від ролі користувача.

S

Scenario planning (Сценарне планування). Метод, що використовує моделювання альтернативних сценаріїв розвитку подій у проєкті.

Scheduling conflicts (Конфлікти у плануванні). Ситуація, коли два або більше завдань заплановані на той самий час і ресурс.

Shared workspace (Спільний робочий простір). Цифровий простір, де команда має доступ до спільних документів, завдань та інструментів для співпраці.

SLA (Угода про рівень обслуговування). Документ, який визначає очікуваний рівень послуг або показників виконання завдань.

SLA management (Управління угодами про рівень обслуговування). Контроль виконання умов, зазначених у угодах про рівень обслуговування (SLA), для досягнення визначених стандартів якості.

Sprint (Спринт). Короткий цикл розробки (зазвичай 1-4 тижні), протягом якого команда виконує певний набір завдань.

Sprint Backlog (Беклог спринту). Список завдань, запланованих для виконання в межах одного спринту в Agile-процесі.

Sprint planning (Планування спринту). Процес визначення цілей та завдань для короткого циклу роботи (спринту) в Agile.

Sprint Retrospective (Ретроспектива спринту). Зустріч команди Scrum, яка проводиться після завершення кожного спринту з метою аналізу процесу роботи. Під час цієї події команда обговорює, що було зроблено добре, що потребує покращення та які конкретні дії варто вжити, щоб підвищити ефективність у наступному спринті.

Sprint velocity (Швидкість спринту). Кількість роботи, яку команда може виконати протягом одного спринту в Agile-процесі.

Stakeholder (Зацікавлена сторона). Особа або група, що має інтерес у виконанні проєкту та може впливати на його результати.

Stakeholder engagement (Залучення зацікавлених сторін). Процес встановлення комунікації та співпраці з особами або групами, які мають інтерес у проєкті.

Stakeholder management (Управління зацікавленими сторонами). Процес взаємодії із зацікавленими сторонами проєкту для забезпечення їхньої підтримки та участі.

Status labels (*Мітки статусу*). Короткі текстові або кольорові позначення, які відображають стан завдання, наприклад, «У процесі», «Завершено».

Subtask (*Підзавдання*). Дрібніші завдання, що належать до основного завдання і виконуються в межах його реалізації.

T

Tags (*Мітки*). Ключові слова чи категорії, які використовуються для групування та фільтрації завдань.

Task (*Завдання*). Окремий елемент роботи, який необхідно виконати в рамках проєкту.

Task archive (*Архів завдань*). Місце зберігання завершених або неактуальних завдань для збереження історії проєкту.

Task archiving (*Архівування завдань*). Функція для перенесення завершених завдань у архів для збереження історії.

Task assignment (*Призначення завдань*). Процес призначення завдання конкретному члену команди або групі для виконання.

Task auto-assignment (*Автоматичне призначення завдань*). Функція, яка автоматично призначає завдання виконавцям відповідно до їхніх ролей чи доступності.

Task checklist (*Контрольний список завдань*). Список підзавдань або кроків, які мають бути виконані для завершення основного завдання.

Task comments (*Коментарі до завдань*). Можливість додавати примітки чи повідомлення до завдання для полегшення комунікації в команді.

Task comments history (*Історія коментарів до завдань*). Журнал усіх коментарів, залишених до конкретного завдання, для відстеження обговорень.

Task delegation (*Делегування завдань*). Процес передачі відповідальності за виконання завдання іншому члену команди.

Task dependencies (*Залежності завдань*). Зв'язки між завданнями, які визначають порядок їх виконання.

Task dependencies visualization (*Візуалізація залежностей завдань*). Графічне представлення взаємозалежностей між завданнями, яке допомагає уникнути конфліктів у плануванні.

Task dependency chain (*Ланцюг залежностей завдань*). Послідовність взаємопов'язаних завдань, яка визначає порядок їх виконання.

Task due date (*Крайній термін виконання завдання*). Дата, до якої завдання має бути завершено.

Task filtering (*Фільтрація завдань*). Функція, яка дозволяє швидко знаходити завдання за певними критеріями, такими як статус або виконавець.

Task history (Історія завдань). Хронологія змін, виконаних у завданні, включно з оновленнями статусу, термінів та виконавців.

Task labels (Мітки завдань). Теги або категорії, які додаються до завдань для їх швидкого сортування та ідентифікації.

Task lifecycle (Життєвий цикл завдання). Усі етапи, через які проходить завдання: створення, виконання, завершення та архівування.

Task overlap (Перекриття завдань). Ситуація, коли кілька завдань виконуються паралельно, що може створювати конфлікти ресурсів.

Task priority (Пріоритет завдань). Визначення важливості завдань у порівнянні з іншими елементами проєкту.

Task reassignment (Перепризначення завдання). Зміна виконавця завдання на іншого члена команди.

Task rescheduling (Перепланування завдань). Процес зміни строків виконання завдань у проєкті.

Task rollup (Агрегація завдань). Функція, яка об'єднує статус підзавдань для відображення загального прогресу основного завдання.

Task sorting (Сортування завдань). Функція, що дозволяє впорядковувати завдання за певними критеріями, наприклад, за датою чи пріоритетом.

Task subdivision (Поділ завдання). Розбиття завдання на менші підзавдання для більш зручного управління.

Task synchronization (Синхронізація завдань). Процес узгодження завдань між різними платформами чи командами для забезпечення актуальності даних.

Task tracker (Трекер завдань). Інструмент для відстеження виконання завдань у реальному часі.

Team calendar (Календар команди). Загальний календар, у якому відображаються завдання, зустрічі та ключові події проєкту.

Team capacity (Пропускна здатність команди). Кількість роботи, яку команда може виконати за певний період часу.

Team collaboration (Командна співпраця). Процес спільної роботи членів команди над виконанням завдань і досягненням цілей проєкту.

Team collaboration tools (Інструменти співпраці команди). Платформи чи програми, що сприяють взаємодії членів команди, такі як чати чи дошки завдань.

Team member workload (Завантаженість учасників команди). Рівень зайнятості членів команди завданнями проєкту.

Team permissions (Дозволи команди). Система контролю доступу, яка визначає, які дії можуть виконувати окремі члени команди на платформі.

Team utilization rate (Рівень використання команди). Метрика, яка показує, наскільки ефективно використовуються члени команди у проєкті.

Template library (Бібліотека шаблонів). Збірка готових шаблонів для різних типів проєктів чи процесів.

Time blocking (Блокування часу). Метод управління часом, який передбачає виділення конкретних часових інтервалів для виконання завдань.

Time budgeting (Бюджетування часу). Процес розподілу часу на виконання завдань або етапів проєкту.

Time buffer (Часовий буфер). Запас часу, доданий до строків для компенсації потенційних затримок.

Time estimates (Оцінки часу). Прогнозування часу, необхідного для виконання завдання або проєкту.

Time estimation. Попереднє визначення, скільки часу знадобиться для виконання конкретного завдання або етапу проєкту.

Time estimation. Процес визначення кількості часу, необхідного для виконання завдання.

Time logs (Журнали часу). Записи часу, витраченого на виконання конкретного завдання або етапу.

Time overrun (Перевищення строків). Ситуація, коли завдання або проєкт не завершуються у встановлений час.

Time tracking (Відстеження часу). Функціонал для моніторингу кількості часу, витраченого на виконання конкретних завдань.

Timeboxing (Обмеження часу). Техніка управління часом, що передбачає встановлення фіксованого періоду для виконання конкретної діяльності.

Timeline (Хронологія). Часова шкала, яка показує початок і завершення завдань проєкту.

Timeline view (Хронологічний перегляд). Режим відображення завдань у вигляді часової шкали, що дозволяє бачити взаємозв'язки та дедлайни.

U

User permissions (Дозволи користувачів). Налаштування прав доступу до різних частин проєкту для окремих членів команди.

User roles (Ролі користувачів). Призначення ролей (наприклад, адміністратор, виконавець) для визначення обсягу доступу до функцій проєкту.

User story (Історія користувача). Короткий опис функціоналу або потреби з точки зору кінцевого користувача.

V

Version control (*Контроль версій*). Механізм для відстеження змін у файлах чи документах, що дозволяє зберігати історію редагувань.

Visualization tools (*Інструменти візуалізації*). Функції, які надають графічне представлення даних, наприклад, діаграми, графіки чи канбан-дошки.

W

Work allocation (*Розподіл роботи*). Процес розподілу завдань між членами команди відповідно до їхньої доступності або компетенцій.

Work breakdown structure (WBS) (*Декомпозиція робіт*). Ієрархічна структура, яка розбиває проєкт на менші та керовані елементи.

Work distribution chart (*Діаграма розподілу роботи*). Графічний інструмент, що показує, як розподілена робота між членами команди.

Workflow (*Робочий процес*). Послідовність кроків або етапів, через які проходить завдання від початку до завершення.

Workflow automation (*Автоматизація робочого процесу*). Процес автоматизації завдань або повторюваних дій у рамках робочого процесу.

Workflow optimization (*Оптимізація робочого процесу*). Процес удосконалення робочих процесів для підвищення ефективності та продуктивності команди.

Workflow scalability (*Масштабованість робочого процесу*). Здатність системи підтримувати ефективність навіть при збільшенні обсягу завдань чи користувачів.

Workflow status (*Статус робочого процесу*). Поточний стан завдання у проєкті, наприклад, «Заплановано», «Виконується», «Завершено».

Workspace (*Робочий простір*). Віртуальне середовище, де команда працює над завданнями, обмінюється файлами та спілкується.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. A Guide to the Project Management of the Knowledge (PMBOK® Guide). Seventh Edition, 2021, PMI, 370 p.
2. Agile Practice Guide: Paperback. USA, Project Management Institute, 2017, 210 p.
3. Al-Nimer M., Anwar M., Hani I.B. et al. Project management team and project success: an overview. *Manag Rev Q*, 2024. <https://doi.org/10.1007/s11301-024-00484-3>
4. Bočková K., Sláviková G., Gabrhel J. Game Theory as a Tool of Project Management», *Procedia - Soc. Behav. Sci.*, vol. 213, P. 709–715, <https://doi.org/10.1016/j.sbspro.2015.11.491>.
5. Bredikhin D., Sopov S., Ziuziun V. Competences of technical and digital higher education in project management. *Information Technology and Implementation (Satellite): Conference Proceedings*, Kyiv, Ukraine, 2024, P. 161-162.
6. Bushuyev S., Bushuyeva N., Ivko A., Bushuiev D., Bushuieva V. and Iazykov D. Principles of Circular Economy as a Driver of Development Projects. *IEEE International Conference on Smart Information Systems and Technologies (SIST)*, Astana, Kazakhstan, 2023, P. 37-41, doi: 10.1109/SIST58284.2023.10223513.
7. Bushuyev S., Bushuyeva N., Puziichuk A. Modelling of value chain creative level for managing innovation projects in BANI environment. *ScienceRise*, 2024, (2), P. 132-142. <https://doi.org/10.21303/2313-8416.2024.003622>
8. Bushuyev S., Tanaka H., Chetin E., Babayev I. Inspirational Intuition and Innovation in IT Project Management. *Scientific Journal of Astana IT University*, 2020, 10 (10). <https://doi.org/10.37943/IXYM7063>
9. Ciancarini P., Giancarlo R., Grimaudo G. Scrum@PA: Tailoring an Agile Methodology to the Digital Transformation in the Public Sector. *Information* 2024, 15, 110. <https://doi.org/10.3390/info15020110>
10. Dotsenko N., Chumachenko I., Bondarenko A., Chumachenko D. Methodological support for agile resource reallocation in a multi-project healthcare

environment. *Advanced Information Systems*, 2023, 7(4), 86–93.
<https://doi.org/10.20998/2522-9052.2023.4.12>

11. Gupta N., Sharma H., Kumar S., Kumar A., Kumar R., A Comparative Study of Implementing Agile Methodology and Scrum Framework for Software Development, 11th International Conference on System Modeling & Advancement in Research Trends (SMART), Moradabad, India, 2022, pp. 1088-1092, <https://doi.org/10.1109/SMART55829.2022.10047477>.

12. Huynh T.Q., Trinh N.B., Nguyen T.X. Nash equilibrium model for conflicts in project management, <https://doi.org/10.15625/1813-9663/34/3/13095>.

13. IPMA International Project Management Association. URL: <https://www.ipma.world/>

14. Kerzner Harold. Project Management: A Systems Approach to Planning, Scheduling, and Controlling 13th Edition. *John Wiley & Sons Inc*, 2022, 880 p.

15. Kolomiets A., Miroshnychenko I., Ziuziun V., Datsenko N., Kmytiuk T. Development of Project Management Models for Information Systems to Improve Website SEO Metrics. XI International Scientific Conference "Information Technology and Implementation" (IT&I 2024). CEUR Workshop Proceedings, 2024, Vol-3909, P. 334-345. https://ceur-ws.org/Vol-3909/Paper_27.pdf

16. Kononenko I., Sushko H. Mathematical model of software development project team composition optimization with fuzzy initial data. *Radioelectronic and computer systems*, 2021, no. 3, pp. 149-159. <https://doi.org/10.32620/reks.2021.3.12>.

17. Korytnyi O., Ziuziun V. Digital Ways of Controlling Risk Management and Assessment in Projects. 2st international scientific and practical conference «Information Systems and Technology: Results and Prospects» (IST 2025), Kyiv, Ukraine, 2025, pp. 178-181.

18. Korytnyi O., Ziuziun V. Justification of the need to create an information system for risk management of IT projects. *Information Technology and Implementation (Satellite): Conference Proceedings*, Kyiv, Ukraine, 2024, P. 236-237.

19. Kubiavka L., Khlevna Iu., Timinskyi O., Kolesnikova K., Raichuk I. Development of a model of personal data protection in the context of digitalization of

the educational sphere using information technology tools. The 14th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN 2023) November 1-3, 2023, Almaty, Kazakhstan. <https://doi.org/10.1016/j.procs.2023.12.215>

20. Kubiavka L., Kubiavka M., Lienkov S., Pyrogov K., Bernaz A.. Reflex Systems of Natural Language Processing in Educational Management Information Systems COLINS-2023. 7th International Conference on Computational Linguistics and Intelligent Systems, 2023, Kharkiv, Ukraine.

21. Kubiavka L., Zaremba V., Ziuziun V. Application of Game Theory Methods to Optimize the Stakeholder Management Process, IEEE 4th International Conference on Smart Information Systems and Technologies (SIST) 15-17 May, 2024, Astana, Kazakhstan. <https://doi.org/10.1109/SIST61555.2024.10629255>

22. Leong J., et al. Hybrid Project Management between Traditional Software Development Lifecycle and Agile Based Product Development for Future Sustainability. Sustainability (Basel), 2023, 15(2), 2. <https://doi.org/10.3390/su15021121>

23. Mihailides G., Young W. Is Agile Scrum Improving IT Project Outcomes?. International Journal of Management Science and Business Administration, 2024, P. 41-55.

24. Patrick Li. Jira 8 Essentials: Effective project tracking and issue management with enhanced Jira 8.21 and Data Center features, 6th Edition : Packt Publishing – ebooks Account, 2019. 412 p.

25. PMI Ukraine Chapter – Інститут проєктного менеджменту в Україні. Режим доступу: <https://pmiukraine.org/>

26. Project management methodologies: 12 popular frameworks, 2021. URL: <https://asana.com/resources/project-management-methodologies>

27. RACI Matrix. ProjectSmart. URL:

<https://www.projectsmart.co.uk/tools/raci-matrix.php>

28. Salikhov D., Succi G., Tormasov A. An empirical analysis of success factors in the adaption of the scaled agile framework – First outcomes from an empirical study (arXiv:2012.11144), 2020. <https://doi.org/arXiv.2012.1114410.48550>

29. Tam C., Moura E. J. da C., Oliveira T., Varajão J. The factors influencing the success of on-going agile software development projects. *International Journal of Project Management*, 2020, 38 (3), 165–176. <https://doi.org/10.1016/j.ijproman.2020.02.001>
30. Trihardianingsih L., Istighosah M., Alin A. Y., Asgar M. R. G. Systematic Literature Review of Trend and Characteristic Agile Model. *Jurnal teknik informatika*, 2023, 16 (1), 1. <https://doi.org/10.15408/jti.v16i1.28995>
31. Waterfall vs. Agile vs. Kanban vs. Scrum: What's the difference? 2022. URL: <https://asana.com/ru/resources/waterfall-agile-kanban>
32. Yücenur G. N. MCDM approach to investigate the effectiveness of SCRUM events in minimizing risk factors in project management. *Journal of Project Management*, 2023, 8(4), 227–238. <https://doi.org/10.5267/j.jpm.2023.7.001>
33. Ziuziun V. Analysis of possible risks in human resources management in IT companies. 5th International Scientific and Practical Internet Conference «Integration of Education, Science and Business in Modern Environment: Summer Debates», 2023. P.79-81.
34. Ziuziun V. Analysis of the impact of information technologies for making management decisions, including project ones. 2nd International Scientific and Practical Internet Conference, «Recent Trends in Science», 2023. P. 53-55.
35. Ziuziun V., Bredikhin D. Theoretical Justification for the Development of an IoT System for Freight Transportation Management. *Scientific journal Transactions of Kremenchuk Mykhailo Ostrohradskyi National University*, 2025, 1(9).
36. Ziuziun V., Kolomiets A. Aspects of decision-making in the management of human resources in IT projects of organizations. Міжнародна науково-практична конференція «Інтелектуальні інформаційні системи в управлінні проектами та програмами», Харків: ХНУРЕ, 2023. P. 45-49. <https://doi.org/10.30837/IISRRM.2023.09>
37. Ziuziun V., Kubiavka L., Kolomiets A., Starodubets V. The use of CMS as a mechanism for developing internet resources in IT project management. XXI Міжнародна науково-практична конференція Управління проектами у розвитку суспільства, Київ, КНУБА, 2024, С. 23-27.

38. Ziuziun V., Kulkovets V., Parasiuk L. Development of a Decision Support Information System for Managing Large Agile Teams in IT Projects. Collection of Scientific Papers of Admiral Makarov National University of Shipbuilding, 2024, 4(497), P. 166-172. [https://doi.org/10.15589/znp2024.4\(497\).23](https://doi.org/10.15589/znp2024.4(497).23)

39. Ziuziun V., Osoka D. Mathematical Rationale for Creating an Application for Conducting Random Meetings «Coffee Break». Bulletin of National Technical University «KhPI». Series: System Analysis, Control and Information Technologies, 2024, (2) 12, P. 76-80. <https://doi.org/10.20998/2079-0023.2024.02.11>

40. Ziuziun V., Petrenko N., Baida D. Application of Artificial Intelligence for SCRUM Optimization: AI Assistant for Team Velocity Prediction, IEEE 19th International Conference on Computer Science and Information Technologies (CSIT), Lviv, Ukraine, 2024, P. 1-4.

41. Ziuziun V., Starodubets V. Application of set theory for the mathematical justification of developing an IoT system for automated soil moisture monitoring. Taurida Scientific Herald. Series: Technical Sciences, 2024, 6, P. 29-39. <https://doi.org/10.32782/tnv-tech.2024.6.4>

42. Блага Н. В. Управління проектами: навч. посібник. Львів : Львівський державний університет внутрішніх справ, 2021. 152 с.

43. Бушуєв С., Пілюгіна К. Ціннісно-орієнтований проактивний менеджмент у командах високотехнологічних проєктів. Управління розвитком складних систем, 2023, (53), С. 5–15. <https://doi.org/10.32347/2412-9933.2023.53.5-15>

44. Бушуєва Н.С., Черниш О.В. Менеджмент проєктів сталого розвитку неприбуткових організацій у ризиковому оточенні. Управління розвитком складних систем, 2023, (55), P. 12–17. <https://doi.org/10.32347/2412-9933.2023.55.12-17>

45. Довідник з IPMA ІСВ4 в світі Agile (версія 2.3). Режим доступу: <http://ipma.kiev.ua/category/books/>

46. Євтушенко Г.І. Формування команди проєкту та організація її ефективної роботи (теоретичний аспект). Східна Європа: економіка, бізнес та управління,

2019, вип. 21. Режим доступу: http://www.easterneurope-ebm.in.ua/journal/21_2019/14.pdf

47.Зачко О. Б., Івануса А.І., Кобилкін Д.С. Управління проєктами: теорія, практика, інформаційні технології. Львів: ЛДУ БЖД, 2019, 173 с.

48.Зюзюн В. І. Аналіз аспектів підвищення ефективності управління ІТ-проєктами. XX Міжнародна конференція Управління проєктами у розвитку суспільства, Київ, КНУБА, 2023, С. 106-110.

49. Інститут проєктного менеджменту (США). Режим доступу: <https://www.pmi.org/>

50.Ковшун Н.Е., Левун О.І. Аналіз та реалізація проєктів: навч. посіб. [Електронне видання]. Рівне : НУВГП, 2022, 350 с. Режим доступу: <https://ep3.nuwm.edu.ua/23305/1/Аналіз%20та%20реалізація%20проєктів.pdf>

51.Кон Майкл. Оцінювання і планування в Agile / пер. з англ. Г. Якубовська. Харків : Вид-во «Ранок» : Фабула, 2019, 336 с.

52.Кубявка Л.Б., Латишева Т.В. Концепція побудови і принципи управління проєктного та продуктового ІТ-менеджменту. Управління розвитком складних систем, 2024, (57), 45–50.

53.Кузьмініх В.О., Коваль О.В., Тараненко Р.А. Моделі та засоби управління ІТ-проєктами [Електронний ресурс]: навч. посіб. Для студ. спеціальності 121 «Інженерія програмного забезпечення». Київ: КПІ ім. Ігоря Сікорського, 2023, 222 с. Режим доступу: <https://ela.kpi.ua/server/api/core/bitstreams/057779d8-d88f-4cef-b2d5-67086a013516/content>

54.Кузьмініх В.О., Тараненко Р.А. Основи управління ІТ проєктами [Електронний ресурс]: навч. посіб. Для студ. спеціальності 122 «Комп'ютерні науки». Київ: КПІ ім. Ігоря Сікорського, 2019, 75 с. Режим доступу: <https://ela.kpi.ua/server/api/core/bitstreams/7c313e5c-5477-4be2-9806-d32e9eace0c3/content>

55.Морозов В.В., Хандрік О.В., Коломієць А.С. Управління проєктами розвитку ІТ організацій: Навчальний посібник. К.: ВПЦ «Київський університет», 2020, 339 с.

56. Настанова до зводу знань з управління проєктами. Режим доступу: https://pmiukraine.org/wp-content/uploads/2022/08/PMBOK7_Ukr_ForPersonalUseOnly.pdf

57.Приймак В.М. Управління проєктами. Збірник кейсів [Електронний ресурс] : навч. посіб. / В. М. Приймак. К.: Київський національний університет імені Тараса Шевченка, 2021. 268 с.

58.Радченко Г., Левковська Т., Соболева, А. Особливості методології Kanban та Scrum при реалізації принципів Agile-маркетингу. Економіка та суспільство, 2023, (50). <https://doi.org/10.32782/2524-0072/2023-50-24>

59. Світова сертифікація проєктних менеджерів у РМІ. Режим доступу: <https://www.pmi.org/certifications>

60. Стандарти з управління проєктами від IPMA. Режим доступу: <https://www.ipma.world/individuals/standard-2/>

61. Українська асоціація управління проєктами «УКРНЕТ». Режим доступу: <http://ipma.kiev.ua/category/news/>

62.Храпкін О., Кіндрат О., Чопей, Р. Управління проєктами в ІТ-галузі: методики, інструменти та керування ризиками. Економіка та суспільство, 2023, 55. Режим доступу: <https://doi.org/10.32782/2524-0072/2023-55-110>

63.Швабер К., Сазерленд Дж. Посібник зі Скраму (Повний навчальний посібник зі Скраму: правила гри) [Електронний ресурс]. Режим доступу: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Ukrainian.pdf>

ПРЕДМЕТНИЙ ПОКАЖЧИК

<i>Active sprints</i>	25	<i>Row report</i>	152, 158
<i>Agile</i>	11, 15, 16	<i>Scrum</i>	11, 15, 24, 62
<i>Assign issues</i>	42	<i>Scrum-проект</i>	23
<i>Backlog</i>	18, 20, 23, 27, 30	<i>Set a reminders</i>	156
<i>Battery widget</i>	107, 110	<i>Set availability</i>	158
<i>Board</i>	19, 20	<i>Sheet summary report</i>	158
<i>Board settings</i>	37	<i>Smartsheet</i>	133, 135, 138, 150, 161
<i>Burn down chart</i>	24, 62, 65	<i>Software development</i>	21, 26
<i>Calendar view</i>	156	<i>Sprint</i>	19, 20, 30, 68
<i>Chart widget</i>	107, 112	<i>Sprint burndown</i>	70
<i>Comment</i>	20	<i>Sprint report</i>	32, 65
<i>Complete sprint</i>	25	<i>Start date</i>	154, 170
<i>Control chart</i>	37, 39, 48	<i>Status</i>	20, 112, 117, 144
<i>Create issue</i>	18, 42	<i>Status meetings</i>	174
<i>Create project</i>	26	<i>Story points</i>	24
<i>Create sprint</i>	24	<i>Sub-task</i>	19
<i>Cumulative flow diagram</i>	37, 39, 48	<i>Task</i>	24
<i>Done</i>	19, 27, 31, 117	<i>Team performance report</i>	62
<i>Due date</i>	20	<i>Timeline</i>	154
<i>Edit issues</i>	42	<i>Timeline widget</i>	108, 109, 112
<i>End date</i>	154, 170	<i>To do</i>	19, 27, 31
<i>Epic</i>	19, 24	<i>User management</i>	161
<i>Gantt view</i>	159	<i>Uspacy</i>	80, 86, 92, 98
<i>Github</i>	16, 49, 50	<i>Velocity</i>	24
<i>In progress</i>	19, 31, 117	<i>Velocity chart</i>	65
<i>Issue statistics</i>	62	<i>Velocity report</i>	26, 32, 62
<i>Issues</i>	19	<i>Wip limit</i>	39
<i>Jira</i>	14, 16, 17, 33, 40, 44, 45, 47, 49, 61	<i>Work in progress</i>	37
<i>Kanban</i>	11, 15, 28, 38, 56	<i>Workflows</i>	24, 34, 35, 37, 44
<i>Kanban-проект</i>	37	<i>Автоматизація</i>	128
<i>Log work</i>	20	<i>Акаунт</i>	50
<i>Monday.com</i>	103, 109, 126	<i>Алгоритм</i>	51, 53, 66
<i>Move to in progress</i>	46	<i>Аналіз</i>	9, 68, 75
<i>Notifications</i>	151	<i>Виконання</i>	24, 40, 57, 68, 91, 93, 112, 202
<i>Online Gantt</i>	168, 170, 173, 174	<i>Вимоги</i>	18, 141
<i>Permission scheme</i>	42	<i>Візуалізація</i>	70, 141, 159
<i>Priority</i>	19	<i>Дані</i>	55, 68
<i>Progress</i>	169	<i>Дашборд</i>	16, 63, 64, 69, 117, 121, 133
<i>Project</i>	21, 122, 125, 126	<i>Експорт звітів</i>	175
<i>Project management</i>	114	<i>Завдання</i>	18, 42, 93, 94, 98, 104, 196, 201
<i>Project settings</i>	23, 24, 27, 28, 41	<i>Задача</i>	52, 57, 59
<i>Project view</i>	168	<i>Звіт</i>	68, 94
<i>Report</i>	156, 160	<i>Звіт про ефективність команди</i>	62
<i>Reports</i>	67	<i>Звіт про залишковий час</i>	62
<i>Resolution</i>	20	<i>Звіт про проблеми</i>	62
<i>Resolve issues</i>	42	<i>Звіт про швидкість</i>	24, 26, 62
<i>Resource allocation</i>	149, 154	<i>Інструменти</i>	16, 110, 169
<i>Resource management</i>	149, 156	<i>Інформаційна система</i>	80
<i>Resources</i>	171	<i>Інформаційні технології</i>	8, 9, 11

<i>Інформація</i>	135, 140	<i>Спринт</i>	33, 67, 68
<i>Команда</i> ...	21, 23, 33, 54, 59, 62, 85, 89, 109, 123, 158	<i>Статус</i>	14, 20, 21, 32, 56, 93, 95
<i>Коментар</i>	20	<i>Теги</i>	20, 122
<i>Комунікація</i>	95, 104, 120	<i>Терміни</i>	14, 93
<i>Користувач</i>	52, 55, 135, 139, 154, 169	<i>Тестування</i>	128
<i>Менеджер</i>	63, 135, 144, 175	<i>Управління</i>	9, 114, 146
<i>Моніторинг</i>	117	<i>Управління WIP-лімітами</i>	39
<i>Планування</i>	8, 15, 32, 99, 103	<i>Управління завданнями</i>	90, 97, 133, 146
<i>Платформа</i>	105, 122	<i>Управління командою</i>	90
<i>Проект</i> ..	30, 41, 44, 49, 50, 51, 55, 64, 67, 83, 98, 111, 134, 135, 152, 203	<i>Управління командою розробників</i>	114
<i>Процеси</i>	9, 17, 59, 67, 69	<i>Управління проектами</i> ..	8, 10, 11, 15, 18, 54, 59, 64, 66, 70, 80, 82, 86, 88, 103, 105, 113, 118, 120, 122, 133, 134, 135, 138, 141, 149, 157, 169
<i>Процеси управління</i>	9	<i>Управління ресурсами</i>	154, 155
<i>Ризики</i>	9, 143, 174	<i>Управління ризиками</i>	82
<i>Рішення</i>	94	<i>Шаблон</i>	114, 137, 142
<i>Робочий процес</i>	35, 47, 94, 156		
<i>Система</i>	40, 172		