

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

**Факультет інформаційних технологій**

Кафедра технологій управління

Спеціальність 122 «Комп'ютерні науки»  
Освітньо-наукова програма «Управління проектами»

**КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА**

на тему

**“Дослідження процесів управління проектом створення програми  
автоматичного створення та розрахунку схем вишивання бісером”**

**Студента 2-го курсу групи УП-21**

Сопова Сергія Сергійовича

**Науковий керівник:**

кандидат техн. наук, асистент  
Хлевний Андрій  
Олександрович

---

*(підпис студента)*

---

*(дата)*

---

*(підпис)*

**Попередній захист:**

---

*(Висновок: “До захисту в Екзаменаційній комісії”)*

Завідувач кафедри  
технологій управління

---

*(підпис)*

Морозов В. В.

---

*(прізвище, ініціали)*

---

*(дата)*

**ЗАТВЕРДЖУЮ**

Завідувач кафедри  
професор Морозов В. В.

“\_\_\_” \_\_\_\_\_ 2025 року

## **ЗАВДАННЯ**

### **НА ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ**

Студент: Сопов Сергій Сергійович

Група: УП-21

1. Тема кваліфікаційної роботи: «Дослідження процесів управління проектом створення програми автоматичного створення та розрахунку схем вишивання бісером». Затверджена на засіданні кафедри технологій управління протокол №4 від 18.12.2024 р.

2. Строк подання студентом роботи – «10» травня 2025 р.

3. Цільова установка та вихідні дані до роботи: дослідження різних методів та інструментів для управління проектом, їх використання у плануванні проекту, управління ризиками та управління якістю; вихідними даними є складений план проекту.

4. Зміст роботи: розробка концепції проекту та аналіз ринку, аналіз предметної галузі, мета, цілі та продукт проекту, SWOT-аналіз, розробка концепції проекту, розробка програмного забезпечення проекту, концептуальна та математична модель проекту, організаційна структура команди проекту, опис модулів програмного забезпечення, розроблення концептуальної та фізичної моделей бази даних проекту, гнучкі технології створення продукту та планування ІТ-проекту, визначення вимог до проекту, календарне планування та віхи проекту, використання методології Scrum для управління проектом, життєвий цикл проекту та його етапи, технології управління ІТ-проектом, управління ризиками проекту, запобігання

негативним наслідкам та контроль за прогресом, управління якістю проєкту та забезпечення високої якості продукту, аналіз розробленого продукту.

5. Перелік графічного матеріалу: Загальна діаграма кошторисів проєкту, концептуальна модель бази даних, логічна модель бази даних, схема ієрархії команди проєкту, дошки задач учасників команди, приклад дошки задач, час потрібний для створення семи, кількість помилок у роботі.

6. Календарний план виконання роботи:

№ з/п	Назва частин роботи	План виконання роботи
1	Вивчення інформації з предмету дослідження.	20.12.2024 – 08.01.2025
2	Збір і вивчення матеріалів досліджуваної теми.	09.01.2025. – 10.03.2025
3	Складання плану кваліфікаційної роботи магістра.	10.03.2025 – 17.03.2025
4	Ознайомлення наукового керівника з планом кваліфікаційної роботи магістра. Внесення змін.	17.03.2025 – 20.03.2025
5	Підготовка розділу 1	20.03.2025 – 03.04.2025
6	Підготовка розділу 2	04.04.2025 – 18.04.2025
7	Підготовка розділу 3	19.04.2025 – 03.05.2025
8	Підготовка розділу 4	04.05.2025 – 10.05.2025
9	Оформлення кваліфікаційної роботи.	16.05.2025
10	Передача роботи на рецензування.	16.05.2025
11	Попередній захист кваліфікаційної роботи.	12.05.2025
12	Захист роботи.	26.05.2025 - 28.05.2025

Дата видачі завдання: «19» грудня 2025 р.

Керівник роботи кандидат техн. наук, асистент Хлевний А. О.

\_\_\_\_\_  
(підпис)

Завдання прийняв до виконання:

студент групи УП-21 Сопов С. С

\_\_\_\_\_  
(підпис)

## Зміст

Зміст.....	4
Анотація.....	7
Вступ.....	9
<b>РОЗДІЛ 1. АНАЛІЗ РИНКУ ТА ФОРМУВАННЯ ОСНОВ РОЗРОБКИ ПРОЄКТУ.....</b>	<b>13</b>
1.1 Розгляд тенденцій рішень по розробці програм у сегменті.....	13
1.2 Дослідження впливу технології на ринок виробництва .....	24
1.2.1 Автоматизація як ключ до масштабування .....	24
1.2.2 Зниження витрат і контроль за собівартістю .....	25
1.2.3 Покращення гнучкості й швидкості реагування на ринок.....	26
1.2.4 Стандартизація якості та мінімізація браку.....	26
1.2.5 Нові можливості для аналітики та управління .....	26
1.2.6 Зміна ролі персоналу та скорочення часу навчання .....	27
1.3 Оцінка економічної доцільності впровадження програми .....	27
1.3.1 Скорочення витрат, підвищення продуктивності та якісна оптимізація ресурсів.....	30
1.3.2 Підвищення гнучкості виробництва, адаптація до ринку та індивідуалізація продукту .....	31
1.3.3 Оптимізація фінансового контролю та підвищення рентабельності .....	31
1.3.4 Термін окупності, економічні ризики та потенціал масштабування .....	32
1.4 Формулювання цілей та формулювання технічного завдання на розробку .....	33
1.4.1 Цілі розробки програмного забезпечення .....	33
1.4.2 Технічне завдання на розробку.....	34
<b>РОЗДІЛ 2. РОЗРОБКА ТА ПЛАНУВАННЯ ПРОЄКТУ .....</b>	<b>36</b>
2.1 Обґрунтування вибору методології управління проєктом.....	36
2.2 Планування проєкту та кошторисні розрахунки .....	43

2.3 Команда проєкту.....	44
2.4 Розрахунко витрат проєкту.....	48
2.5 SCRUM аналіз.....	50
2.6 Концептуальна модель програми .....	52
3.1 Функціональні та нефункціональні вимоги до програмного забезпечення для створення схем вишивання бісером .....	63
3.2 Ієрархія учасників команди проєкту.....	66
3.3 Ризики проєкту .....	68
3.4 Зацікавлені сторони проєкту та управління ними.....	76
3.5 Використання ПЗ у проєкті для керування процесами .....	78
<b>РОЗДІЛ 4. ОЦІНКА ЕФЕКТИВНОСТІ ВПРОВАДЖЕННЯ .....</b>	<b>84</b>
4.1 Характеристика середовища впровадження.....	84
4.1.1 Опис підприємства, в якому впроваджено систему .....	84
4.1.2 Основні бізнес-процеси, пов'язані з проєктом .....	85
4.2 Аналіз використаних моделей, методів і засобів управління проєктом .....	86
4.2.1 Використані моделі та засоби .....	86
4.2.2 Обґрунтування вибору інструментів управління.....	87
4.3 Оцінка ефективності впровадження інформаційної системи.....	88
4.4 Статистичні результати використання програмного забезпечення .....	89
4.5 Моніторинг результатів реалізації проєкту .....	91
4.6 Рекомендації щодо подальшого вдосконалення системи .....	93
4.6.1 Масштабування системи .....	94
4.6.2 Можливість комерціалізації.....	94
4.6.3 Розширення функціональності системи.....	94
4.6.4 Оптимізація внутрішніх процесів .....	95
<b>Висновки.....</b>	<b>96</b>
<b>Список використаних джерел .....</b>	<b>99</b>

<b>Додаток А.....</b>	<b>101</b>
<b>Додаток Б.....</b>	<b>102</b>

## Анотація

Метою даної кваліфікаційної роботи є створення програмного забезпечення для автоматизованого формування схем вишивання бісером із розрахунком матеріалів та інтеграцією з обліковою системою 1С. Проект спрямований на оптимізацію ключових виробничих процесів підприємства, що займається виготовленням наборів для творчості, зокрема картин для вишивання бісером.

Ціль проєкту полягає у впровадженні інструменту, який дозволяє значно скоротити витрати часу на підготовку схем, автоматизувати підбір і підрахунок матеріалів відповідно до складу, розрахувати собівартість виробу та інтегрувати ці дані у внутрішні бізнес-процеси підприємства.

Наукова новизна роботи полягає у створенні комплексного програмного рішення, орієнтованого виключно на вишивку бісером, яке об'єднує графічний аналіз зображень, логіку облікової системи та функції калькулятора собівартості в єдину систему. Такий підхід дозволяє підприємству досягти наскрізної автоматизації виробничого ланцюга — від прийому зображення до фінального пакування продукту.

Кваліфікаційна робота складається з вступу, чотирьох розділів основної частини, висновків, переліку використаних джерел та додатків. У першому розділі проведено аналіз ринку та оцінено сучасні рішення у сфері автоматизації створення схем для вишивки бісером, визначено економічну доцільність розробки, а також сформульовано технічне завдання. У другому розділі детально описано управлінсько-методологічний підхід до реалізації проєкту, розроблено архітектуру програмного продукту, план дій, команду та фінансову модель реалізації. Третій розділ присвячено управлінню ризиками, організації взаємодії з зацікавленими сторонами, моніторингу й управлінню проєктом за допомогою Trello. У четвертому розділі проведено оцінку ефективності впровадження, описано статистичні зміни в роботі підприємства після впровадження ПЗ, розглянуто моделі управління, результати моніторингу та рекомендації щодо вдосконалення.

Результати проєкту впроваджено у виробничу діяльність підприємства «АртБісер», що спеціалізується на виготовленні наборів для вишивки. Після впровадження програми було зафіксовано скорочення часу на створення схем на 65%, зменшення помилок у підборі матеріалів, а також підвищення точності обрахунку собівартості готового виробу. Отримані результати підтверджують як практичну ефективність розробки, так і доцільність її масштабування.

Обсяг роботи становить 92 сторінки, вона містить 9 рисунків, 7 таблиць та 2 додатки. Основні результати презентовано на студентській науково-практичній конференції з управління інноваційними проєктами.

Ключові слова: автоматизація, вишивка бісером, створення схем, розрахунок матеріалів, управління проєктами, програмне забезпечення, інтеграція з ІС.

*Ключові слова: автоматизація, вишивка бісером, створення схем, розрахунок матеріалів, управління проєктами, програмне забезпечення, інтеграція з ІС.*

## Вступ

У сучасних умовах цифрової трансформації, коли технології проникають у всі сфери економіки, особливого значення набуває автоматизація навіть у тих галузях, які тривалий час залишалися здебільшого ручними та ремісничими. Однією з таких є виробництво товарів для творчості, зокрема вишивання бісером, що активно розвивається в Україні та за її межами. Цей вид декоративно-прикладного мистецтва поєднує естетичну складову з високою трудомісткістю процесу виготовлення. Підприємства, що спеціалізуються на виробництві наборів для вишивання, зіштовхуються з низкою викликів: потребою в точності, стабільності якості, високими витратами часу на розробку схем, ручним підбором матеріалів і складністю розрахунків. Усе це значно знижує ефективність бізнесу, ускладнює масштабування виробництва та підвищує собівартість продукції.

Інформаційні технології сьогодні відкривають широкі можливості для автоматизації рутинних творчих і виробничих процесів, навіть у таких традиційно "ручних" галузях. Зокрема, на стику графічних технологій, систем управління обліком (ERP) та математичних алгоритмів з'являються умови для створення програмного забезпечення, що не лише полегшує роботу дизайнера, а й інтегрується в управлінську систему підприємства. Проте огляд сучасного ринку програм показує, що більшість існуючих рішень або не адаптовані під вишивку саме бісером, або не мають можливості точного розрахунку матеріалів відповідно до залишків на складі та фінансової моделі підприємства. Відсутність галузевої спеціалізації, недосконалість інтерфейсів, відсутність модулів інтеграції з системами обліку — усе це створює потребу у спеціалізованому, адаптованому до реальних умов виробництва програмному продукті.

У цьому контексті, дана робота присвячена розробці програмного забезпечення, що вирішує одразу кілька ключових задач: автоматично

перетворює зображення на схему для вишивання бісером, дозволяє швидко і точно розрахувати необхідну кількість матеріалів, перевірити їхню наявність на складі через зв'язок з обліковою системою 1С та визначити загальну собівартість готового набору для вишивки. Розробка такої системи є не лише технічним викликом, а й управлінським завданням — побудувати ефективний, контрольований проєкт розробки, адаптований до обмежених ресурсів малого підприємства.

Реалізація цього проєкту дозволяє перейти від фрагментарних ручних дій до єдиної наскрізної цифрової логіки виробництва: від обробки вхідного зображення — до готової калькуляції, сформованої упаковки і передачі в логістику. Програма повинна не лише відповідати технічному завданню, а й враховувати специфіку творчої діяльності, вимоги зручності, адаптивності інтерфейсу, підтримки кольорових палітр, обліку складських залишків і можливості персоналізації замовлень. Це робить задачу надзвичайно складною і водночас перспективною з точки зору впливу на організаційні процеси підприємства.

Об'єктом дослідження в даній роботі є бізнес-процеси підприємства, що спеціалізується на виробництві наборів для вишивання бісером — від етапу прийому зображення до розрахунку матеріалів, складання пакування і формування замовлення. Предметом дослідження є процес управління проєктом розробки інформаційної системи, орієнтованої на автоматизацію створення схем вишивки та інтеграцію з інфраструктурою підприємства, зокрема з 1С. Робота орієнтована не лише на технічну реалізацію алгоритму генерації схеми, а й на управлінське обґрунтування вибору моделей, методів, командної структури, способів організації проєктного процесу.

У межах проєкту було сформульовано мету: створення прикладного програмного забезпечення для автоматизованої генерації схем для вишивки бісером, з функціями розрахунку матеріалів і повною інтеграцією з внутрішньою обліковою системою підприємства. Для досягнення цієї мети

були поставлені такі завдання: проаналізувати ринок і технології, розробити технічне завдання, визначити архітектуру рішення, побудувати команду, реалізувати і протестувати ключові модулі системи, оцінити економічний ефект впровадження, розробити рекомендації з масштабування.

Для дослідження були використані методи системного аналізу для виявлення проблем у бізнес-процесах підприємства, порівняльний аналіз існуючих програм, моделювання архітектури системи, анкетування працівників, методи управління проектами (Agile, Scrum), вартісне моделювання, методи тестування й аналізу якості, інструменти для організації командної роботи (Trello, Excel).

Новизна роботи полягає у побудові програмної системи, яка вперше об'єднує генерацію схем вишивки бісером, автоматичну калькуляцію матеріалів, інтеграцію з 1С та організацію цифрового контролю за собівартістю. Такий комплекс не представлений на ринку у вигляді готового рішення. Розроблене ПЗ може бути адаптоване як для внутрішнього використання, так і для комерційного поширення серед аналогічних виробників.

Практична цінність результатів проявляється у доведеній економічній доцільності впровадження: за результатами тестування на підприємстві було досягнуто скорочення часу створення однієї схеми в середньому з 3 годин до 45 хвилин, зниження кількості помилок у підборі матеріалів на 85%, підвищення точності калькуляції витрат, зменшення відходів і перевитрат. Роботу апробовано в рамках пілотного впровадження на базі конкретного підприємства, а її результати були представлені на студентській науково-практичній конференції, що проходила в межах кафедрального семінару з управління інноваційними проектами.

Таким чином, виконана магістерська робота демонструє практичне втілення підходів до цифрової трансформації малого виробництва, об'єднує технічне, управлінське та економічне обґрунтування створення

інформаційної системи і може стати основою для розвитку галузевих цифрових рішень у сфері рукодільної промисловості.

Зважаючи на високий потенціал розробленого рішення, доцільним є подальше розширення сфери його застосування як всередині підприємства, так і за його межами. Насамперед, програма може бути масштабована на інші виробничі напрямки в межах підприємства, які пов'язані з індивідуальним виготовленням творчих товарів, наприклад, створенням схем для вишивання нитками, ткацтвом або аплікацією. Важливо підкреслити, що гнучка архітектура системи та модульний підхід дозволяють адаптувати програму до специфіки інших технік без необхідності повної переробки ядра. Це відкриває перспективу для підприємства сформувати багатofункціональну цифрову платформу для всіх напрямів творчого виробництва.

Крім того, розроблений програмний продукт має комерційний потенціал як окреме програмне рішення для зовнішнього ринку. Ураховуючи значну кількість малих студій, приватних майстрів та онлайн-магазинів, які працюють у сфері вишивки бісером, існує попит на програму, яка дозволяє швидко і точно формувати схеми, автоматично розраховувати матеріали, узгоджуючи їх з наявними на складі ресурсами. Таким чином, програму можна адаптувати до SaaS-моделі або створити версію для розповсюдження з ліцензійною підпискою, що забезпечить додаткове джерело прибутку підприємству-розробнику або стане основою для запуску нового бізнес-напряму.

Особливої уваги заслуговує подальший розвиток інтерфейсних рішень і сервісних функцій для кінцевих користувачів. Доцільно впровадити мобільний додаток або веб-інтерфейс, що дозволить розширити коло користувачів, спростити доступ до сервісу та підвищити рівень взаємодії з клієнтом.

## **РОЗДІЛ 1. АНАЛІЗ РИНКУ ТА ФОРМУВАННЯ ОСНОВ РОЗРОБКИ ПРОЄКТУ**

### **1.1 Розгляд тенденцій рішень по розробці програм у сегменті**

Останні тенденції у створенні програм для розбиття зображень на частинки з метою створення схем для вишивання або вишивання бісером включають кілька ключових напрямів розвитку, що демонструють еволюцію програмного забезпечення від простих графічних редакторів до складних систем автоматизованого проєктування з елементами штучного інтелекту та хмарних обчислень.

#### **Інтеграція штучного інтелекту та машинного навчання:**

Однією з найбільш помітних і технологічно значущих тенденцій у сфері створення програм для вишивання є впровадження алгоритмів штучного інтелекту (AI) та машинного навчання (ML). Такі алгоритми здатні здійснювати автоматизовану обробку вхідного зображення на глибокому рівні, включаючи розпізнавання контурів об'єктів, виділення кольорових зон, виявлення світлотіньових переходів і навіть ідентифікацію повторюваних візерунків чи орнаментів.

Глибокі нейронні мережі, натреновані на великих наборах зображень готових схем вишивки, картин та фото вишитих робіт, дозволяють програмі не лише ділити зображення на кольорові області, але й визначати, які саме візуальні елементи є ключовими для збереження художньої цінності зображення. На цьому етапі система аналізує колірну гармонію, композиційну структуру, домінуючі контури, і пропонує найбільш доцільне їх зображення у вигляді символічної схеми з прив'язкою до конкретних кольорів бісеру або ниток.

Алгоритми ML також можуть вивчати попередні проєкти підприємства або вподобання клієнтів (якщо вони фіксуються в системі), щоб формувати індивідуальні рекомендації при повторних замовленнях. Наприклад, система може запам'ятовувати, які типи кольорових рішень частіше вибирають для пейзажів чи ікон, і в подальшому враховувати ці переваги автоматично.

Ще одним важливим аспектом є здатність штучного інтелекту адаптувати схему відповідно до заданих обмежень — наприклад, якщо користувач або система управління запасами ІС визначає, що деякі кольори або номери бісеру тимчасово недоступні, АІ може підібрати найближчі альтернативи, зберігаючи цілісність художнього вигляду схеми.

Додатково АІ-модулі можуть генерувати попередній перегляд готової картини у вишитому вигляді на основі обробленої схеми, що суттєво полегшує оцінку результату ще до початку виготовлення набору або передачі схеми майстрові. Таке візуальне моделювання базується на технологіях згорткових нейронних мереж (CNN), що забезпечують реалістичну текстуру і відображення структури шва чи плетіння.

Впровадження АІ і ML у програмне забезпечення не лише автоматизує рутинні процеси, але й підвищує якість готового продукту, забезпечує гнучкість виробництва, зменшує людський фактор при підборі матеріалів і дозволяє реалізовувати унікальні дизайнерські рішення на основі аналізу великих обсягів візуальних і комерційних даних.

#### **Адаптивна кольорокорекція та точне зіставлення палітр:**

У сучасному програмному забезпеченні для створення схем вишивання адаптивна кольорокорекція відіграє ключову роль у процесі перетворення зображення на точну і реалістичну схему для вишивання бісером чи нитками. Такі системи дозволяють автоматично або напівавтоматично зіставити кольори вхідного зображення з реальними відтінками наявних матеріалів — бісеру, муліне, акрилових ниток, канви тощо.

Програмні модулі кольорокорекції не обмежуються простим порівнянням пікселів. Вони враховують такі фактори, як освітлення джерельного зображення, колірний баланс, тип візуалізації (фото, скан, цифрова ілюстрація), а також фізичні властивості матеріалів (блиск, прозорість, фактура поверхні бісеру). Це дозволяє уникнути ситуацій, коли вибраний візуально "близький" колір на екрані в реальності має зовсім інше враження — надто яскравий, темний або холодний.

Інтелектуальні алгоритми зіставлення кольорів використовують вбудовані або підключені бібліотеки палітр, які базуються на офіційних каталогах провідних виробників — Preciosa, ТОНО, Miyuki, DMC, Anchor та інших. Дані з палітр включають не лише RGB-коди, а й фізичні характеристики матеріалів, коди постачання та інформацію про наявність в регіональному складі (через інтеграцію з ERP або ІС-системами). Таким чином, система одразу враховує не лише ідеальний колір, а й реальну доступність компонентів у виробництві.

Окремі модулі програм дозволяють користувачеві встановлювати критерії для оптимізації підбору — наприклад, обирати між максимально точним візуальним відтворенням або мінімізацією кількості різних кольорів для здешевлення набору. Деякі з них навіть можуть проводити порівняльний аналіз кількох варіантів підбору: точний (реалістичний), спрощений (з меншою кількістю кольорів), економічний (на основі залишків на складі).

Ще одним важливим напрямом є підтримка користувацьких палітр — підприємства можуть створювати власні бібліотеки кольорів із закріпленими кодами внутрішнього обліку, до яких алгоритм також зіставляє зображення. Це особливо корисно для оптимізації логістики, зменшення залишків матеріалів та формування стандартних виробничих наборів.

Кольорокорекційні системи також можуть працювати в парі з графічними модулями попереднього перегляду, які дозволяють візуалізувати, як виглядатиме картина після вишивки певними кольорами та матеріалами. У поєднанні з віртуальним 3D-прев'ю це створює майже повну імітацію кінцевого результату, що особливо корисно при прийнятті рішень про друк схем, закупівлю матеріалів чи продаж готових наборів.

Таким чином, сучасна адаптивна кольорокорекція — це не лише візуальна функція, а потужний інструмент економічного, технологічного та художнього управління процесом виготовлення схем вишивки, що підвищує точність, знижує витрати та дозволяє забезпечити високу якість кінцевого продукту.

## **Модульна структура з широкою кастомізацією:**

Сучасні програмні рішення для створення схем вишивання базуються на модульній архітектурі, що дозволяє гнучко адаптувати систему до конкретних потреб користувача. Ключовою перевагою такого підходу є можливість формувати індивідуальне програмне середовище — від мінімального функціоналу для новачків до повноцінного комплексу інструментів для промислового використання.

Базовий модуль зазвичай включає генерацію схеми вишивки з растрового або векторного зображення: система автоматично розбиває зображення на сітку, зіставляє пікселі з кольорами доступної палітри, формує умовні позначення та попередній вигляд майбутньої вишивки. Цього достатньо для швидкого створення простої схеми, наприклад, для приватного використання чи подарунка.

Додаткові модулі розширюють функціональність системи у різних напрямках:

- **Редагування палітри вручну або напівавтоматично**, з можливістю виключення певних кольорів (недоступних або надто дорогих), заміни близьких відтінків або зменшення загальної кількості кольорів з урахуванням допустимої втрати точності;
- **Налаштування алгоритмів стикування бісеру чи ниток**, де враховуються форма, розмір і спосіб укладки — наприклад, імітація круглої або циліндричної форми бісеру, вибір техніки "мозаїка", "плетіння" або "вишивка хрестиком";
- **Симуляція готової роботи** в реалістичному візуальному відображенні, з тінями, освітленням, блиском матеріалів — це дає змогу побачити, як виглядатиме готова картина не тільки на схемі, але й у матеріальному виконанні;
- **Автоматизований розрахунок матеріалів**, що інтегрується з внутрішніми або зовнішніми базами постачальників (наприклад, 1С), і

дозволяє одразу отримати список потрібних позицій, їхню кількість, вартість та наявність на складі або у постачальника;

- **Генерація комерційного набору**, включно з формуванням друкованих схем, інструкцій, переліку матеріалів, QR-кодів для онлайн-інструкцій і навіть макетів для пакування.

Окремі системи дозволяють розгортати кастомізовані середовища під окремих користувачів або організації — наприклад, великі виробники можуть мати власні модулі контролю якості, інтеграції з виробничими процесами, обліку залишків або CRM. Для користувачів з художньою освітою чи дизайнерами передбачені модулі з ручним керуванням усіма етапами трансформації зображення, включно з фільтрами, ручним редагуванням кольорових зон, векторизацією елементів та накладанням шарів.

Гнучка модульна структура дає змогу не тільки масштабувати систему в залежності від зростання потреб підприємства чи кінцевого користувача, але й знижує вартість впровадження. Користувач платить лише за ті компоненти, які йому дійсно потрібні, що особливо важливо для малого бізнесу або індивідуальних майстрів.

Завдяки відкритій архітектурі деякі програми також підтримують інтеграцію сторонніх плагінів або API, що дозволяє розробляти власні функції або доповнювати систему нестандартним функціоналом, наприклад, генерацією схем з фото в реальному часі, підтримкою мобільних застосунків, або зв'язком з інтернет-магазином.

Таким чином, модульний принцип не лише розширює функціональність програм, а й створює основу для побудови індивідуальних цифрових екосистем у сфері виробництва товарів для вишивання, підвищуючи ефективність, гнучкість і ринкову конкурентоспроможність компаній.

**Хмарні обчислення та збереження даних:**

Останні роки спостерігається активний перехід програмного

забезпечення для створення схем вишивки до хмарної інфраструктури. Цей перехід відкриває нові можливості як для кінцевих користувачів, так і для підприємств, зокрема у сфері масштабування, доступності, зберігання й обміну даними.

Передусім, хмарні обчислення дозволяють працювати з великими об'ємами графічної інформації без потреби у високопродуктивному локальному обладнанні. Завантаження зображень, їх обробка, генерація схем, адаптація під обрані матеріали – усі ці процеси можуть здійснюватися на віддалених серверах. Це особливо важливо для підприємств, де швидкість обробки замовлень напряму впливає на виробничу ефективність.

Хмарне середовище також забезпечує багаторівневе збереження даних. Зокрема, зберігаються:

- згенеровані схеми та супутня документація (наприклад, PDF-інструкції, візуалізації готових картин);
- архіви використаних матеріалів і кольорових палітр;
- історія змін і версій кожного замовлення;
- дані про клієнтів і повторні запити (для персоналізованого сервісу);
- аналітичні звіти про використання ресурсів, витрати та прибутки.

Доступ до цих даних можна отримати з **будь-якого пристрою** — комп'ютера, планшета, смартфона, що значно спрощує управління процесами для віддалених працівників, дизайнерів, операторів, менеджерів чи навіть клієнтів. Для внутрішньої комунікації хмарні платформи часто мають вбудовані функції спільної роботи — коментарі до схем, журнал змін, завантаження зображень для узгодження тощо.

Інтеграція з обліковими системами (зокрема 1С, CRM, ERP) у хмарному форматі дозволяє:

- автоматично оновлювати інформацію про залишки матеріалів, відображати доступні ресурси в режимі реального часу;
- формувати динамічні прайс-листи та розрахунки вартості комплектів залежно від поточних цін у базі постачальника;

- оптимізувати логістику, відстежуючи не лише внутрішні виробничі процеси, а й взаємодію з клієнтами й постачальниками.

Ще однією важливою перевагою є безпека та резервне копіювання. Провідні хмарні сервіси пропонують захист даних за допомогою шифрування, двофакторної аутентифікації та автоматичних бекапів, що гарантує стабільну роботу навіть у випадках збою обладнання або атак зловмисників.

Для компаній, що працюють з великою кількістю індивідуальних замовлень або продають набори для вишивання онлайн, хмарні технології створюють основу для масового, але персоналізованого виробництва. Кожне замовлення може зберігатися як окремий проєкт з історією, шаблонами, контактами клієнтів, побажаннями, повторними замовленнями — що спрощує повторне виготовлення, аналіз попиту та підвищення рівня обслуговування.

Зрештою, хмарна архітектура підтримує довготривалу стратегію цифрової трансформації підприємства: вона полегшує впровадження нових сервісів, масштабування інфраструктури без значних витрат, та сприяє впровадженню сучасних підходів до обробки даних, включно з аналітикою, прогнозуванням попиту та рекомендаційними системами.

#### **Автоматичний підбір матеріалів з інтеграцією в складські системи:**

Одна з ключових інновацій у сучасному програмному забезпеченні для створення схем вишивки — це глибока інтеграція з корпоративними обліковими та складськими системами, такими як 1С:Підприємство, SAP, Odoo, Microsoft Dynamics 365 та іншими. Така інтеграція здійснюється через API-інтерфейси, веб-сервіси або внутрішні модулі обміну даними, що забезпечують прямий зв'язок між дизайнерською частиною програми та реальними запасами матеріалів на підприємстві.

На практиці це означає, що програма автоматично:

- зіставляє кольори зображення зі специфічними артикулами бісеру, ниток чи тканин, доступними у складі;

- перевіряє їхню наявність у реальному часі;
- обирає найближчі доступні аналоги, якщо точний відтінок або розмір недоступний;
- формує автоматизовану специфікацію матеріалів із зазначенням кількості, постачальника, залишків, термінів поповнення;
- готує замовлення на закупівлю або внутрішнє переміщення у системі обліку (наприклад, з головного складу на виробничу дільницю).

Інтеграція з базою ІС або іншими ERP-рішеннями дозволяє створювати динамічні та адаптивні схеми: наприклад, якщо певний вид бісеру тимчасово відсутній, програма пропонує заміну з допустимим відхиленням кольору або фактури. Водночас ці зміни фіксуються у технічному описі схеми, що забезпечує точність у виробництві та контролі якості.

Крім поточної перевірки, програмне забезпечення може виконувати аналіз історичних даних, зокрема:

- визначати, які матеріали найчастіше використовуються;
- які кольори залишаються невикористаними;
- як змінюється структура попиту залежно від сезону або типу замовлення.

На основі цих даних система може автоматично формувати прогнози потреб у сировині, оптимізувати складські залишки, мінімізувати надлишкові закупівлі та уникати простоїв через відсутність компонентів.

Для великих підприємств це також відкриває можливість централізованого управління виробництвом і запасами: одна система об'єднує дизайнерів, плановиків, складських працівників, постачальників і фінансистів, забезпечуючи безперервний цикл — від створення схеми до доставки готового набору клієнту.

Крім того, інтегровані рішення дають змогу створити індивідуальний калькулятор вартості в режимі реального часу. Ціни на матеріали оновлюються автоматично, а кінцева вартість картини (включаючи

пакування, інструкції, витрату праці та логістику) розраховується миттєво, що зручно для менеджерів з продажу та клієнтів у B2B і B2C-сегментах.

Загалом, повна інтеграція з обліковими системами дозволяє підвищити операційну ефективність, знизити людський фактор у розрахунках, пришвидшити обробку замовлень і зменшити виробничі втрати, що є критично важливим для підприємств, які працюють у сфері індивідуального або малосерійного виробництва вишивальних картин.

### **Розширення функцій для персоналізованого досвіду:**

Інтерфейси програмного забезпечення для створення схем вишивки стають все більш зручними, гнучкими та інтуїтивно зрозумілими, що дозволяє значно розширити можливості користувачів. Окрім базових функцій, таких як завантаження та обробка зображення, сучасні програми надають численні інструменти для персоналізації досвіду кожного користувача. Однією з таких можливостей є налаштування технік вишивки: користувач може вибрати не лише основну техніку (наприклад, хрестик, напівхрест, бісерне плетіння, гладь), а й налаштувати деталі виконання кожної техніки: розмір стібка, напрямок вишивки, тип вишивальних ниток або бісеру, а також візуальні ефекти (наприклад, ефекти тіні або світлових градієнтів на зображенні).

Такі можливості дозволяють розширити персоналізований досвід, що робить програму корисною як для аматорів, які хочуть експериментувати з новими техніками, так і для професіоналів, яким потрібна точність і контроль на кожному етапі створення схеми. Важливим аспектом є можливість налаштування відображення сітки (наприклад, зміна розміру сітки для різних технік вишивки), що дозволяє налаштувати схему під конкретні вимоги майстра. Крім того, функція перегляду готової картини в 3D-презентації надає користувачам змогу візуалізувати, як виглядатиме готовий виріб, ще до початку вишивання. Цей інструмент дозволяє побачити, як кольори і текстури матеріалів поєднуються, та дає уявлення про кінцевий результат.

Іншим важливим нововведенням є генерація інструкцій для клієнта. Програми дозволяють автоматично створювати детальні інструкції по вишивці, в яких зазначено не тільки техніку і кольори, але й оптимальний порядок роботи, опис кожного етапу, а також рекомендації щодо використання конкретних матеріалів. Це зменшує ймовірність помилок та підвищує ефективність робочого процесу.

Ще однією значною тенденцією є підтримка стандартних форматів обміну даними, таких як XML, JSON, SVG, DXF. Це дозволяє користувачам інтегрувати програму з іншими інструментами та системами. Наприклад, можливість експортувати схеми у форматі SVG або DXF дозволяє використовувати їх для промислових систем автоматизованого друку або розкрою. Таким чином, програма може бути частиною єдиної ланцюжка автоматизації виробництва, де створена схема перетворюється на готовий виріб за допомогою числового програмного управління (ЧПУ). Це значно підвищує ефективність виробничого процесу і зменшує витрати часу на перехід від проектування до виготовлення.

Ще один важливий аспект — можливість обміну даними з іншими дизайнерськими пакетами, що дозволяє об'єднувати програму з більш потужними інструментами для створення складних графічних проектів. Наприклад, програма може бути інтегрована з графічними редакторами, такими як Adobe Illustrator або CorelDRAW, що дає змогу не лише створювати схеми, а й працювати з більш складними візуальними елементами або векторними зображеннями, а також редагувати їх для подальшого використання в вишивці.

Такі функції відкривають нові можливості для співпраці між різними фахівцями (дизайнерами, технологами, виробничими працівниками) і дозволяють адаптувати програму під специфічні вимоги користувача. В результаті, програми нового покоління стають не лише інструментами для створення схем, а й комплексними сервісами, які відповідають вимогам як

аматорів, так і професіоналів в галузі вишивання та виготовлення картин для вишивання бісером.

### **Огляд популярних програм:**

BeadTool — одна з найпоширеніших настільних програм для створення схем вишивання бісером. Підтримує імпорт зображень, автоматичне перетворення, точний підбір кольорів на основі палітр відомих брендів (напр. Miyuki, Preciosa). Має функцію розрахунку кількості бісеру та експорту схем у друкованому вигляді.

EasyBeadPatterns — вебзастосунок з базовим функціоналом, орієнтований на швидке створення простих схем. Підтримує лише кілька форматів, але має мінімалістичний інтерфейс і не потребує встановлення.

BeadCreator Pro — програмне забезпечення професійного рівня з розширеним функціоналом (генерація інструкцій, моделювання готових робіт, підтримка кількох технік вишивки). Вартість програми значно вища, проте вона підтримує роботу з великими проектами.

### **Моделі монетизації:**

Платна ліцензія: BeadTool, BeadCreator Pro — разова покупка або підписка.

Фріміум: Безкоштовна базова версія + платні доповнення або про-функції (Stitch Fiddle).

Повністю безкоштовні рішення: Зазвичай мають обмежений функціонал або рекламу.

### **Популярність та статистика використання:**

BeadTool має понад 100 тис. активних користувачів у США, Канаді та Європі (за відкритими джерелами та оглядами на форумах).

На YouTube налічується понад 500 туторіалів, присвячених BeadCreator Pro, що свідчить про активну спільноту.

Користувачі високо оцінюють зручність експорту схем та підрахунку матеріалів — середній рейтинг на оглядових сайтах — 4,6 з 5.

Таблиця 1 – порівняння програм створення схем

Програма	Платформа	Мова розробки	Підтримка кольорів	Автоматизація	Формат експорту
BeadTool	Windows	C++	Так	Так	PDF, JPEG
BeadCreator Pro	Windows/ Mac	Java	Так (розширено)	Так	PDF, SVG, TXT
EasyBeadPatterns	Web	JavaScript	Обмежено	Частково	PNG
Cross Stitch Creator	Android/i OS	Kotlin/Swift	Так	Так	PNG, PDF

Розвиток програмного забезпечення у цьому сегменті свідчить про загальну тенденцію до автоматизації, спрощення доступу до інструментів для непрофесіоналів, а також орієнтацію на мультиформатність та підтримку індивідуальних вимог користувачів. Враховуючи конкуренцію та динаміку ринку, наступні кроки у розвитку таких рішень, ймовірно, будуть пов'язані з інтеграцією хмарних технологій, штучного інтелекту та елементів доповненої реальності для візуалізації готових виробів перед початком роботи.

## 1.2 Дослідження впливу технології на ринок виробництва

Розвиток технологій автоматизації та цифровізації суттєво змінює виробничі процеси у сфері декоративно-прикладного мистецтва, зокрема у виготовленні картин для вишивання бісером. Створення спеціалізованого програмного забезпечення, яке дозволяє підприємству перетворювати зображення на готову до вишивання схему, розраховувати необхідні матеріали з урахуванням даних із бази 1С, а також оцінювати повну собівартість і рекомендовану роздрібну ціну, має потенціал трансформувати галузь не лише на рівні окремого виробника, а й на рівні ринку загалом.

### 1.2.1 Автоматизація як ключ до масштабування

Одним з основних факторів впливу нової технології є значне підвищення продуктивності виробництва. Замість того щоб вручну

створювати схеми, погоджувати палітру з доступними матеріалами, розраховувати витрати та контролювати запаси — усе це виконується автоматично. Програма самостійно аналізує зображення, формує точну схему, яка відповідає параметрам друку й палітрі бісеру, доступному в базі 1С, підраховує необхідні ресурси (бісер, канву, нитки, упаковку) і формує кошторис на готовий набір. Це кардинально скорочує виробничий цикл — від прийому замовлення до друку готової продукції.

Завдяки скороченню ручної праці підприємство може масштабувати виробництво без пропорційного збільшення персоналу. Замість розширення дизайнерського відділу, достатньо налаштувати систему автоматичної генерації схем і запровадити контроль якості на виході. Це зменшує витрати на заробітну плату та знижує вплив людського фактора на якість.

### **1.2.2 Зниження витрат і контроль за собівартістю**

Інтеграція з 1С — це не просто доступ до списку матеріалів, а повна взаємодія із виробничим, бухгалтерським та логістичним контуром підприємства. Програма аналізує залишки, ціни, постачальників, терміни постачання й обмеження в асортименті, і на цій основі формує оптимальний варіант підбору матеріалів. Це дозволяє:

- уникати надлишкових закупівель або нестачі компонентів;
- миттєво коригувати схеми відповідно до змін у складі доступних ресурсів;
- прогнозувати навантаження на склад і формувати запити постачальникам з урахуванням очікуваного попиту.

Розрахунок повної собівартості, включно з пакуванням, інструкціями, роботою операторів і додатковими витратами, дає змогу точно прогнозувати прибутковість кожного продукту. Визначення рекомендованої роздрібною ціни на основі аналізу собівартості та маржі підвищує цінову стабільність і прибутковість.

### **1.2.3 Покращення гнучкості й швидкості реагування на ринок**

Сучасні споживачі цінують індивідуалізацію та оперативність. Програма, яку розробляється, дозволяє підприємству адаптуватися до таких очікувань. Клієнт може надати будь-яке зображення — і вже за кілька хвилин отримати набір для вишивання, що містить персоналізовану схему, палітру та повний набір матеріалів.

Це відкриває можливість роботи за індивідуальними замовленнями без перевантаження персоналу. Водночас, автоматизоване створення нових дизайнів сприяє швидкому оновленню асортименту продукції, що особливо актуально в умовах сезонного попиту (подарунки, тематичні набори, колекційні випуски тощо).

### **1.2.4 Стандартизація якості та мінімізація браку**

Відсутність людського втручання в технічні параметри схеми — це гарантія стандартизації продукції. Кожна схема створюється згідно з єдиними алгоритмами, а вибір матеріалів автоматизований відповідно до встановлених правил, що знижує ймовірність помилок або несумісності елементів.

Це особливо важливо для підтримання репутації бренду: клієнт, який одного разу отримав набір, може бути впевнений у тому, що всі наступні відповідають тим самим стандартам якості.

### **1.2.5 Нові можливості для аналітики та управління**

Оскільки система фіксує всі дії — від завантаження зображення до виготовлення продукту — вона формує великий масив даних про:

- популярність конкретних зображень, кольорів, стилів;
- ефективність використання ресурсів;
- частоту замовлення певних типів схем;
- маржинальність кожної одиниці продукції.

Ці дані можна використовувати для бізнес-аналітики, планування виробництва, маркетингу та стратегічного управління. Це значно підвищує

ефективність менеджменту, дозволяючи приймати рішення на основі фактів, а не припущень.

### **1.2.6 Зміна ролі персоналу та скорочення часу навчання**

Завдяки інтуїтивному інтерфейсу та автоматизації рутинних завдань змінюється структура кадрових потреб підприємства. Відпадає необхідність у вузькоспеціалізованих фахівцях для створення схем — достатньо мати працівника, який здатен завантажити зображення, задати параметри й проконтролювати результат. Це скорочує час адаптації нових працівників і підвищує загальну продуктивність.

Впровадження технології автоматичного створення схем на основі зображень із одночасним розрахунком матеріалів та собівартості на основі даних з ІС має глибокий і системний вплив на ринок виробництва картин для вишивання. Це не просто програмний продукт — це інструмент трансформації бізнесу, що охоплює виробництво, логістику, маркетинг, облік і стратегічне планування.

Таким чином, підприємство, яке першим впровадить таке рішення, отримає суттєву конкурентну перевагу, зможе вийти на нові ринки, підвищити прибутковість і стабільно розвиватися в умовах зростаючої конкуренції та індивідуалізації попиту.

### **1.3 Оцінка економічної доцільності впровадження програми**

У сучасних умовах зростаючої конкуренції на ринку товарів ручної праці, підприємства, що виготовляють картини для вишивання, стикаються з численними викликами, які вимагають ефективного управління ресурсами та оптимізації виробничих процесів. Зокрема, в умовах обмежених ресурсів та високих вимог до якості продукції, важливо знижувати витрати на виробництво, скорочувати час виготовлення виробів та підвищувати якість кінцевого продукту. Для цього необхідно використовувати сучасні технології, які допоможуть не лише автоматизувати окремі етапи виробничого процесу, а й оптимізувати їх за допомогою програмного забезпечення.

Особливо це актуально для малих і середніх виробників, де кожна одиниця витрат і кожна хвилина часу мають вирішальне значення для загального прибутку. В таких умовах автоматизація процесу створення схем для вишивання стає необхідною умовою для досягнення конкурентоспроможності. Впровадження спеціалізованої програмної системи дозволяє оптимізувати багато аспектів виробничого циклу, зокрема автоматизацію розрахунку матеріалів, налаштування схем, корекцію кольорів та інтеграцію з іншими системами підприємства, такими як облік матеріалів або планування виробництва.

Впровадження програми для автоматизованого створення схем для вишивання має великий потенціал для зниження витрат на виробництво. Одним з найбільших витратних елементів у процесі виготовлення картин є витрати на матеріали (бісер, нитки, канва тощо). За допомогою програмного забезпечення можна значно знизити витрати на закупівлю матеріалів завдяки точному розрахунку необхідної кількості кожного виду матеріалів, а також автоматичному підбору найвигідніших варіантів з урахуванням доступних постачальників. Автоматизація цього процесу дозволяє зменшити ризик надмірних або недостатніх закупівель, що позитивно позначається на ефективності витрат.

Програма також дозволяє оптимізувати використання матеріалів на кожній одиниці продукції, що дозволяє зменшити втрати бісеру або ниток в процесі вишивання. Це забезпечує економію ресурсів і знижує собівартість одиниці продукції. Зниження витрат на матеріали прямо сприяє зростанню прибутковості виробництва.

Іншою важливою перевагою впровадження програмного забезпечення є значне скорочення часу на створення схеми для вишивання. Без автоматизації цей процес зазвичай займає значну кількість часу, оскільки кожна схема створюється вручну, з врахуванням численних деталей і корекцій. Програма здатна автоматично перетворювати зображення в готову схему, що значно скорочує час, необхідний для цього етапу. Це дає змогу

зосередитись на інших аспектах виробництва, таких як налаштування кольорових палітр, вибір матеріалів та оптимізація виробничих потоків.

Програма дозволяє значно скоротити час, необхідний для розрахунку кількості бісеру, ниток, канви, що також позитивно впливає на швидкість виконання замовлень та дозволяє збільшити кількість продукції, що виготовляється за певний період часу.

Точність та якість кінцевого продукту є ключовими факторами на ринку товарів ручної праці. Впровадження спеціалізованої програми для створення схем дозволяє значно підвищити точність схем вишивки, що знижує ймовірність помилок та підвищує якість готової продукції. Автоматизоване визначення кольору, розміру стібка, типу матеріалів гарантує точне відтворення кожного елемента схеми, що важливо для досягнення високої якості готових картин. В результаті підвищується довіра клієнтів, що позитивно позначається на репутації підприємства та стимулює зростання попиту.

Інтеграція програмного забезпечення з системами обліку, такими як 1С, SAP або Odoo, дозволяє автоматично управляти складськими запасами. Завдяки інтеграції з базами даних, програма може забезпечити своєчасне поповнення матеріалів, контролювати їх запаси та автоматично формувати заявки на закупівлю матеріалів, коли рівень запасів досягає мінімального значення. Це допомагає зменшити витрати на складування та запобігти дефіциту матеріалів, що, в свою чергу, може призвести до затримки в виробництві або невиконання замовлень вчасно.

Ще однією важливою економічною перевагою є покращення фінансового контролю та прогнозування завдяки інтеграції програми з системами обліку та фінансового управління. Програма дозволяє отримувати аналітичні дані про витрати на матеріали, витрати на робочу силу, собівартість одиниці продукції та інші важливі фінансові показники. Це дозволяє керівникам підприємства більш точно планувати бюджети, визначати прибутковість та прогнозувати майбутні витрати. В результаті стає

можливим більш ефективно використання фінансових ресурсів та оптимізація витрат на всіх етапах виробництва.

Впровадження програми для автоматизації створення схем для вишивання дозволяє підприємствам бути більш гнучкими та оперативними на ринку. Завдяки автоматизації значно підвищується швидкість виконання замовлень, що дозволяє вийти на нові ринки та знижує час, необхідний для виконання індивідуальних замовлень. Програма дозволяє швидше адаптуватися до змінюваних вимог ринку, що підвищує конкурентоспроможність підприємства в умовах високої конкуренції.

В результаті реалізації цього проекту підприємство отримає значну економічну вигоду завдяки зниженню витрат, підвищенню ефективності виробництва та забезпеченню високої якості готової продукції. Технології автоматизації створення схем для вишивання дозволяють підприємствам бути більш конкурентоспроможними, що в свою чергу сприяє збільшенню ринкової частки та підвищенню загальної прибутковості бізнесу.

### **1.3.1 Скорочення витрат, підвищення продуктивності та якісна оптимізація ресурсів**

Основною перевагою впровадження програми є скорочення витрат на ручну працю, усунення дублювання дій та зниження рівня помилок у процесі створення схем для вишивання. Раніше такий процес займав значні трудові ресурси, особливо коли йшлося про індивідуальні замовлення. Програма ж дозволяє автоматично перетворити цифрове зображення у схему для вишивання, враховуючи формат, роздільну здатність, палітру кольорів та інші важливі параметри. Це скорочує час на розробку однієї картини в 5–10 разів.

Завдяки інтеграції з системою 1С відбувається автоматичний підбір потрібних матеріалів (бісеру, ниток, тканини, рамок тощо) на основі актуальних даних складу. Таким чином, підприємство уникає зайвих закупівель, зменшує обсяги нереалізованих залишків і може оперативно планувати поповнення запасів лише у разі потреби.

Також відбувається суттєве зниження витрат на навчання персоналу — замість складного процесу створення схем вручну достатньо навчити працівників користуватися програмою, що значно дешевше, швидше та ефективніше. Усе це забезпечує підвищення продуктивності праці, особливо в умовах великого потоку замовлень або сезонного збільшення попиту.

### **1.3.2 Підвищення гнучкості виробництва, адаптація до ринку та індивідуалізація продукту**

Сучасні споживачі дедалі частіше прагнуть до індивідуалізації товарів, і ринок вишивання не є винятком. Завдяки програмі підприємство отримує можливість створювати картини за індивідуальними запитамі клієнтів (фото, портрети, пейзажі тощо), що раніше вимагало складної ручної підготовки. Такий функціонал відкриває нові ринкові ніші, наприклад, подарункову продукцію, іменні схеми, дизайнерські колекції тощо.

Гнучкість програми дозволяє швидко вносити зміни до схеми — змінити формат, розміри, тип бісеру, кольорову гаму — без повторного створення з нуля. Таким чином, навіть при незначних змінах у вподобаннях клієнтів чи сезонному попиту підприємство оперативно перебудовується без втрати ефективності.

Крім того, система дозволяє проводити А/В-тестування нових моделей картин: запускати обмежені серії, оцінювати продажі, виявляти популярні дизайни. Це мінімізує ризики при запуску нових продуктів та робить управління асортиментом більш гнучким і обґрунтованим.

### **1.3.3 Оптимізація фінансового контролю та підвищення рентабельності**

Завдяки точному автоматичному розрахунку собівартості кожного виробу, програма дозволяє формувати обґрунтовану систему ціноутворення, що враховує усі складові вартості — від вартості матеріалів до витрат на пакування, збирання комплекту та логістику. Це дає змогу уникати як недооцінки (збиткових цін), так і переоцінки (втрата клієнтів через завищену вартість).

Інструменти звітності, інтегровані у програму, дозволяють керівництву бачити повну картину витрат і прибутків у режимі реального часу, здійснювати планування закупівель, прогнозування доходів та оперативно реагувати на фінансові відхилення.

У довгостроковій перспективі впровадження програми сприяє підвищенню рентабельності підприємства, зростанню обсягів реалізації, покращенню репутації серед партнерів та клієнтів, а також залученню нових ринків збуту завдяки можливості швидко масштабувати виробництво.

#### **1.3.4 Термін окупності, економічні ризики та потенціал масштабування**

Згідно з проведеними розрахунками, орієнтовний термін окупності проєкту — від 6 до 12 місяців при середньому навантаженні та обсягах виробництва. У великих підприємствах цей термін може бути ще меншим за рахунок ефекту масштабу. Факторами, що прискорюють окупність, є: велика кількість замовлень, висока частка індивідуальних схем, наявність розгалуженої бази клієнтів, ефективна логістика.

Можливі економічні ризики (наприклад, зменшення попиту, коливання цін на сировину, інфляція) частково нівелюються завдяки автоматизованим прогнозам попиту і витрат, які формуються в системі на основі статистичних даних з ІС. Це дозволяє заздалегідь виявляти тенденції, що загрожують стабільності виробництва, і адаптувати стратегію розвитку.

Програма також має великий потенціал для масштабування: її можна адаптувати під великі виробництва, доповнити новими функціями (наприклад, онлайн-каталог, автоматичне формування інструкцій для клієнтів, генерація QR-кодів для схем), а також інтегрувати з іншими ERP-системами, CRM або платформами електронної комерції.

## **1.4 Формулювання цілей та формулювання технічного завдання на розробку**

### **1.4.1 Цілі розробки програмного забезпечення**

Метою даного проєкту є створення ефективного інструменту автоматизації процесу виготовлення схем для вишивання на підприємстві, яке спеціалізується на виробництві наборів для рукоділля. Програмне забезпечення має оптимізувати технологічні та облікові процеси, зменшити трудовитрати, прискорити підготовку продукції до виготовлення, а також підвищити точність розрахунків необхідних матеріалів і фінансової складової виробництва.

Головною стратегічною ціллю є цифрова трансформація процесу виготовлення індивідуальних та серійних картин для вишивання з метою підвищення продуктивності праці, зменшення кількості помилок у підрахунках, зниження собівартості та поліпшення обслуговування клієнтів.

#### **Конкретні цілі проєкту:**

- Автоматизація перетворення зображення у схему для вишивання відповідно до заданих параметрів (розмір, формат, кількість кольорів, тип бісеру);
- Інтеграція з базою даних 1С для автоматичного підбору матеріалів на складі;
- Розрахунок кількості матеріалів, собівартості набору та формування попередньої вартості готової картини з урахуванням пакування та додаткових витрат;
- Зменшення часу підготовки одного виробу та підвищення точності фінансового і ресурсного планування;
- Підвищення конкурентоспроможності підприємства за рахунок індивідуалізації замовлень та оперативної обробки нових замовлень.

### 1.4.2 Технічне завдання на розробку

Для реалізації поставлених цілей визначено технічне завдання, яке включає перелік функціональних, програмних і інтеграційних вимог до майбутнього програмного забезпечення. Нижче наведено основні вимоги:

#### 1. Вхідні дані:

- Файли зображень у форматах PNG, JPEG, BMP;
- Налаштування користувача: бажаний розмір готової картини, щільність бісеру, тип палітри;
- Дані з 1С: наявність і залишки матеріалів на складі, ціни закупівлі та реалізації.

#### 2. Функціональність програми:

- Автоматичне створення схеми з пікселізацією зображення за заданими параметрами;
- Підбір палітри бісеру з урахуванням доступності на складі (зв'язок з 1С);
- Генерація технічної карти вишивки (колірна таблиця, кількість бісеру кожного кольору);
- Розрахунок загальної кількості необхідних матеріалів: тканини, бісеру, ниток, голок, інструкцій, упаковки тощо;
- Автоматичний розрахунок вартості комплекту: матеріали + упаковка + умовна вартість роботи;
- Формування друкованої документації: схема, легенда бісеру, список матеріалів, цінник;
- Збереження даних по кожному проекту у базі.

#### 3. Вимоги до програмної реалізації:

- Мова програмування: Python/C#/або інша з урахуванням технологій підприємства;
- Інтерфейс користувача: графічний, інтуїтивно зрозумілий, з можливістю попереднього перегляду схеми;

- Можливість ручного коригування (наприклад, редагування кольорів вручну);
- Підтримка багатокористувацького режиму (для кількох працівників одночасно).

#### 4. Інтеграція з 1С:

- Читання даних про залишки матеріалів і їх ціни;
- Можливість експорту результатів у формат, сумісний з 1С (наприклад, Excel, CSV, JSON або пряма передача через API);
- Можливість оновлення залишків після формування комплекту (опціонально).

#### 5. Безпека та збереження даних:

- Резервне копіювання даних;
- Обмеження доступу на основі ролей користувачів;
- Захист від втрати інформації при аварійному завершенні роботи програми.

#### 6. Очікуване середовище використання:

- Операційна система: Windows 10 або вище;
- Сумісність з внутрішньою мережею підприємства;
- Робота на типових ПК офісного рівня без потреби у високопродуктивному обладнанні.

## **РОЗДІЛ 2. РОЗРОБКА ТА ПЛАНУВАННЯ ПРОЄКТУ**

### **2.1 Обґрунтування вибору методології управління проєктом**

Успішне виконання проєкту з розробки інформаційної системи значною мірою залежить від правильно обраної методології управління. Оскільки даний проєкт передбачає створення програмного забезпечення з декількома взаємопов'язаними модулями, гнучкою логікою обробки зображень та інтеграцією з існуючими внутрішніми системами (зокрема 1С), важливим є не лише технічний аспект, а й організація процесу розробки.

Класична каскадна модель (Waterfall) передбачає суворо послідовне проходження усіх етапів розробки – від формування вимог, проєктування, кодування, тестування до впровадження і супроводу. Такий підхід історично застосовувався у великих інженерних або державних проєктах, де завдання чітко визначені з самого початку, а зміни або уточнення вимог практично виключені. Однак в умовах сучасного динамічного середовища розробки програмного забезпечення, особливо у сфері прикладних корпоративних рішень, ця модель виявляється недостатньо гнучкою.

Для проєкту, метою якого є створення програми для автоматизації створення схем вишивки з урахуванням графічного аналізу зображення, інтеграції з базами 1С, розрахунку матеріалів і собівартості, каскадний підхід має ряд критичних обмежень. Насамперед, потреби бізнесу та користувачів можуть змінюватися в ході реалізації, наприклад:

- виявлення нових особливостей в обробці нетипових зображень;
- потреба в розширенні списку матеріалів або зміні структури їх обліку;
- уточнення логіки розрахунку собівартості залежно від пакування або сезонних знижок.

У Waterfall-моделі внесення таких змін на пізніх етапах проєкту (після завершення фази проєктування чи початку розробки) часто потребує повного перегляду технічної документації, що веде до втрати часу, перевитрат бюджету і затримок у реалізації. До того ж, в такій моделі користувач бачить

програму вже на фінальному етапі, коли змінити щось фундаментально дуже складно або економічно недоцільно.

Оскільки розробка даного застосунку орієнтована на інтенсивну взаємодію між замовником і командою розробки, важливо мати постійний доступ до інтерфейсу, регулярно тестувати функціонал, оцінювати результати автоматичної генерації схем, точність зіставлення з палітрою, інтеграцію з 1С тощо. У таких умовах жорстка структура каскаду створює ризики втрати зворотного зв'язку та невідповідності очікувань замовника кінцевому результату.

Також недоліком Waterfall є низька адаптивність до невизначеності в специфікації — на початку проєкту складно передбачити всі нюанси реалізації обробки зображень, варіативність технік вишивання чи потреби з боку бухгалтерського обліку. Якщо ж ці особливості виявляються після початку реалізації, у каскадній моделі немає інструментів швидкої адаптації чи переорієнтації.

На практиці це може призвести до ситуацій, коли:

- розроблений функціонал виявляється неефективним або надмірно складним для користувача;
- інтерфейс не відповідає очікуванням замовника;
- інтеграція з 1С відбувається поверхнево або без урахування поточних бізнес-процесів компанії.

Незважаючи на свою структурованість і передбачуваність, каскадна модель не забезпечує необхідного рівня гнучкості, адаптивності та швидкого реагування, що є критично важливим у випадку розробки прикладного програмного забезпечення для індивідуальних виробничих процесів, таких як виготовлення схем для вишивки на замовлення.

У контексті розробки програмного забезпечення, що має тісну інтеграцію з бізнес-процесами підприємства, важливо забезпечити оперативну адаптацію до змін, постійний контроль якості, ефективну комунікацію між розробниками та замовником, а також гнучкість

масштабування. Усі ці потреби найкраще задовольняє Agile-підхід, зокрема Scrum — одна з найбільш перевірених у практиці форм організації командної роботи в ІТ-проектах.

### **Ключові переваги застосування Scrum у цьому проєкті:**

#### **- Ітеративна розробка та швидке отримання результатів**

Реалізація проєкту ділиться на спринти — короткі періоди (зазвичай 1–2 тижні), під час яких команда розробляє та презентує замовнику функціонально завершену частину системи. Така структура дозволяє:

- чітко планувати робочі задачі відповідно до пріоритетів бізнесу;
- уже на ранніх етапах мати робочі прототипи системи (наприклад, модуль імпорту зображень, генератор схеми, калькулятор собівартості тощо);
- швидко демонструвати прогрес проєкту зацікавленим сторонам;
- реалізовувати складну систему поетапно, з акцентом на найважливіших компонентах.

#### **- Регулярна участь замовника і постійний зворотний зв'язок**

У Scrum передбачено активне залучення Product Owner'a (представника замовника), який надає технічне бачення, перевіряє результат кожного спринту, вносить зміни або доповнення у вимоги. Це особливо важливо у вашому проєкті, де:

- потрібне чітке розуміння логіки роботи з базою 1С;
- функціональність калькуляції та підбору матеріалів має бути адаптована до специфіки виробництва;
- інтерфейс має бути зручним для майстрів, які не завжди мають ІТ-компетенції.

#### **- Постійне тестування та контроль якості**

У Scrum тестування не відкладається на завершальну стадію проєкту, а виконується на кожному етапі розробки. Це дозволяє:

- виявляти помилки й недоліки на ранніх стадіях, коли їх усунення менш затратне;

- перевіряти працездатність окремих модулів — наприклад, точність генерації кольорової сітки, правильність розрахунку матеріалів, коректність імпорту/експорту даних;
- покращити якість системи через регулярні перевірки інтеграції модулів між собою.

#### - **Гнучкість у зміні вимог**

Agile передбачає, що вимоги не є фіксованими, а можуть еволюціонувати відповідно до нового бачення замовника або змін ринкової ситуації. Наприклад:

- якщо підприємство змінює постачальника бісеру і потрібно оновити логіку відповідності кольорів;
- якщо вводиться нова система знижок на пакування — модуль розрахунку собівартості легко адаптується;
- можна додати підтримку нових форматів зображень або розширити алгоритми AI-обробки.

#### - **Масштабування команди та проєкту**

Scrum добре масштабується: команда може зростати разом із проєктом, додаватися нові спеціалісти — наприклад, для розробки мобільної версії, інтеграції з додатковими платформами чи створення навчальних модулів. Це дозволяє:

- збільшити швидкість реалізації без втрати якості;
- делегувати окремі модулі (наприклад, аналітику або експорт даних) спеціалізованим фахівцям;
- гнучко розширювати функціональність у відповідь на нові ринкові потреби.

#### - **Прозорість процесу та мотивація команди**

Щоденні короткі наради (daily stand-up), наочні борди завдань (Scrum Board, Trello, Jira тощо) та ретроспективи спринтів забезпечують високу прозорість роботи команди й дозволяють усувати “вузькі місця” в процесі.

Команда постійно бачить прогрес, а замовник — має розуміння, на якому етапі знаходиться реалізація.

Підсумовуючи, Scrum-методологія ідеально підходить для реалізації вашого програмного продукту, оскільки:

- враховує потребу у тісній співпраці з підприємством;
- підтримує високу гнучкість і швидку адаптацію функціоналу;
- дозволяє мінімізувати ризики, пов'язані з невизначеністю;
- забезпечує поступове розгортання продукту, починаючи з MVP і завершуючи повноцінною системою.

Проект із розробки програмного забезпечення для створення схем вишивання та розрахунку собівартості охоплює широкий спектр технічно різноманітних завдань, які потребують залучення фахівців різного профілю: розробників алгоритмів комп'ютерного зору, спеціалістів з інтеграції з обліковими системами (наприклад, 1С або SAP), дизайнерів UI/UX, інженерів баз даних, а також аналітиків і тестувальників. Такий міждисциплінарний характер проекту створює потенційні виклики у координації командної роботи, управлінні залежностями між задачами та забезпеченні узгодженості функціоналу.

Застосування Agile-підходу на базі Scrum дозволяє ефективно організувати роботу такої гетерогенної команди завдяки:

- чіткому структуруванню роботи по ітераціях;
- регулярному обміну інформацією між учасниками;
- швидкому виявленню й усуненню проблем у процесі розробки;
- гнучкому впровадженню нових ідей та змін без ризику для цілісності проекту.

Кожен **спринт** (тривалістю зазвичай 2–3 тижні) передбачає проходження п'яти основних етапів:

- **Планування спринту (Sprint Planning)**

На цьому етапі команда разом із Product Owner'ом (представником замовника) визначає:

- цілі поточного спринту — наприклад, реалізація модуля обробки зображень, базової кольорокорекції або початкова інтеграція з 1С;
- конкретні задачі, що мають бути виконані (розбивка на підзадачі, оцінка складності та часу);
- критерії приймання результату (Definition of Done) — тобто, які саме параметри має задовольняти реалізований функціонал.

#### - **Розробка (Development Work)**

Команда приступає до виконання запланованих задач, серед яких можуть бути:

- створення алгоритмів сегментації зображення та перетворення у схему з бісеру;
- налаштування логіки розрахунку витрат на основі цін із бази даних;
- реалізація елементів інтерфейсу, зокрема редактора палітри або інтерактивного перегляду схеми;
- написання unit-тестів для перевірки функціональної коректності.

Використання системи керування завданнями (наприклад, Jira, ClickUp або Trello) дозволяє візуалізувати прогрес, трекати час і розподіляти відповідальність.

#### - **Щоденні зустрічі (Daily Stand-Ups)**

Кожного робочого дня команда проводить коротку (15–20 хвилин) нараду, на якій кожен учасник відповідає на три ключові питання:

- Що було зроблено вчора?
- Що буде зроблено сьогодні?
- Чи виникли перешкоди або труднощі?

Це дозволяє:

- оперативно реагувати на затримки або технічні складнощі;
- синхронізувати зусилля фахівців із різних напрямів;
- тримати всю команду в одному інформаційному полі.

#### - **Демонстрація результатів (Sprint Review)**

Після завершення спринту команда презентує замовнику реалізований функціонал. Наприклад:

- як працює модуль генерації схеми;
- як відображаються доступні палітри бісеру;
- як формується звіт із собівартості матеріалів.

Замовник або його представник може надати відгук, висловити зауваження чи пропозиції. Це дозволяє гнучко адаптувати наступні задачі до змін у вимогах або пріоритетах.

#### - Ретроспектива (Sprint Retrospective)

На завершення спринту команда збирається для внутрішнього аналізу:

- що було зроблено добре;
- що потребує покращення;
- які рішення можна прийняти для підвищення ефективності наступного спринту.

Ретроспектива сприяє зростанню командної згуртованості, підвищенню продуктивності та уникненню повторення організаційних або технічних помилок.

Завдяки чітко структурованому, ітеративному підходу Agile/Scrum, команда розробки може:

- ефективно взаємодіяти навіть у складному, багатокomпонентному середовищі;
- швидко реагувати на зворотний зв'язок замовника;
- забезпечити поступове, але стабільне зростання функціональності системи;
- отримати повноцінну та конкурентоспроможну програму за контрольованих ресурсів і термінів.

Крім того, важливо відзначити, що використання гнучкої методології дозволить впровадити поетапне тестування на реальних виробничих процесах, що особливо актуально для такого підприємства, яке одночасно виконує багато унікальних замовлень. Завдяки цьому можливо своєчасно

виявити вузькі місця, знизити ризики помилок при обробці нестандартних зображень, та забезпечити високу точність розрахунків матеріалів.

Таким чином, обраний підхід дозволяє не лише мінімізувати ризики та забезпечити контроль над термінами, а й гнучко адаптувати систему до потреб підприємства в умовах постійно змінного попиту та структури замовлень.

## **2.2 Планування проєкту та кошторисні розрахунки**

Реалізація програмного забезпечення відбуватиметься у кілька етапів, згідно з принципами гнучкої методології управління (Scrum), з урахуванням технічної складності та міжфункціональної координації.

### **1. Підготовчий етап (1 тиждень)**

- Формування проєктної команди (розробник, системний аналітик, дизайнер UI/UX, тестувальник).
- Аналіз бізнес-вимог підприємства.
- Узгодження специфікацій з технічним і комерційним відділами.
- Визначення критеріїв успішності впровадження.

### **2. Архітектурне та технічне проєктування (2 тижні)**

- Вибір технологічного стеку.
- Проєктування структури БД.
- Опис модульної взаємодії (імпорт зображення → генерація схеми → розрахунок → вивід → інтеграція з 1С).
- Планування API взаємодій із внутрішніми системами.

### **3. Розробка MVP-версії (4 тижні)**

- Створення модуля імпорту та обробки зображення.
- Реалізація базового алгоритму перетворення в схему.
- Вбудована система кольорокорекції.
- Інтерфейс для перегляду результату.

### **4. Інтеграція з 1С та обліковими системами (2 тижні)**

- Розробка модуля підбору матеріалів із бази даних.
- Синхронізація залишків на складі.

- Формування калькуляцій собівартості.

## 5. Тестування та доопрацювання (2 тижні)

- Модульне, інтеграційне та користувацьке тестування.
- Виправлення помилок, оптимізація функціоналу.
- Підготовка технічної документації.

## 6. Запуск та супровід (після розробки)

- Навчання персоналу.
- Моніторинг ефективності.
- Випуск оновлень на основі зворотного зв'язку.

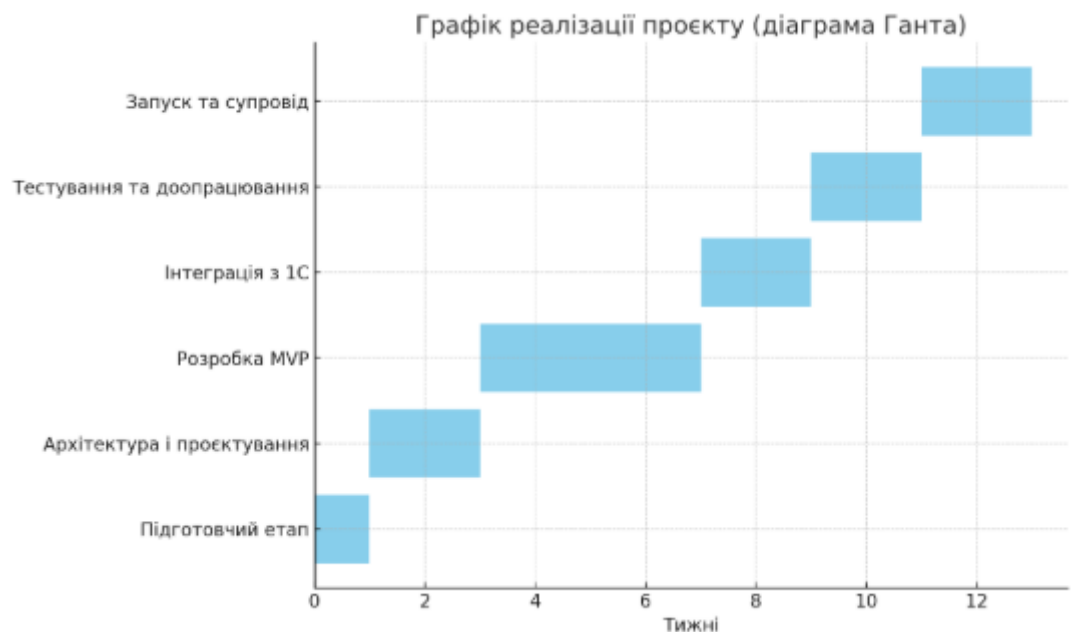


Рисунок 1 – Загальна діаграма кошторисів проєкту

### 2.3 Команда проєкту

Для успішної реалізації проєкту зі створення програмного забезпечення, що автоматизує процес створення схем для вишивання та виконує розрахунок матеріалів і собівартості продукції, необхідна мультидисциплінарна команда. До її складу мають входити фахівці з різних галузей: програмування, комп'ютерного зору, UX/UI-дизайну, систем інтеграції з обліковим ПЗ (1С), тестування, а також менеджменту проєктів.

Таблиця 2 - Загальний склад команди проєкту та її фінансуванн

Посада	Кількість осіб	Загальний час роботи (годин)	Орієнтована ставка (грн/год)	Загальна зарплатня (грн)
Керівник проєкту (PM)	1	240	500	120 000
Бізнес-аналітик	1	160	450	72 000
Провідний розробник	1	240	600	144 000
Backend-розробник	1	200	500	100 000
Frontend-розробник	1	160	450	72 000
Інженер схеми алгоритму	1	180	600	108 000
Інженер з інтеграції (IC)	1	160	500	80 000
UX/UI дизайнер	1	120	400	48 000
Тестувальник (QA)	1	150	400	60 000
<b>Разом</b>				<b>804 000 грн</b>

### **Обов'язки та компетенції членів команди**

#### **- Керівник проєкту (Project Manager):**

Відповідає за планування, координацію, контроль строків, бюджету та ресурсів проєкту. Проводить спринт-планування, модернує щоденні стендапи та ретроспективи.

Необхідні навички: управління проєктами (Scrum/Agile), стратегічне мислення, комунікації.

#### **- Бізнес-аналітик:**

Збирає вимоги від замовника, формує технічну специфікацію, описує бізнес-процеси. Створює логіку взаємодії між модулем створення схеми та обліковими системами.

Необхідні навички: BPMN, створення Use Case, аналіз систем, володіння 1С-логікою.

**- Провідний розробник:**

Приймає архітектурні рішення, координує роботу розробників, відповідає за якість коду. Контролює інтеграцію між модулями, вибирає технологічний стек.

Необхідні навички: досвід роботи з архітектурою ПЗ, керування командою, знання Python/JS.

**- Backend-розробник:**

Реалізує логіку обробки зображення, створення схем, збереження даних, API-комунікацію. Відповідає за реалізацію модуля розрахунку собівартості та інтеграцію з базами.

Необхідні навички: Python, Django/Flask, REST API, робота з базами даних.

**- Frontend-розробник:**

Створює веб-інтерфейс для користувача: імпорт зображення, вибір палітр, попередній перегляд схеми. Впроваджує інтерактивність і адаптивний дизайн.

Необхідні навички: HTML/CSS, JavaScript, React/Vue.

**- Інженер схеми алгоритму:**

Розробляє алгоритми перетворення зображення у схему з урахуванням форми, кольору, тіней.

Необхідні навички: OpenCV, TensorFlow/PyTorch, Python, обробка зображень.

**- Інженер з інтеграції (1С API Specialist):**

Забезпечує зв'язок програмного продукту з обліковою системою 1С. Реалізує двосторонній обмін даними (матеріали, ціни, залишки).

Необхідні навички: 1С:Підприємство, робота з XML/JSON API, інтеграційна логіка.

**- UX/UI дизайнер:**

Розробляє зручний, інтуїтивний інтерфейс для користувача. Готує макети, прототипи, бере участь у юзабіліті-тестуваннях.

Необхідні навички: Figma, Adobe XD, основи взаємодії людина-комп'ютер.

**- Тестувальник (QA Engineer):**

Створює сценарії тестування функціональності, виконує автоматизоване й ручне тестування. Фіксує баги, забезпечує якість продукту на кожному спринті.

Необхідні навички: тестування вебзастосунків, Selenium/Postman, написання test cases.

Таблиця 3 – Розподіл робіт між членами команди

<b>Етап розробки</b>	<b>Відповідальні спеціалісти</b>
Збір вимог, проектування	Бізнес-аналітик, Project Manager
Архітектура, постановка задач	Team Lead, Backend, CV Engineer
Реалізація модулів	Backend, Frontend, CV Engineer, ERP Engineer
UX/UI дизайн	UX/UI дизайнер
Інтеграція з 1С	ERP Engineer, Backend
Тестування	QA Engineer
Документування	Бізнес-аналітик, Технічний письменник
Демо, адаптація за відгуками	Уся команда, під керівництвом Project Manager

## **2.4 Розрахунко витрат проєкту**

У межах реалізації проєкту, окрім витрат на оплату праці членів команди, передбачається низка супутніх фінансових витрат, безпосередньо пов'язаних із забезпеченням належних умов для виконання робіт. До таких витрат належать:

### **1. Оренда приміщення**

Для забезпечення ефективної командної роботи, організації регулярних зустрічей, планування діяльності, тестування та демонстрації функціоналу необхідно орендувати офісне приміщення. Офіс забезпечить зручне робоче середовище з доступом до стабільного інтернет-зв'язку, електроживлення та базової інфраструктури (робочі місця, оргтехніка тощо), що є особливо важливим на етапах розробки та інтеграційного тестування.

### **2. Платне програмне забезпечення**

З метою дотримання законодавства щодо авторських прав, а також для підвищення ефективності роботи, передбачено використання ліцензійного програмного забезпечення. Серед іншого, це середовище розробки (наприклад, JetBrains IDE або Visual Studio Professional), засоби графічного проєктування інтерфейсів (Figma, Adobe XD) та програмні рішення для управління проєктами (Jira, Notion тощо). Наявність ліцензійного ПЗ сприятиме безперебійній та безпечній розробці системи.

### **3. Сервер для тестових робіт**

На етапах модульного та інтеграційного тестування доцільно використовувати окремий сервер або хмарний обчислювальний ресурс (наприклад, VPS), на якому буде розгортатися поточна версія програмного забезпечення. Це дасть змогу здійснювати віддалений доступ до продукту, тестувати API, оцінювати швидкодію та забезпечувати паралельну роботу над різними модулями системи.

### **4. Ноутбуки для команди**

З урахуванням можливої відсутності в окремих учасників команди належної техніки, доцільно передбачити закупівлю або короткострокову оренду ноутбуків для основних спеціалістів проєкту: розробника, дизайнера, системного аналітика та тестувальника. Це дозволить уніфікувати технічне забезпечення та мінімізувати ризики, пов'язані з несумісністю інструментів.

## 5. Канцелярські витрати

Попри цифровий характер проєкту, передбачаються мінімальні витрати на офісні матеріали: друк документації, блокноти, письмове приладдя, засоби візуалізації (маркери, фліпчарти), необхідні для планування спринтів, узагальнення зворотного зв'язку, створення інструкцій та супровідної документації.

## 6. Резерв на випадок форс-мажорів

У структурі бюджету проєкту передбачено резервний фонд у розмірі 10% загальної суми інших витрат. Він покриває можливі форс-мажорні ситуації, зокрема поломки обладнання, затримки поставань, необхідність термінових доопрацювань або змін у функціональних вимогах. Наявність такого резерву є важливим елементом ризик-менеджменту проєкту.

Таблиця 4 – Розрахунок витрат проєкту

Стаття витрат	Кількість / Одиниці	Ціна за одиницю, грн	Загальна вартість, грн	Примітка
Оренда приміщення	3 місяці × 1 офіс	20 000	60 000	Простір на 4–5 робочих місць
Платне ПЗ (ліцензії)	4 пакети × 3 міс	1 200	14 400	ІДЕ, графіка, трекінг завдань
Сервер для тестування	3 міс. VPS	700	2 100	Хостинг + база даних

Ноутбуки	4 шт. , оренда	25 000	100 000	Середня конфігурація для розробки
Канцелярські витрати	Загальна оцінка	—	2 000	Ручки, папір, роздруківки, маркери
Резервний фонд	~10% від загального	—	93 000	Непередбачені витрати
Заробітні плати	~10% від загального	—	804 000	До уваги взято середні заробітні плати на ринку
<b>Усього</b>	<b>1 075 500 грн</b>			

## 2.5 SCRUM аналіз

Розробка програмного забезпечення для автоматизації створення схем вишивки та розрахунку собівартості продукції була організована згідно з методологією Scrum, що дозволяє структурувати проєкт як послідовність коротких ітерацій (спринтів), кожна з яких має завершений результат і може бути самостійно оцінена.

### I. Формування беклогу продукту (Product Backlog)

На початковому етапі було сформовано беклог продукту – перелік функціональних вимог до системи, які повинні бути реалізовані для досягнення кінцевої мети. До беклогу увійшли такі ключові функціональні компоненти:

- модуль імпорту графічних зображень;
- модуль генерації вишивальної схеми (з кольоровою мапою);
- система автоматичного підбору матеріалів на основі палітри;
- алгоритм розрахунку собівартості (на основі інтеграції з базами 1С);
- інтерфейс для візуалізації готової схеми;

- експорт результатів та формування технічної документації для друку;
- базова система керування користувачами.

Кожен елемент беклогу було деталізовано у вигляді користувацьких історій (user stories), що описують очікувану поведінку системи з точки зору кінцевого користувача.

## **II. Планування першого спринту (Sprint Planning)**

Під час планування першого спринту було визначено задачі, які забезпечують побудову технічного каркасу системи:

- налаштування середовища розробки;
- створення шаблону проєкту;
- реалізація модуля завантаження зображень;

первинна обробка зображення та формування піксельної мапи.

Також було встановлено орієнтовну тривалість кожного спринту — 2 тижні.

## **III. Ролі в команді**

Команда складалася з наступних ключових учасників:

- Product Owner – формував бачення продукту, пріоритети беклогу, приймав результати кожного спринту;
- Scrum Master – забезпечував дотримання процесів Scrum, організовував зустрічі, координував командну роботу;
- Development Team – відповідала за реалізацію функціоналу, тестування, інтеграцію.

Команда була крос-функціональною: кожен член мав змогу працювати з декількома модулями, що підвищувало гнучкість і швидкість реалізації.

## **IV. Проведення щоденних зустрічей (Daily Stand-ups)**

Щоденні короткі наради дозволяли команді:

- інформувати про прогрес роботи;
- виявляти труднощі на ранніх етапах;
- приймати мікрорішення щодо змін у плані;
- перерозподіляти завдання залежно від обставин.

Це дозволяє підтримувати постійну синхронізацію команди та забезпечити прозорість розробки.

#### **V. Ретроспективи спринтів (Sprint Retrospectives)**

Після завершення кожного спринту команда проводила ретроспективу, під час якої:

- аналізувались технічні труднощі;
- виявлялися ефективні та неефективні практики;
- формувались пропозиції щодо оптимізації процесів;
- приймалися рішення щодо змін у майбутніх спринтах.

У першій ретроспективі було вирішено удосконалити систему документування вимог і впровадити фреймворк для автоматизованого тестування, що зменшило навантаження на ручну перевірку у наступних ітераціях.

#### **VI. Демонстрація результатів (Sprint Review)**

Після кожного спринту відбувалась демонстрація готового функціоналу за участі Product Owner'a. Це дозволяло:

- отримати оперативний зворотний зв'язок;
- скоригувати пріоритети наступного спринту;
- приймати або відхиляти реалізовані елементи беклогу.

Наприклад, після другого спринту було запропоновано спростити інтерфейс завантаження зображень на основі коментарів замовника.

#### **VII. Ревізія беклогу**

Після кожного спринту здійснювалась ревізія Product Backlog:

- видалялися вже реалізовані пункти;
- додавалися нові задачі, виявлені в процесі розробки;

змінювались пріоритети функціоналу відповідно до очікувань користувачів.

Це дозволяло адаптувати стратегію розробки в умовах змін і уточнень.

### **2.6 Концептуальна модель програми**

Програмне забезпечення для автоматизованого створення схем вишивання бісером розроблено з метою оптимізації та цифровізації всіх

ключових етапів, пов'язаних із проєктуванням схем, підбором відповідних матеріалів, а також обліком і адмініструванням користувачів. Система орієнтована як на індивідуальних майстрів, так і на виробничі підприємства, що займаються виготовленням комплектів для вишивки або персоналізованих схем за замовленням.

Програма забезпечує:

- **Автоматизоване перетворення будь-якого графічного зображення у схему для вишивання бісером**, з можливістю налаштування параметрів (розмір, щільність сітки, палітра кольорів, вид бісеру тощо);
- **Інтерактивний вибір матеріалів** (бісер, канва, нитки, фурнітура) з бази даних, яка оновлюється в режимі реального часу на основі наявності на складі або довідників з 1С;
- **Побудову інструкції для вишивання**, яка включає легенду кольорів, кількість матеріалів, зображення готового виробу тощо;
- **Управління обліковими записами користувачів**, ролями, історією замовлень, персональними налаштуваннями та статистикою використання;
- **Інтеграцію з бухгалтерськими та обліковими системами (1С, SAP, інші через API)** для автоматизації логістики та формування кошторисів.

Хоча система не використовує штучний інтелект або складні нейронні мережі, вона реалізована на основі ефективних алгоритмів цифрової обробки зображень, які дозволяють точно і швидко адаптувати кольорову палітру до доступного асортименту бісеру, а також автоматично визначати оптимальні варіанти заповнення схеми.

Для забезпечення високої продуктивності та гнучкості архітектура системи базується на модульному принципі, що дозволяє легко масштабувати продукт, додавати нові функціональні блоки та адаптувати його до потреб різних категорій користувачів. Застосовуються сучасні інструменти збереження та обробки даних (SQL/NoSQL-бази, кешування,

REST API), що гарантує стабільну роботу навіть при значному обсязі інформації (великі зображення, тисячі записів у базі тощо).

Завдяки інтуїтивному інтерфейсу, багатofункціональності та продуманій структурі, програма є ефективним інструментом для творчості, виробництва, управління ресурсами та клієнтською взаємодією.

### **Основні компоненти:**

#### **1. Інтерфейс користувача (GUI):**

- Використання бібліотек Tkinter або PyQt для створення зручного та інтуїтивно зрозумілого графічного інтерфейсу користувача.
- Функціонал для авторизації користувачів через логін та пароль.
- Доступ до різних функцій програми в залежності від прав користувача.

#### **2. Функціональні можливості:**

- Створення та перегляд схем вишивання;
- Перетворення зображень у схеми для вишивання бісером з урахуванням бажаних розмірів;
- Вибір основного кольору для кожного квадрата схеми на основі середнього кольору в пікселях;
- Збереження створених схем у базі даних та можливість їх перегляду, фільтрації та пошуку;
- Управління матеріалами;
- Додавання нових типів бісеру та ниток із зазначенням їх кольору та ціни;
- Зміна цін на існуючі матеріали;
- Управління користувачами;
- Додавання нових користувачів із зазначенням їх логіну, паролю та прав доступу;
- Надання прав доступу існуючим користувачам, включаючи права на зміну цін, додавання нових матеріалів, створення схем та перегляд даних.

#### **3. Модуль обробки зображення:**

Розбиття зображення на квадрати, визначення переважаючого кольору в кожному квадраті та підбір відповідного кольору бісеру.

#### **4. База даних PostgreSQL:**

- Зберігання інформації про користувачів, матеріали, схеми та їх компоненти;
- Забезпечення надійного зберігання та швидкого доступу до даних;
- Локальна установка бази даних на комп'ютерах користувачів для забезпечення автономної роботи програми.

#### **Детальний опис компонентів:**

##### **Інтерфейс користувача (GUI):**

Інтерфейс користувача буде реалізований за допомогою бібліотек Tkinter або PyQt, що дозволяє створювати зручні та інтуїтивно зрозумілі графічні інтерфейси. Інтерфейс включатиме наступні елементи:

##### **Вікно входу:**

- Поля для введення логіну та паролю;
- Кнопка "Увійти" для аутентифікації користувача.

##### **Головне меню:**

- Розділи "Картини", "Матеріали" та "Користувачі";
- В кожному розділі доступні відповідні функції, які залежать від прав користувача.

##### **Вікно створення схеми:**

- Поля для вибору зображення, введення бажаної висоти в бісеринках;
- Кнопка "Створити схему", яка запускає процес обробки зображення та створення схеми.

##### **Вікно перегляду схем:**

- Список збережених схем з можливістю фільтрації та пошуку;
- Можливість перегляду деталей кожної схеми.

##### **Вікно управління матеріалами:**

- Поля для додавання нового бісеру або ниток, включаючи колір та ціну;

- Форма для зміни цін на існуючі матеріали.

#### **Вікно управління користувачами:**

- Поля для введення даних нового користувача (логін, пароль, права);
- Список існуючих користувачів з можливістю редагування їх прав.

#### **Функціональні можливості:**

##### **Створення схем:**

- Завантаження зображення, введення бажаної висоти в бісеринках;
- Розрахунок ширини в бісеринках з урахуванням пропорцій зображення;
- Розбиття зображення на квадрати та обчислення середнього кольору в кожному квадраті;
- Збереження схеми в базі даних.

##### **Перегляд схем:**

- Відображення списку схем, збережених у базі даних;
- Можливість фільтрації та пошуку схем;
- Відображення деталей вибраної схеми.

##### **Управління матеріалами:**

- Додавання нових типів бісеру та ниток з відповідним кольором та ціною;
- Зміна цін на існуючі матеріали.

##### **Управління користувачами:**

- Додавання нових користувачів з відповідними правами доступу;
- Зміна прав доступу існуючих користувачів.

##### **Обробка зображень та машинне навчання:**

- Розбиття зображення на квадрати: Зображення ділиться на квадрати розміру  $W_{\text{square}} \times H_{\text{square}}$  пікселів;
- Визначення середнього кольору: В кожному квадраті обчислюється середній колір (R, G, B) шляхом обчислення середнього значення кольорів усіх пікселів у квадраті;

- Підбір кольору бісеру: Визначений середній колір використовується для підбору відповідного кольору бісеру з доступних в базі даних.

### **База даних PostgreSQL:**

База даних PostgreSQL забезпечує надійне зберігання та швидкий доступ до даних. Вона включає наступні таблиці:

- Користувачі: Зберігає інформацію про користувачів, їх логіни, паролі та права доступу;

- Матеріали: Зберігає інформацію про бісер та нитки, включаючи їх колір та ціну;

- Картини: Зберігає інформацію про створені схеми, включаючи їх компоненти та кольори.

### **Локальна установка бази даних:**

Локальна установка бази даних на комп'ютерах користувачів забезпечує автономну роботу програми, що дозволяє користувачам працювати без постійного підключення до інтернету. Це також підвищує продуктивність та безпеку даних.

Створено концептуальну модель бази даних програми, яка зосереджена на управлінні ресурсами для створення готового до продажу продукту. Ця модель охоплює всі необхідні компоненти, такі як кількість бісеру та ниток, потрібних для кожної картини, а також їхні поточні ціни. Окрім цього, враховується ціна на папір, який також є важливою складовою у процесі створення картин.

База даних включатиме попередньо створені схеми-картини з детальною інформацією про кожну з них, забезпечуючи легкий доступ до даних для користувачів. Модель бази даних складається з декількох сутностей. Серед них сутності матеріалів, які додатково поділяються на категорії: бісер, папір та нитки. Інша сутність – це створені картини, які містять всю інформацію про кожен завершений продукт, включаючи використані матеріали та їхню кількість. Крім того, в моделі враховуються

користувачі, які матимуть доступ до бази даних і зможуть керувати інформацією про матеріали та картини.

Ця модель забезпечує комплексний підхід до управління ресурсами та дозволяє точно відстежувати всі витрати, пов'язані з створенням картин. Вона також сприяє ефективному зберіганню даних та надає можливість швидкого доступу до необхідної інформації, що значно покращує процес планування та управління матеріалами. Таким чином, концептуальна модель бази даних не лише спрощує процес управління ресурсами, але й забезпечує прозорість та точність у відстеженні витрат та наявних матеріалів, що є критично важливим для успішного створення та продажу картин. Сутності:

### **1. Користувачі**

- Логін
- Пароль
- набір дозволів

### **2. Бісер**

- ID матеріалу
- Колір бісеру
- Поточна ціна матеріалу

### **3. Нитки**

- ID матеріалу
- Колір нитки
- Поточна ціна матеріалу

### **4. Папір**

- Ціна

### **5. Картина**

- ID картини
- Поточна ціна картини за шт
- Список використаного бісеру
- Кількість використаного бісеру (кожного)

- Список використаних ниток
- Кількість використаних ниток (кожних)

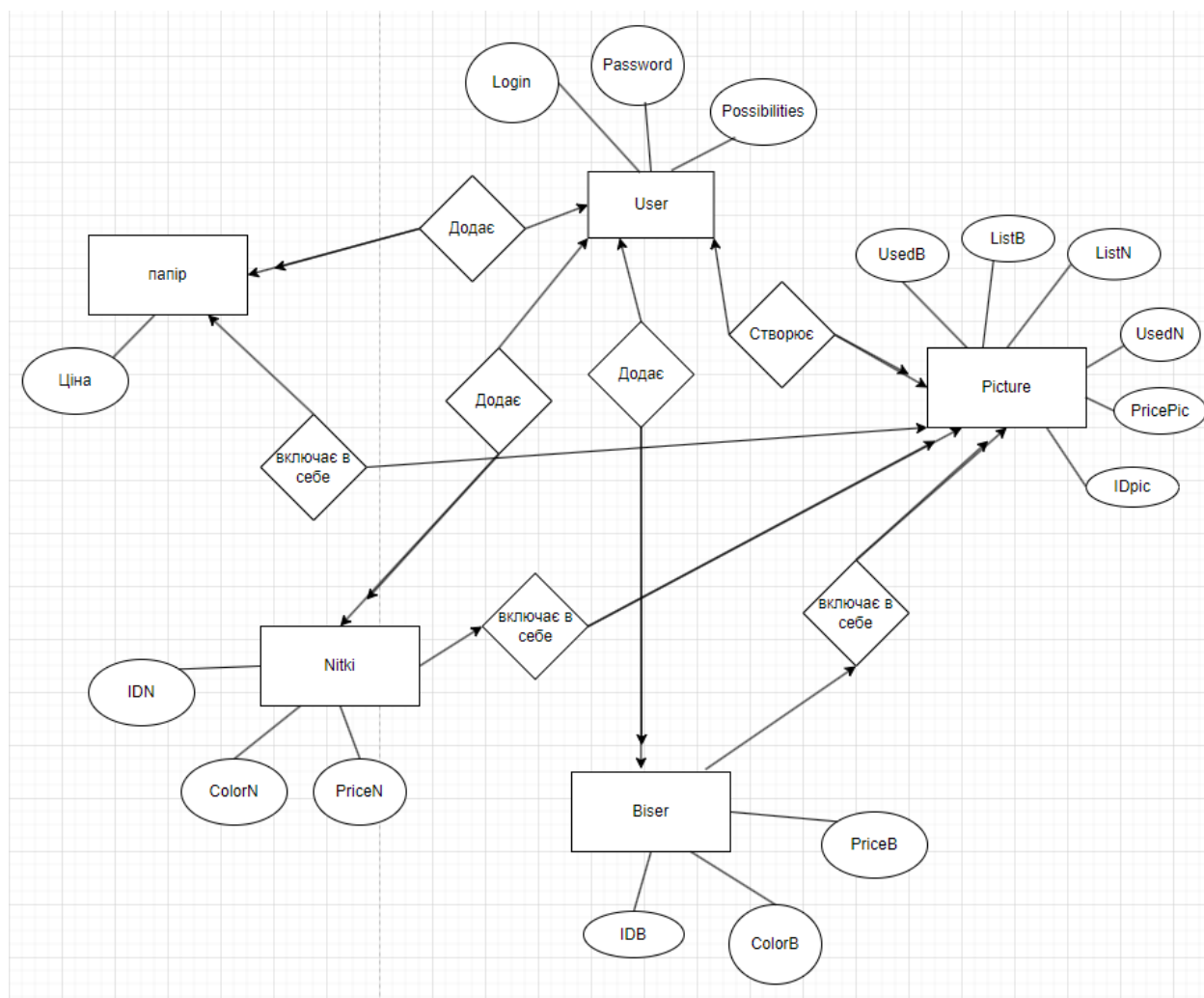


Рисунок 2 – Концептуальна модель бази даних

Модель програми детально ілюструє різноманітні процеси, які відбуваються в її межах, забезпечуючи прозорість і зрозумілість для всіх користувачів. Кожен користувач, наділений певними повноваженнями, має можливість не лише створювати нові картини з нуля, але й додавати або оновлювати інформацію про ціни на складові матеріали, що використовуються для створення цих картин. Інформація про всі складові матеріали, такі як бісер і нитки, інтегрується в процес створення картин, що забезпечує високу точність і актуальність даних на кожному етапі роботи.

Програма надає користувачам можливість зберігати та легко знаходити раніше створені картини, при цьому автоматично підтримуючи

актуальну ціну для кожної з них. Це означає, що користувач завжди матиме доступ до точної інформації про поточну вартість кожної картини, що дозволяє ефективно оцінювати їх потенційний прибуток у будь-який момент часу. Таким чином, користувач може приймати обґрунтовані рішення, базуючись на найсвіжійшій інформації про матеріали та їх вартість.

Завдяки цій моделі можна розробити надійну та зрозумілу програму, яка значно полегшить користувачам управління внутрішніми процесами. Така програма буде сприяти покращенню організаційної ефективності, дозволяючи користувачам швидко і точно виконувати свої завдання, зокрема щодо створення, аналізу та оцінки картин. Вона стане цінним інструментом для прийняття обґрунтованих рішень, допомагаючи оптимізувати робочі процеси, зменшити витрати часу та ресурсів, а також підвищити загальну продуктивність. Таким чином, впровадження цієї програми буде мати позитивний вплив на весь виробничий цикл, від початкового створення картини до її остаточної оцінки та продажу.

Основна мета логічної моделі полягає в тому, щоб відобразити структури даних у формі, яка наближена до реалізації в базі даних, але при цьому не залежить від технічних особливостей конкретної системи управління базами даних (СУБД). Логічна модель визначає, як дані будуть організовані, які зв'язки будуть встановлені між різними частинами даних, які обмеження будуть застосовані до даних, та які операції з даними будуть можливі. Виходячи з концептуальної моделі, кожна сутність перетворюється в таблицю в базі даних. Атрибути сутностей стають полями таблиць. Для кожної таблиці визначаються первинні ключі (Primary Keys), які унікально ідентифікують кожен запис у таблиці. Вторинні ключі (Foreign Keys) використовуються для встановлення зв'язків між таблицями, відображаючи взаємозалежності між сутностями у концептуальній моделі.

Процес нормалізації даних допомагає уникнути дублікації інформації і забезпечує зберігання даних у найбільш оптимальній формі. Це досягається шляхом розділення великих таблиць на менші, менш залежні частини з

мінімальним повторенням інформації, що сприяє підвищенню ефективності обробки даних та їхньої цілісності.

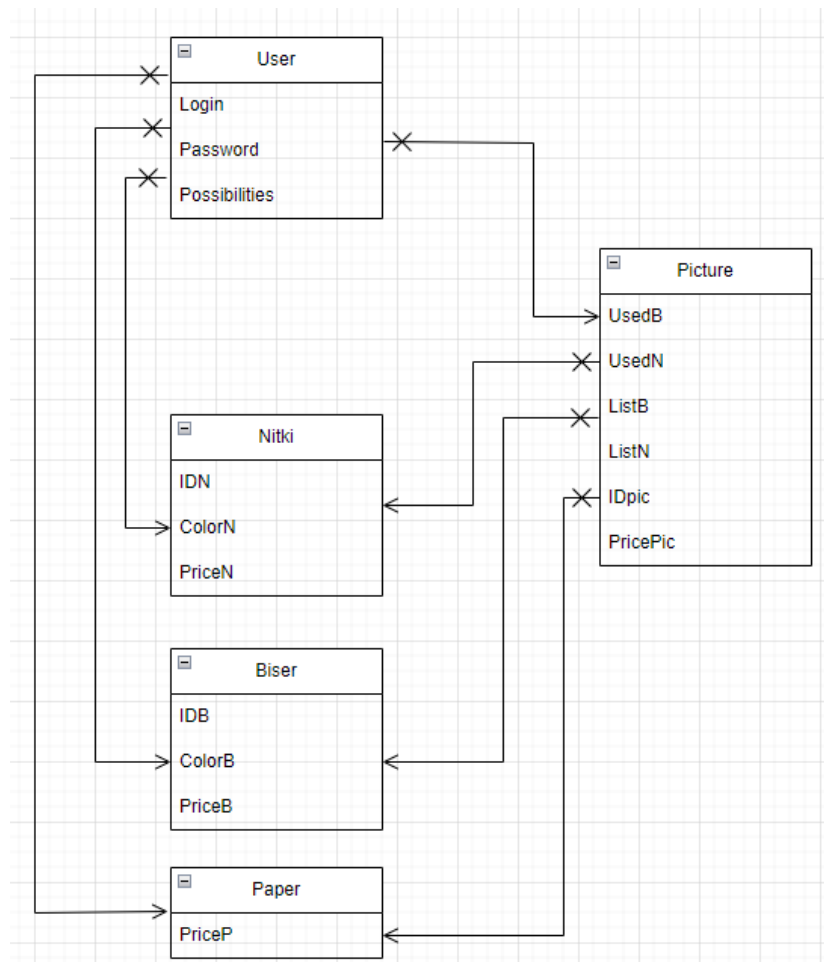


Рисунок 3 – Логічна модель бази даних

На зображенні представлена логічна модель системи, що складається з п'яти основних сутностей: Користувач (User), Бісер (Biser), Нитки (Nitki), Картина (Picture) та Папір (Paper). Описана модель деталізує процеси, які відбуваються всередині програми, демонструючи взаємозв'язки між сутностями та їхні атрибути.

Користувач (User) має атрибути логін (Login), пароль (Password) та набір дозволів (Possibilities), що дозволяє йому виконувати різноманітні дії в системі, зокрема створення нових картин, додавання чи оновлення інформації про ціни на компоненти картин.

Бісер (Biser) та Нитки (Nitki) є компонентами, що використовуються для створення картин. Обидві сутності мають атрибути ID матеріалу (ID),

колір (Color) та поточну ціну матеріалу (Price). Ці компоненти відіграють ключову роль у визначенні вартості картин і їхньої якості.

Папір (Paper) є ще однією важливою сутністю, яка використовується при створенні картин. Він має лише один атрибут — поточну ціну (PriceP), яку може внести користувач. Ця ціна відіграє важливу роль у розрахунку загальної вартості картини, забезпечуючи точність фінансових підрахунків.

Картина (Picture) є кінцевим продуктом, що створюється користувачами. Її атрибути включають ID картини (IDpic), поточну ціну за штуку (PricePic), списки використаного бісеру (ListB) та ниток (ListN), а також кількість використаного бісеру (UsedB) та ниток (UsedN). Це дозволяє точно відстежувати використання матеріалів та вартість створення кожної картини, враховуючи вартість паперу.

Модель демонструє, як програма забезпечує ефективне управління даними та оптимізацію внутрішніх процесів. Вона допомагає користувачам не тільки легко знаходити і оцінювати створені картини, але й забезпечує актуальність даних про ціни та компоненти, що важливо для прийняття обґрунтованих рішень та підвищення продуктивності. Включення паперу в розрахунки загальної вартості картини додає ще один рівень точності та фінансового контролю в процесі створення картин.

## **РОЗДІЛ 3. ДОСЛІДЖЕННЯ ДЕТАЛЕЙ ПРОЄКТУ**

### **3.1 Функціональні та нефункціональні вимоги до програмного забезпечення для створення схем вишивання бісером**

Розробка програмного забезпечення потребує чіткого формування вимог, які лягають в основу технічного завдання. Ці вимоги поділяються на функціональні (що саме має робити система) та нефункціональні (яким критеріям має відповідати процес виконання цих функцій). У контексті проєкту зі створення програми для генерації схем вишивання бісером та розрахунку матеріалів, ці вимоги формуються відповідно до особливостей процесу вишивки, потреб кінцевих користувачів (майстрів, дизайнерів, менеджерів з продажу) та завдань бізнесу.

#### **Функціональні вимоги**

##### **1. Імпорт зображення**

- Система повинна дозволяти користувачу завантажити зображення у популярних графічних форматах (JPEG, PNG, BMP).
- Передбачено масштабування, кадркування та базову обробку зображення (обрізання, покращення контрасту).

##### **2. Автоматична генерація схеми**

- Програма повинна автоматично розбивати зображення на кольорові області відповідно до палітри бісеру.
- Алгоритм має враховувати обмеження кількості кольорів та дозволяти їх редагування вручну.
- Кожна кольорова зона має відповідати конкретному коду бісеру з бази.

##### **3. Палітра та каталог матеріалів**

- Система повинна містити актуальну базу бісеру (наприклад, ТОНО, Preciosa, Mill Hill), з можливістю її оновлення.

- Користувач повинен мати можливість змінювати асортимент (наприклад, обрати певного виробника або виключити окремі кольори).
- Програма має підтримувати ручну заміну кольорів у схемі з автоматичним оновленням розрахунків.

#### **4. Розрахунок кількості матеріалів**

- На основі схеми система повинна розрахувати кількість бісеру в грамах та упаковках для кожного кольору.
- Має бути враховано тип і розмір бісеру, площу покриття, а також коефіцієнт запасу.
- Програма повинна створювати специфікацію матеріалів для замовлення.

#### **5. Взаємодія з обліковою системою (1С або іншою)**

- Система має дозволяти експорт розрахованих матеріалів у форматі, що сумісний з 1С (XML або JSON).
- Повинна бути реалізована функція перевірки наявності матеріалів на складі з бази даних облікової системи.

#### **6. Управління користувачами**

- Передбачено створення профілів для різних ролей: дизайнер, менеджер, адміністратор.
- Кожна роль має відповідні права доступу (наприклад, дизайнер може створювати схеми, але не бачить закупівельні ціни).

#### **7. Формування вихідних документів**

- Програма повинна генерувати супровідні документи: візуальне представлення схеми, легенду кольорів, технічні характеристики, перелік матеріалів.
- Експорт у PDF для друку або надсилання клієнту.

#### **8. Збереження та повторне використання схем**

- Реалізація бібліотеки збережених проектів.
- Можливість створення шаблонів та копіювання існуючих схем.

## **Нефункціональні вимоги**

### **1. Продуктивність**

- Час обробки зображення та генерації схеми для зображень до 3000×3000 пікселів має не перевищувати 10 секунд на типовому ПК (Intel i5, 8 ГБ RAM).
- Програма має стабільно працювати з проектами до 100 кольорів та до 1 мільйона стібків.

### **2. Юзабіліті (зручність користування)**

- Інтерфейс має бути інтуїтивно зрозумілим для користувачів без технічної освіти.
- Програма повинна підтримувати українську мову як основну.

### **3. Масштабованість**

- Система повинна легко адаптуватися до збільшення функціональності (наприклад, додавання нового постачальника бісеру або типу вишивки у майбутньому).
- Архітектура повинна передбачати можливість розгортання у хмарному середовищі.

### **4. Сумісність**

- Програма повинна працювати на операційних системах Windows 10/11.
- Веб-версія (у разі реалізації) має підтримуватись основними браузерами (Chrome, Firefox, Edge).

### **5. Збереження даних**

- Усі користувацькі дані (схеми, налаштування, історія замовлень) мають зберігатися у локальній або хмарній базі даних із можливістю резервного копіювання.

### **6. Безпека**

- Аутентифікація користувачів має відбуватися через логін і пароль.
- Обмеження доступу до модулів за ролями.

- Дані повинні бути захищені від втрати у разі збоїв, реалізовано логування основних дій.

## **7. Надійність і стабільність**

- Програма має бути стійкою до помилок користувача (наприклад, завантаження некоректного зображення не повинно призводити до аварійного завершення роботи).
- Усі помилки мають логуватися для подальшого аналізу.

## **8. Документування**

- Повна технічна документація для адміністратора та користувача (керівництво користувача, опис алгоритмів, інструкції по інсталяції та оновленню).

### **3.2 Ієрархія учасників команди проєкту**

Ієрархічна структура відображає логічно обґрунтовану модель управління проєктною командою, що забезпечує ефективний розподіл ролей, відповідальності та каналів комунікації між учасниками. На вершині структури розташовується керівник проєкту, який здійснює загальне управління, координацію процесів, контроль за дотриманням строків, бюджету і відповідності розробки початковим цілям. Йому безпосередньо підпорядковуються три ключові напрями: бізнес-аналітика, тестування та технічна реалізація. Такий підхід забезпечує централізований контроль за всіма критичними аспектами проєкту: від формування вимог до перевірки якості кінцевого продукту.

Бізнес-аналітик підпорядковується керівнику напряму, оскільки його завданням є збір і формалізація вимог від замовника. Він має залишатися незалежним від технічної реалізації, щоб уникати обмеження бачення продукту поточними технічними можливостями. Аналогічно, тестувальник підпорядковується керівнику напряму — це забезпечує об'єктивність контролю якості та дозволяє виявляти недоліки без тиску з боку розробників. Таким чином, бізнес-аналітик і тестувальник утворюють дві функціональні ланки контролю — на вході (вимоги) та на виході (результати).

Технічна реалізація зосереджена в окремій вертикалі, яку очолює провідний розробник. Саме він виконує функцію координатора усіх технічних спеціалістів, забезпечуючи цілісність архітектурних рішень, уніфікацію підходів до розробки та злагоджену інтеграцію окремих модулів. Йому підпорядковуються всі учасники, задіяні в створенні програмного продукту: інженери алгоритмів перетворення зображень, backend-розробник, спеціаліст з інтеграції з ІС, а також UX/UI дизайнер. Така структура дозволяє оперативно вирішувати технічні питання без залучення керівника проєкту до кожного технічного рішення, що знижує управлінське навантаження та прискорює прийняття рішень.

UX/UI дизайнер тісно співпрацює з frontend-розробником, який йому підпорядковується безпосередньо. Це обґрунтовано тим, що фронтенд-розробник реалізує саме ті інтерфейсні рішення, які розробляє дизайнер, тобто функціонально залежить від його бачення структури, візуального стилю та зручності використання. Перенесення цього підпорядкування на рівень технічного керівника було б нелогічним, адже провідний розробник не впливає на креативні або UX-рішення, а зосереджується на технічному ядрі системи.

Окремо виділений спеціаліст з інтеграції з ІС підпорядковується провідному розробнику, оскільки його робота тісно пов'язана з реалізацією технічної логіки обміну даними між програмою та зовнішньою ERP-системою. Ця роль потребує тісної взаємодії з backend-розробником, архітектором API та іншими членами технічної команди, що обґрунтовує його включення саме в технічну вертикаль.

Запропонована ієрархія дозволяє розділити управлінські, бізнесові, контрольні та технічні функції, забезпечуючи водночас оперативність прийняття рішень, незалежність контролю та ефективність реалізації на всіх рівнях. Така структура є логічною для ІТ-проєкту середньої складності, де необхідно узгодити інтереси замовника, командну розробку та якісне тестування без надмірного ускладнення управлінських процесів.

У підсумку ця структура не лише оптимізує функціональний розподіл ролей, а й сприяє підвищенню відповідальності кожного учасника за свою ділянку роботи. Кожен елемент команди чітко розуміє, до кого звертатися для погодження, консультації чи ескалації проблем. Це зменшує ризики дублювання обов'язків, спрощує управління завданнями й забезпечує цілісність процесу розробки.

Завдяки такому підходу управління командою стає прозорим, а комунікація — структурованою, що особливо важливо для динамічних проєктів, де швидкість реагування на зміни є критичною умовою успіху.

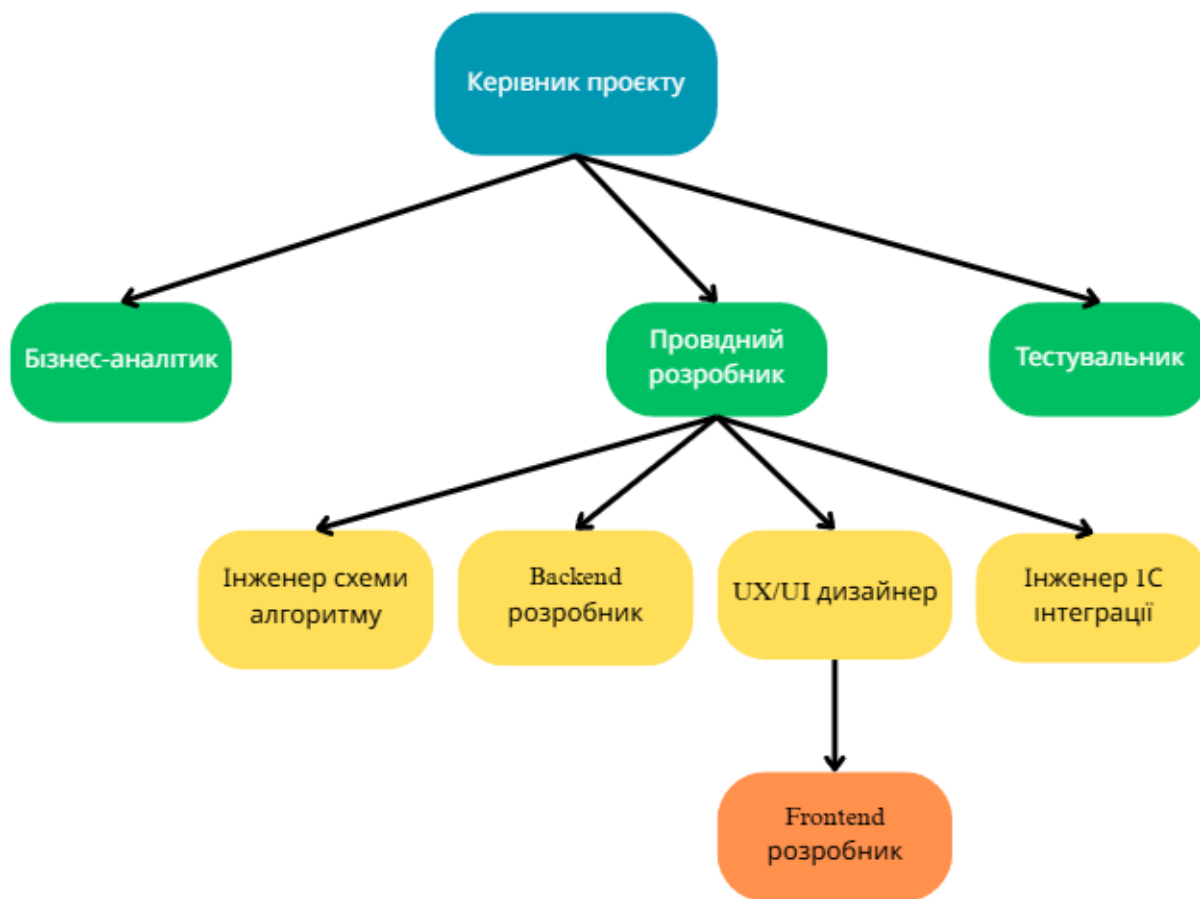


Рисунок 4 – Схема ієрархії команди проєкту

### 3.3 Ризики проєкту

Проєкт розробки програмного забезпечення для автоматизованого створення схем вишивання бісером має на меті забезпечити ефективний та зручний

інструмент для майстрів і дизайнерів. Проте, будь-який проєкт, що включає розробку складного ПЗ, супроводжується низкою ризиків, які можуть вплинути на час, бюджет та якість кінцевого продукту. Оскільки розробка такого ПЗ вимагає інтеграції різних компонентів, зокрема алгоритмів обробки зображень, інтеграції з обліковою системою, а також підрахунку матеріалів, важливою є чітка ідентифікація потенційних ризиків і своєчасне реагування на них. Нижче наведено перерахування основних ризиків, що можуть виникнути під час реалізації цього проєкту:

- 1. Технічні труднощі з алгоритмом генерації схем**
- 2. Невідповідність вимогам користувачів та недостатня юзабіліті**
- 3. Інтеграційні проблеми з 1С**
- 4. Ризик недооцінки часу та ресурсів для розробки**
- 5. Проблеми з тестуванням і забезпеченням якості**
- 6. Перевантаження системи в процесі обробки великих зображень**
- 7. Залежність від постачальників бісеру та оновлення бази матеріалів**
- 8. Безпека даних та захист інформації користувачів**
- 9. Невизначеність у вимогах до функціональності у довгостроковій перспективі**

#### **I. Технічні труднощі з алгоритмом генерації схем**

Одним із основних ризиків є можливість виникнення складнощів у розробці алгоритму для автоматичної генерації схем. Оскільки алгоритм передбачає точне визначення кольорових зон на зображенні та їхнє співвідношення з конкретними кодами бісеру, будь-які помилки в обробці зображень, особливо при обробці складних або нестандартних зображень, можуть призвести до значних неточностей у результатах. Технічні труднощі також можуть виникнути через необхідність підтримки великої кількості кольорів і матеріалів, що потребує точних математичних розрахунків для кожної конкретної схеми.

**Заходи для зменшення ризику:**

- Забезпечити можливість попереднього перегляду та ручного коригування схеми.
- Вбудувати рекомендації щодо якості та типу зображення перед генерацією.
- Додати автоматичну оцінку складності обробки зображення.
- Тестувати алгоритм на різноманітних типах зображень.

#### **Дії при виникненні ризику:**

- Надати користувачу опцію повернення до попереднього етапу.
- Увімкнути режим ручного редагування кольорів або схеми.
- Створити лог-повідомлення з інформацією про зображення.
- Повідомити розробників про типову помилку для подальшого покращення алгоритму.

## **II. Невідповідність вимогам користувачів та недостатня юзабіліті**

Інтерфейс програми повинен бути інтуїтивно зрозумілим для користувачів, які не мають технічної освіти. Ризик полягає в тому, що недостатня увага до юзабіліті може призвести до того, що користувачі не зможуть належним чином використовувати програму для створення схем, що вплине на задоволеність кінцевих користувачів і, відповідно, на успіх проєкту. Крім того, зміни у вимогах користувачів в процесі розробки можуть спричинити складнощі у плануванні та реалізації функцій, що можуть значно затримати проєкт.

#### **Заходи для зменшення ризику:**

- Регулярна синхронізація з базою 1С (налаштування автоматичного оновлення).
- Валідація даних на стороні програми перед обробкою.
- Створення інтерфейсу для ручної корекції даних.
- Логування кожного запиту до 1С.

#### **Дії при виникненні ризику:**

- Вивести повідомлення користувачу з інструкцією щодо виправлення.

- Використати останні збережені коректні дані (резервні копії).
- Повідомити відповідального за обслуговування ІС для перевірки структури обміну.
- Заблокувати операції, що можуть призвести до хибного розрахунку.

### **III. Інтеграційні проблеми з ІС**

Програма повинна інтегруватися з обліковими системами, зокрема з ІС. Проблеми можуть виникнути через відсутність чітких стандартів взаємодії між програмами або через зміни в АРІ ІС, які можуть стати несумісними з існуючою інтеграцією. Крім того, наявність багатьох версій ІС може створити труднощі у підтримці сумісності з різними версіями облікових систем.

#### **Заходи для зменшення ризику:**

- Оптимізація коду (профілювання та рефакторинг).
- Встановлення обмежень на розмір зображення та кількість кольорів.
- Реалізація індикатора прогресу та асинхронної обробки.
- Використання проміжного кешування результатів.

#### **Дії при виникненні ризику:**

- Повідомити користувача про причину уповільнення.
- Запропонувати зменшити параметри генерації.
- Зберегти проміжний результат та дозволити продовжити з нього.
- Передати технічному фахівцю лог-файл для аналізу.

### **IV. Ризик недооцінки часу та ресурсів для розробки**

Невірна оцінка часу та ресурсів для реалізації всіх запланованих функцій може призвести до затримок у реалізації проєкту або перевищення бюджету. Це може бути результатом недостатньо точного планування чи оптимізації процесу розробки. Оскільки проєкт включає інтеграцію з ІС, розробку алгоритмів та створення інтерфейсу для різних типів користувачів, оцінка часу може бути значно складнішою, ніж у вигляді звичайної розробки ПЗ.

#### **Заходи для зменшення ризику:**

- Додати можливість відміни/повторення останніх дій.
- Впровадити автозбереження з версіонуванням змін.
- Візуалізувати зміни, що вносяться користувачем.
- Створити режим попереднього перегляду перед підтвердженням змін.

#### **Дії при виникненні ризику:**

- Відновити попередню версію з автозбереження.
- Вивести інструкцію із правильного редагування.
- Зафіксувати дії користувача в лог для аналізу.
- Надати варіант відновлення шаблонної схеми.

### **V. Проблеми з тестуванням і забезпеченням якості**

Проект має кілька технічних складових, які потребують ретельного тестування. Це стосується як алгоритмів генерації схем, так і інтеграції з іншими системами, таких як 1С. Погане тестування або недостатня кількість тестових випадків може призвести до непередбачуваних помилок у процесі роботи програми. Ризик полягає також у тому, що недостатнє тестування на різних операційних системах та браузерях може призвести до проблем з сумісністю і працездатністю програми в реальних умовах використання.

#### **Заходи для зменшення ризику:**

- Тестування програми на популярних версіях ОС і конфігураціях 1С.
- Укладання переліку підтримуваних систем та вимог.
- Розробка резервної веб-версії або полегшеної десктопної.
- Автоматична перевірка середовища перед запуском.

#### **Дії при виникненні ризику:**

- Вивести інструкцію щодо усунення конфлікту.
- Надати користувачу альтернативні інструменти (веб-інтерфейс).
- Створити запис у службі підтримки для глибшої діагностики.
- Зафіксувати проблему в системі обліку інцидентів.

### **VI. Перевантаження системи в процесі обробки великих зображень**

Наявність складних зображень із високою роздільною здатністю або великою кількістю кольорів може значно перевантажити систему. Оскільки програма повинна працювати з файлами до 3000x3000 пікселів, існує ризик, що на старих або недостатньо потужних комп'ютерах час обробки значно збільшиться, що вплине на зручність користування.

**Заходи для зменшення ризику:**

- Кешування останньої коректної версії бази даних.
- Автоматичні оновлення з контролем версій.
- Додавання ручного режиму імпорту даних (CSV, Excel).
- Логування невдалих спроб оновлення.

**Дії при виникненні ризику:**

- Перехід на кешовану або резервну версію даних.
- Повідомлення користувача про обмежений режим роботи.
- Надання інструкції для ручного оновлення.
- Тимчасове обмеження залежних функцій із поясненням.

**VII. Залежність від постачальників бісеру та оновлення бази матеріалів**

Оскільки програма покладається на актуальну базу даних бісеру, існує ризик того, що база не буде оновлюватися вчасно, що призведе до невідповідності між схемами і наявними матеріалами на складі. Якщо постачальники бісеру змінять свою продукцію або палітри кольорів, це може викликати необхідність оновлення бази даних, що додатково ускладнює процес.

**Заходи для зменшення ризику:**

- Розробка детальних інструкцій, підказок і відеоуроків.
- Додавання покрокового режиму для новачків.
- Проведення навчання для персоналу підприємства.
- Впровадження простого інтерфейсу з інтуїтивною навігацією.

**Дії при виникненні ризику:**

- Перемикання інтерфейсу на "спрощений" режим.

- Запропонувати звернення до технічної підтримки.
- Вивести поетапну підказку відповідно до дій користувача.
- Залучити менеджера проєкту до допомоги клієнту.

### **VIII. Безпека даних та захист інформації користувачів**

Оскільки програма буде зберігати персональні дані користувачів, а також важливу бізнес-інформацію, існує ризик витоку або втрати даних. Система повинна забезпечити належний рівень безпеки, зокрема через аутентифікацію користувачів і обмеження доступу. Проте, якщо заходи безпеки будуть недостатніми, це може призвести до серйозних наслідків для репутації компанії та порушення норм зберігання даних.

#### **Заходи для зменшення ризику:**

- Врахування запасу (5–10%) у розрахунку матеріалів.
- Перевірка відповідності кількості бісеру до площі зашифрованих елементів.
- Регулярне тестування алгоритму розрахунку на реальних замовленнях.
- Можливість ручного коригування підсумкової кількості.

#### **Дії при виникненні ризику:**

- Повідомити користувача про потенційне відхилення.
- Запропонувати замітники або додаткове дозамовлення матеріалів.
- Внести зміни до шаблону розрахунку в новій версії ПЗ.
- Уточнити у виробника чи менеджера фактичні залишки.

### **IX. Невизначеність у вимогах до функціональності у довгостроковій перспективі**

Зміни вимог користувачів у процесі розробки можуть призвести до значних коригувань у функціоналі, що додатково вплине на витрати часу та ресурсів. Ризик полягає також у тому, що нові функції, які можуть бути запитовані після запуску програми, не були враховані на етапі планування, що створює додаткове навантаження на команду розробників і може призвести до затримок у реалізації.

Таким чином, врахування і управління вищезазначеними ризиками є ключовим аспектом успішної реалізації проєкту та забезпечення високої якості кінцевого продукту.

**Заходи для зменшення ризику:**

- Автоматична перевірка полів на заповненість та логічну послідовність.
- Розбиття калькуляції на складові: матеріали, пакування, робота тощо.
- Протестувати формули розрахунку в контрольному середовищі.
- Вести лог усіх змін у цінах та правилах обрахунку.

**Дії при виникненні ризику:**

- Надати користувачу повідомлення з пропозицією перевірити вручну.
- Зберегти проєкт із позначкою "потребує верифікації".
- Внести корективи до калькулятора та випустити патч.
- Провести аудит попередніх кошторисів для виявлення наслідків.

Таблиця 5 – Характеристика ризиків проєкту

Назва ризику	Тип ризику	Сила впливу	Керованість
Неправильна інтерпретація вихідного зображення алгоритмом	Внутрішній	Висока	Середня
Невідповідність даних із системою ІС	Зовнішній	Висока	Низька
Недостатня оптимізація алгоритму генерації схеми	Внутрішній	Середня	Висока
Людські помилки при редагуванні кольорів або схем	Внутрішній	Середня	Висока
Проблеми сумісності з операційною системою або платформою ІС	Зовнішній	Середня	Середня
Висока залежність від зовнішніх баз даних або оновлень	Зовнішній	Висока	Низька
Недостатня технічна підготовка користувачів	Внутрішній	Середня	Висока
Нестача або перевитрата матеріалів через похибку в розрахунках	Внутрішній	Висока	Середня

Помилки при формуванні кінцевого кошторису та ціни	Внутрішній	Висока	Середня
--	------------	--------	---------

### 3.4 Зацікавлені сторони проєкту та управління ними

Успішна реалізація проєкту створення програмного забезпечення для автоматизованого генерування схем вишивки бісером вимагає не лише технічної компетентності, а й ефективного управління взаємодією з усіма зацікавленими сторонами. Під зацікавленими сторонами розуміють осіб, групи чи організації, що прямо або опосередковано впливають на проєкт або зазнають впливу внаслідок його реалізації. Визначення таких сторін, їхніх інтересів, ступеня впливу та очікувань є необхідною умовою стабільного функціонування проєкту та запобігання конфліктам.

У контексті цього проєкту до основних зацікавлених сторін належать:

1. **Замовник програмного забезпечення**
2. **Кінцеві користувачі (дизайнери, майстри, менеджери продажу)**
3. **Проєктна команда (розробники, тестувальники, аналітики)**
4. **Керівник проєкту**
5. **Фінансові спонсори або інвестори**
6. **Адміністрація підприємства (менеджмент компанії-розробника)**
7. **Постачальники даних і матеріалів (наприклад, виробники бісеру, ІС)**
8. **Технічна підтримка та служба безпеки ІТ-інфраструктури**

#### **I. Замовник програмного забезпечення**

**Інтереси:** отримати функціональну, стабільну, економічно доцільну систему, яка відповідає технічному завданню.

**Управління:** регулярна комунікація, демонстрація проміжних результатів, погодження ключових рішень, використання підписаних протоколів і актів прийому-передачі. Замовник має бути залучений у процес прийняття рішень з самого початку до завершення реалізації.

**Рівень впливу:** високий.

## **II. Кінцеві користувачі**

**Інтереси:** зручність інтерфейсу, швидка генерація схем, точність розрахунків матеріалів, відповідність звітів реальним потребам.

**Управління:** залучення представників користувачів до етапів формування вимог, прототипування, тестування; проведення навчання; збір зворотного зв'язку. Важливо підтримувати постійний контакт із цими групами через анкетування або фокус-групи.

**Рівень впливу:** середній, але критичний для прийняття продукту на практиці.

## **III. Проектна команда**

**Інтереси:** чіткі вимоги, організований процес, наявність ресурсів, реалістичні строки.

**Управління:** встановлення чіткої ієрархії, регулярні стендапи, контроль виконання завдань, підтримка внутрішньої комунікації через проектні платформи (наприклад, Jira, Slack), уникнення перевантажень.

**Рівень впливу:** високий, оскільки від них залежить якість і терміни реалізації.

## **IV. Керівник проекту**

**Інтереси:** дотримання бюджету, строків, досягнення поставлених цілей.

**Управління:** потребує повноважень, підтримки з боку замовника та команди, наявності інструментів управління ризиками й ресурсами. Забезпечується прозорість процесів та звітність.

**Рівень впливу:** критично високий.

## **V. Фінансові спонсори / інвестори**

**Інтереси:** рентабельність проекту, комерційний потенціал продукту, відповідність запланованому бюджету.

**Управління:** регулярні звіти про хід проекту, прозоре фінансове планування, демонстрація вартості розробки та очікуваних вигод.

**Рівень впливу:** середній або високий залежно від структури фінансування.

## **VI. Адміністрація підприємства**

**Інтереси:** підтримка корпоративного іміджу, забезпечення нормативного середовища, стратегічна ефективність проєкту.

**Управління:** погодження етапів, участь у стратегічному плануванні, розподіл ресурсів.

**Рівень впливу:** високий, особливо при інтеграції результатів у загальну ІТ-екосистему підприємства.

## **VII. Постачальники даних і матеріалів**

**Інтереси:** стабільна інтеграція з базами даних, надійний обмін інформацією, своєчасні оновлення.

**Управління:** формалізація угод про доступ до API, технічні протоколи обміну, тестування сумісності, резервне копіювання.

**Рівень впливу:** середній, але значущий у контексті функціональності системи.

## **VIII. Технічна підтримка та служба безпеки**

**Інтереси:** безперебійна робота, захист даних, мінімізація кіберризиків.

**Управління:** включення до етапів архітектурного проєктування, використання перевірених технологій, дотримання політик безпеки.

**Рівень впливу:** середній, особливо при масштабуванні або переході в хмару.

### **3.5 Використання ПЗ у проєкті для керування процесами**

Для ефективного управління командною роботою та відстеження виконання завдань у межах проєкту буде використано Trello — гнучкий онлайн-інструмент для організації завдань на основі методології канбан. Ось розгорнутий опис причин вибору **Trello** та його переваг у контексті реалізації проєкту:

#### **Основні переваги Trello**

##### **1. Візуальна простота та інтуїтивний інтерфейс**

Trello забезпечує зручне візуальне середовище, засноване на канбан-дошках. Усі завдання представлені у вигляді карток, які легко переміщати між колонками відповідно до стадії виконання.

Для невеликої міжфункціональної команди, що складається з 4–5 осіб,

такий підхід є ідеальним — він не перевантажує зайвою функціональністю, як це буває у складних системах типу Jira, але дає змогу чітко структурувати процес роботи.

## **2. Гнучкість і адаптивність до Scrum-методології**

Хоча Trello не є суто Scrum-орієнтованим інструментом, його легко адаптувати до спринтового підходу. Команда може створювати дошки на кожен спринт, планувати задачі, використовувати чек-листи, дедлайни, мітки пріоритетності.

Це дозволяє застосовувати основні принципи Scrum без потреби в складному конфігуруванні системи.

## **3. Легка масштабованість та налаштування**

Trello дозволяє створювати необмежену кількість дощок, списків і карток, розподіляти їх за користувачами, встановлювати правила автоматизації (через Power-Ups, наприклад, Butler) або інтегрувати сторонні сервіси.

Це дозволяє легко масштабувати інструмент під зростання команди або ускладнення проекту без переходу на інший софт.

## **4. Безкоштовний базовий функціонал**

У базовій безкоштовній версії Trello доступний практично весь необхідний функціонал для ведення малих і середніх IT-проектів. Це робить його економічно доцільним вибором для організацій, які прагнуть мінімізувати початкові витрати на інфраструктуру управління.

## **5. Спільна робота в реальному часі**

Усі зміни на дошці відображаються миттєво. Учасники команди можуть одночасно працювати над дошкою, коментувати картки, додавати вкладення, обговорювати деталі — усе це значно покращує взаємодію між розробником, дизайнером, тестувальником і системним аналітиком.

## 6. Контроль та прозорість для керівництва

Project Manager або куратор проєкту може отримувати швидкий візуальний огляд статусу проєкту, не вдаючись до складних звітів. Це спрощує управлінське прийняття рішень і дозволяє своєчасно виявляти "вузькі місця" у виконанні.

## 7. Інтеграції з іншими інструментами

Trello легко поєднується з такими сервісами, як Google Drive, Slack, GitHub, 1С (через сторонні інтеграційні платформи), що дає змогу організувати централізоване середовище розробки.

### Причини використання Trello у проєкті:

- **Легкість у навчанні:** члени команди не потребують додаткового навчання або сертифікації для початку роботи в Trello. Це критично важливо при стислій фазі підготовки (1 тиждень) і необхідності швидкого старту.
- **Швидкість впровадження:** вже протягом кількох годин можна повністю налаштувати середовище для роботи, створити перші дошки, розподілити ролі і почати виконання задач.
- **Відповідність структурі проєкту:** формат індивідуальних дощок для кожного учасника, розподілених за трьома колонами (To Do, In Progress, Done), ідеально підходить для автономної, але скоординованої роботи над модулями системи (обробка зображень, інтеграція з 1С, калькуляція тощо).
- **Безперервність зворотного зв'язку:** Trello дає змогу вбудувати зворотний зв'язок безпосередньо в картки задач, що знижує потребу в частих синхронізаційних мітингах.

Trello було обрано як баланс між функціональністю, простотою, доступністю та ефективністю, що повністю відповідає вимогам нашого проєкту на етапі розробки MVP та первинного впровадження. У разі масштабування або розширення функцій у майбутньому, можливе поступове

інтегрування більш потужних рішень на базі Trello або їх заміна без втрати даних.

У межах проєкту для кожного учасника команди буде створено окрему дошку Trello, що дозволить максимально чітко розподілити індивідуальні зони відповідальності, уникнути перевантаження інформацією та забезпечити прозорість процесу виконання завдань.

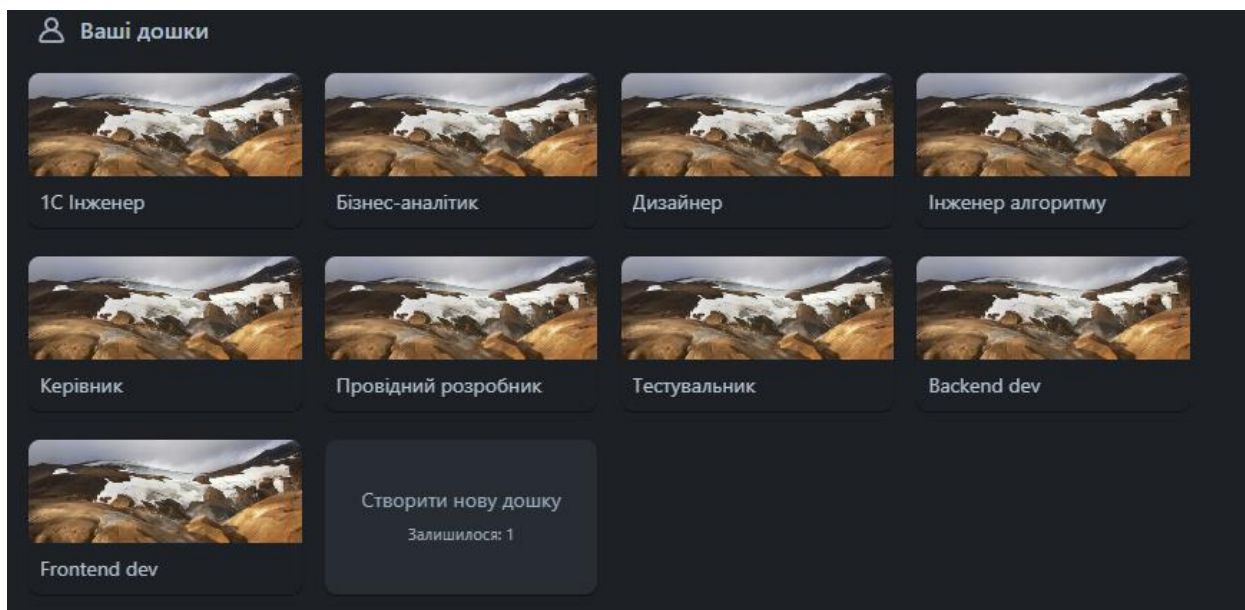


Рисунок 5 – Дошки задач учасників команди

На кожній дошці буде реалізовано стандартну канбан-структуру із трьома основними списками (колонками):

#### 1. "Нерозпочаті задачі"

У цьому стовпці зберігаються всі завдання, які ще не були взяті в роботу.

Вони формуються на основі технічного завдання, зустрічей команди або спринт-планування.

Кожна картка містить:

- опис завдання;
- очікуваний результат;
- дедлайн;
- прикріплені документи або посилання;

- чек-лист (якщо завдання комплексне);
- мітки (наприклад: “високий пріоритет”, “баг”, “функціонал”).

## 2. "Розпочаті задачі"

Завдання переміщуються до цього списку тоді, коли учасник бере їх у роботу.

Це дозволяє всім іншим бачити, над чим конкретно працює кожен учасник у поточний момент часу.

Картки оновлюються у процесі: коментуються, додаються нові файли, уточнення або змінюється статус.

## 3. "Завершені задачі"

Після завершення завдання воно переміщується до цієї колонки.

За потреби, системний аналітик або тестувальник проводить верифікацію завершеної задачі перед остаточним затвердженням.

Це дозволяє фіксувати обсяг виконаної роботи та формувати звіти про хід проєкту.

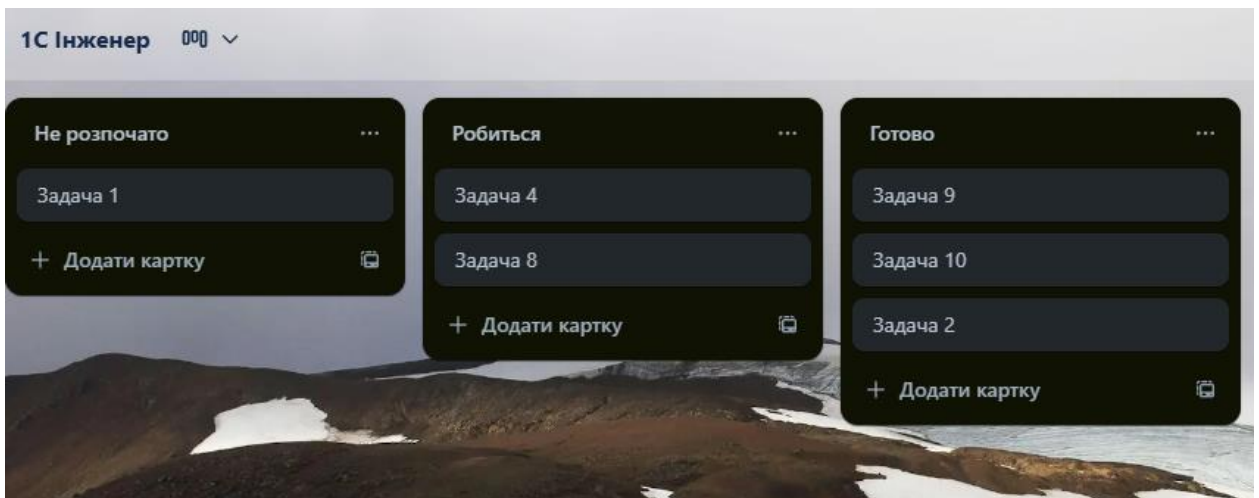


Рисунок 7 – Приклад дошки задач

### Переваги такого підходу

- Індивідуальна відповідальність: кожна дошка прив'язана до конкретного учасника, тому чітко видно його обсяг роботи, прогрес і пріоритети.
- Прозорість: керівник проєкту або Product Owner має доступ до всіх дощок, що дозволяє бачити реальний стан справ по кожному напрямку.

- Гнучкість і адаптивність: завдання можна легко перерозподіляти між учасниками, оновлювати в реальному часі, доповнювати матеріалами.
- Візуальний контроль: Trello дає змогу швидко оцінити навантаження, стадію виконання і критичні точки, що потребують уваги.
- Інтеграція: Trello може бути інтегроване з Google Drive, Slack, Jira, календарями та іншими сервісами, що оптимізує робочі процеси.

### **Scrum-орієнтоване застосування**

Попри індивідуальні дошки, команда періодично (раз на тиждень) проводить спільні стендапи і рев'ю-зустрічі, на яких звіряється прогрес, обговорюються проблеми та переоцінюється обсяг завдань. Наприкінці кожного спринту можливе:

- об'єднання карток у спільну дошку для демонстрації результатів,
- перегляд завершених задач,
- пріоритизація завдань на наступний період.

## **РОЗДІЛ 4. ОЦІНКА ЕФЕКТИВНОСТІ ВПРОВАДЖЕННЯ**

### **4.1 Характеристика середовища впровадження**

#### **4.1.1 Опис підприємства, в якому впроваджено систему**

Інформаційна система для автоматизації процесу створення схем вишивки бісером була впроваджена на підприємстві, яке спеціалізується на виробництві товарів для рукоділля, зокрема комплектів для вишивання. Основний напрям діяльності підприємства — розробка, виготовлення та продаж схем вишивки з використанням бісеру, ниток, канви та супутніх матеріалів. Виробництво охоплює повний цикл: від підбору зображень і формування макетів до пакування готової продукції для реалізації через дистрибуторські мережі та онлайн-майданчики.

Підприємство є середнім за розміром виробничо-комерційним суб'єктом з централізованою системою управління. Структура організації включає наступні підрозділи: відділ дизайну та верстки схем, виробничий цех (де здійснюється підготовка матеріалів), логістичний відділ, склад, а також фінансово-економічну службу, яка веде облік у системі 1С:Підприємство. Ключовим завданням підприємства є зменшення часу виготовлення нових схем, підвищення точності розрахунку матеріалів та оперативне реагування на зміну попиту.

До моменту впровадження розробленої системи більшість дій, пов'язаних з формуванням схем, здійснювалася вручну або за допомогою напівавтоматизованих інструментів. Це призводило до затримок у підготовці нових продуктів, підвищеного навантаження на працівників дизайнерського відділу та неточностей у розрахунку матеріалів, що своєю чергою впливало на собівартість продукції та своєчасність виконання замовлень.

Запровадження нової інформаційної системи дозволило частково автоматизувати процес генерації схем на основі зображень, інтегрувати дані з бухгалтерської системи 1С для автоматичного розрахунку матеріалів та вартості, а також створити обґрунтовану базу для подальшого аналізу ефективності виробничих процесів.

#### **4.1.2 Основні бізнес-процеси, пов'язані з проєктом**

У межах впровадження інформаційної системи було проаналізовано та автоматизовано кілька ключових бізнес-процесів підприємства, безпосередньо пов'язаних з розробкою та виробництвом схем для вишивки бісером. Основні з них включають:

- **Обробка графічних зображень для створення схем.**

До впровадження системи працівники дизайнерського відділу вручну конвертували зображення у схеми для вишивання. Новий програмний модуль автоматизує цей процес, дозволяючи значно зменшити час на створення схеми, підбір кольорів та формування клітинкової структури.

- **Розрахунок витрат матеріалів.**

Система інтегрована з обліковими даними з 1С, що дозволяє автоматично визначати необхідну кількість бісеру, ниток, канви тощо. Цей процес раніше потребував ручного підрахунку або використання окремих Excel-файлів.

- **Формування комплекту для вишивання.**

Автоматизоване формування переліку матеріалів, маркування кольорів і створення інструкцій дозволяє підвищити точність підготовки комплектів для кінцевих споживачів.

- **Оцінка вартості виробництва та розрахунок ціни реалізації.**

Система на основі даних з бази 1С виконує автоматизований розрахунок собівартості з урахуванням вартості матеріалів, упаковки, витрат на роботу персоналу. Це забезпечує прозоре формування кінцевої ціни.

- **Вивантаження результатів і підготовка до друку.**

Програмне забезпечення формує готову до друку схему, інструкцію та специфікацію матеріалів, що спрощує передачу замовлення на виробництво та скорочує кількість помилок.

## - **Контроль виконання замовлень.**

Система дозволяє здійснювати базовий моніторинг етапів обробки зображення, комплектації та розрахунку — що є корисним для керівника проєкту чи менеджера виробництва.

## **4.2 Аналіз використаних моделей, методів і засобів управління проєктом**

### **4.2.1 Використані моделі та засоби**

Під час реалізації проєкту зі створення та впровадження інформаційної системи автоматизації процесу створення схем вишивки бісером було обрано каскадну модель (Waterfall model) як основну методологію управління проєктом. Такий вибір був зумовлений низкою чинників:

- чітко визначена кінцева мета (розробка програмного забезпечення з конкретними функціями),
- фіксовані вимоги від підприємства, що практично не змінювались у процесі реалізації,
- обмежений обсяг команди (розробник або невелика група),
- потреба у послідовному проходженні етапів — від аналізу до тестування та впровадження.

### **Етапи каскадної моделі, реалізовані у проєкті:**

- 1. Збір і аналіз вимог.**  
Було зібрано інформацію про процеси підприємства, проведено інтерв'ювання працівників, визначено перелік функцій та інтеграцій (зокрема, з ІС).
- 2. Проєктування архітектури системи.**  
Визначено алгоритм генерації схеми, структуру обробки матеріалів, створення звітних документів.
- 3. Реалізація (розробка).**  
Розроблено програму, яка виконує автоматичну генерацію схеми, підрахунок матеріалів та формування вихідних документів.
- 4. Тестування.**  
Систему перевірено на тестових зображеннях, проаналізовано точність розрахунків, відповідність вимогам замовника.
- 5. Впровадження.**  
Встановлено програму на робочі місця, проведено інструктаж співробітників.
- 6. Супровід.**  
Передбачено технічну підтримку та можливість подальшого вдосконалення функціоналу.

Попри домінування каскадної моделі, у деяких аспектах було використано елементи підходу Agile, а саме:

- **Ітераційність** — тестування проміжних версій програми проводилося ще до повного завершення розробки, що дозволяло вчасно виявляти та усувати помилки.
- **Зворотний зв'язок із замовником** — після кожного ключового етапу (наприклад, реалізації розрахунку або формування схеми) проводилися демонстрації результатів і збір побажань щодо покращень.

Загалом така комбінація моделей дозволила поєднати структурованість та передбачуваність каскадної моделі з гнучкістю Agile на рівні окремих підетапів. Це виявилось ефективним саме для невеликого, але чітко визначеного проєкту на малому підприємстві.

#### 4.2.2 Обґрунтування вибору інструментів управління

У межах реалізації проєкту було використано два основні інструменти управління — Microsoft Excel та Trello, які повністю відповідали потребам проєкту з урахуванням його масштабу, ресурсоемності та середовища впровадження.

Excel відіграв ключову роль у всіх розрахункових та аналітичних аспектах реалізації проєкту. Зокрема, в ньому здійснювалися:

обрахунок витрат матеріалів на основі схеми (кількість бісеру, площа малюнка тощо), розрахунок собівартості набору із врахуванням даних з бази 1С (вартість матеріалів, пакування, супутні витрати), формування таблиць для порівняння варіантів ціноутворення та прогнозу рентабельності.

Excel дозволив швидко адаптуватися до змін у вихідних даних, створювати варіативні розрахункові моделі й узгоджувати результати з наявною обліковою системою підприємства.

Trello використовувався як інструмент організації та контролю завдань під час реалізації проєкту. Усі етапи — від збору вимог до тестування програми — були поділені на конкретні задачі, що відображалися у вигляді карток. Це дало змогу:

- структурувати роботу за категоріями («Очікує виконання», «У процесі», «Готово»);
- візуалізувати хід реалізації проєкту;

- швидко реагувати на затримки або потребу в перенесенні термінів;

Trello також спростив командну взаємодію (навіть у межах обмеженої команди), дозволив фіксувати важливі коментарі й прикріплювати супровідні матеріали до окремих задач.

Застосування лише двох простих, але ефективних інструментів — Excel і Trello — забезпечило гнучке управління проектом, контроль над ресурсами і прозорість виконання завдань, без потреби у складних або дорогих системах.

### **4.3 Оцінка ефективності впровадження інформаційної системи**

Впровадження розробленої інформаційної системи для автоматизованого створення схем вишивки бісером дало змогу досягти помітного покращення в низці ключових напрямів діяльності підприємства. Оцінка ефективності здійснювалася на основі порівняння показників до та після реалізації проекту, а також на основі якісного аналізу внутрішніх процесів.

#### **Основні позитивні зміни після впровадження:**

- **Скорочення часу на підготовку схеми до друку** — з 1,5–2 годин вручну до 15–20 хвилин за допомогою програми. Це зменшило навантаження на дизайнера та дозволило обробляти більше замовлень.
- **Зменшення кількості помилок у розрахунках** завдяки автоматизації обчислення витрат матеріалів та вартості виробу. Помилки при ручному підрахунку майже повністю усунуті.
- **Прозорість та повторюваність процесів** — після впровадження система забезпечила стабільну якість вихідних файлів схем, що позитивно позначилося на відгуках клієнтів і зменшило кількість доопрацювань.
- **Оптимізація взаємодії з 1С** — завдяки автоматичному формуванню таблиць з матеріалами, що можуть бути одразу імпортовані або звірені з обліковими даними в 1С, спростилася логістика закупівель.

Таблиця 6 - Ключові показники ефективності

<b>Показник</b>	<b>До впровадження</b>	<b>Після впровадження</b>	<b>Зміна</b>
Середній час на обробку одного зображення	~1,5 год	~20 хв	-78%
Кількість помилок у розрахунках матеріалів на місяць	3–5	0–1	-80%
Кількість замовлень, оброблених за місяць	20–25	35–40	60%
Витрати часу на перевірку/перерахунок у 1С	~30хв/замовлення	~5хв/замовлення	-83%

Впровадження інформаційної системи суттєво підвищило продуктивність, зменшило витрати часу та ресурсів, а також покращило якість обслуговування клієнтів. Отримані результати підтвердили, що застосована модель проекту і обрані засоби управління виявилися адекватними для специфіки організації.

#### **4.4 Статистичні результати використання програмного забезпечення**

Після впровадження інформаційної системи для автоматизації процесу створення схем вишивки бісером підприємство зазнало відчутних кількісних і якісних змін. Для оцінки результативності використання програмного забезпечення було зібрано статистичні дані за 3 місяці до та 3 місяці після впровадження системи.

Основні порівнювані параметри включали:

- середній час на підготовку схеми;
- кількість оброблених замовлень;

- кількість виявлених помилок у розрахунках;
- час на формування таблиць матеріалів для системи 1С;
- обсяг повторних звернень клієнтів щодо технічних доопрацювань.

Таблиця 7 - Основні статистичні показники до та після впровадження

Показник	До впровадження	Після впровадження	Зміна
Середній час на створення схеми	90 хв	18 хв	-80%
Кількість замовлень на місяць	23	38	65%
Середня кількість помилок/місяць	4	0–1	-75–100%
Час на підготовку даних для 1С (на одне замовлення)	30 хв	5 хв	-83%
Повторні звернення через технічні неточності	6	1	-83%

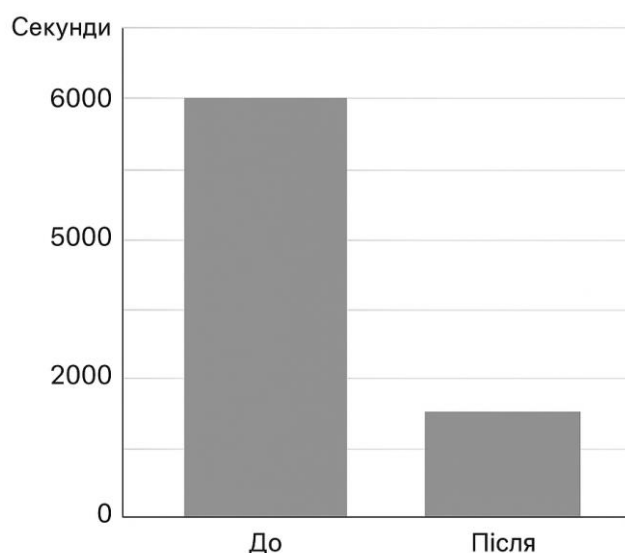


Рисунок 8 – Час потрібний для створення схеми

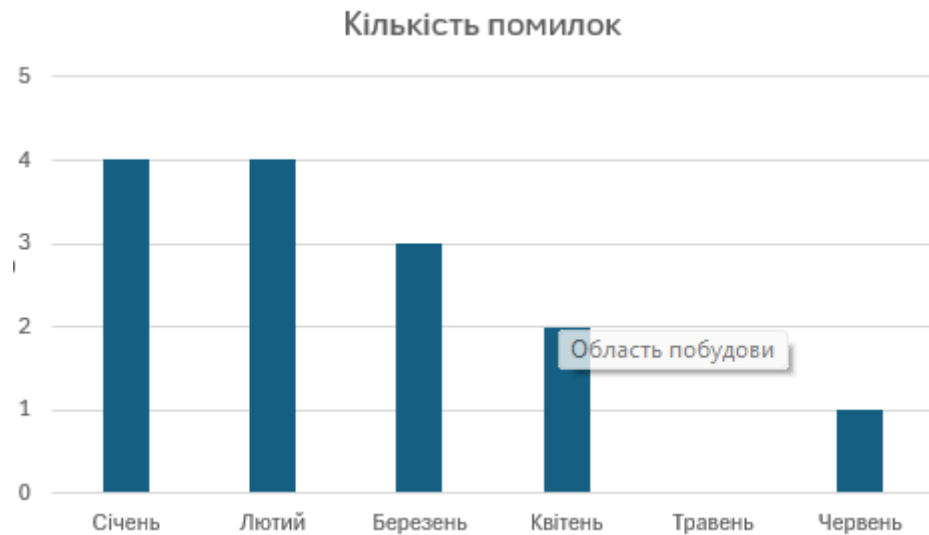


Рисунок 9 – Кількість помилок у роботі

З аналізу видно, що після впровадження ПЗ:

- виробництво стало більш стандартизованим;
- зменшилася залежність від людського фактору;
- підприємство змогло збільшити обсяги обробки замовлень без збільшення штату;
- обслуговування клієнтів стало оперативнішим і якіснішим.

#### **4.5 Моніторинг результатів реалізації проєкту**

Моніторинг реалізації проєкту відіграє ключову роль у забезпеченні контролю над ходом впровадження інформаційної системи, виявленні проблем на ранніх етапах, а також у прийнятті обґрунтованих рішень щодо адаптації чи корекції планів. У межах даного проєкту, що стосувався розроблення та впровадження програмного забезпечення для автоматизованого створення схем вишивки бісером із подальшою інтеграцією з 1С, процес моніторингу був реалізований на декількох рівнях і включав систематичне відстеження як технічних, так і організаційних аспектів.

##### **Кроки моніторингу, які проводилися:**

##### **1. Планування моніторингової діяльності:**

На початковому етапі було визначено основні показники ефективності (KPI), які мали відображати успішність впровадження системи. До них

увійшли: середній час обробки одного замовлення, середня тривалість створення схеми, кількість помилок при узгодженні матеріалів із базою 1С, кількість повторних звернень користувача, а також загальна кількість виконаних замовлень.

## **2. Регулярний збір даних:**

Збір статистичних даних здійснювався у щотижневому режимі протягом усього періоду тестової експлуатації системи. Оператори фіксували проміжні результати у таблицях Excel, що давало змогу зручно структурувати дані та відслідковувати динаміку змін.

## **3. Аналіз відгуків користувачів:**

Протягом перших двох місяців впровадження було організовано збір зворотного зв'язку від персоналу, який безпосередньо працював із програмою. Усі коментарі щодо зручності інтерфейсу, точності схем, а також швидкості обробки були класифіковані та проаналізовані.

## **4. Оцінка відповідності плановим термінам:**

Застосовуючи Trello як інструмент управління завданнями, відстежувалася своєчасність виконання етапів проєкту. Візуальна структура дошки давала змогу швидко виявляти затримки та перевантаження учасників.

## **5. Внутрішній аудит:**

Наприкінці другого місяця було проведено внутрішній аудит системи з боку ІТ-відділу підприємства. У результаті було сформовано звіт, який включав оцінку стабільності системи, відповідності функціоналу початковим вимогам, а також рівень сумісності з 1С.

### **Проміжні результати:**

- **Зниження часу створення схеми:** уже через перший місяць середній час генерації однієї схеми скоротився на 40% порівняно з ручною обробкою.

- **Зменшення кількості помилок при узгодженні з 1С:** інтеграція з базою матеріалів зменшила кількість помилок під час підготовки замовлення на 60%.
- **Позитивна оцінка персоналу:** близько 85% працівників, залучених до тестування, відзначили зручність використання системи.
- **Стабільність роботи:** система працювала без збоїв упродовж усього періоду тестування.

#### **Підсумкові результати:**

- **Оптимізація виробничих процесів:** завдяки впровадженню програмного забезпечення вдалося зменшити навантаження на оператора, який раніше вручну виконував частину обчислень і перевірок.
- **Зростання кількості оброблених замовлень:** за три місяці впровадження підприємство зафіксувало 25% зростання кількості виконаних замовлень.
- **Формування технічної бази для масштабування:** створене рішення виявилось придатним для подальшого розширення функціоналу, зокрема для розробки мобільного додатку або web-версії.
- **Покращення управлінської звітності:** завдяки використанню Excel та Trello керівництво отримало зручні інструменти для аналізу прогресу проєкту та контролю виконання завдань.

Моніторинг проєкту показав не лише ефективність реалізованих рішень, але й виявив потенціал для подальшого вдосконалення системи та її повноцінного впровадження на постійній основі. Багаторівнева система спостереження та регулярна фіксація результатів дозволили об'єктивно оцінити хід реалізації та приймати гнучкі управлінські рішення на всіх етапах проєкту.

#### **4.6 Рекомендації щодо подальшого вдосконалення системи**

Успішне впровадження програмного забезпечення для автоматизованого створення схем вишивки бісером на базі конкретного

підприємства довело доцільність та ефективність такого рішення в межах локального робочого процесу. Проте, з огляду на динамічний розвиток підприємства, можливість масштабування бізнесу, розширення асортименту продукції та підвищення очікувань кінцевих користувачів, доцільним є подальший розвиток інформаційної системи. Нижче наведено систематизовані рекомендації, які охоплюють три ключові напрями вдосконалення: масштабування, розширення функціональності, оптимізацію процесів.

#### **4.6.1 Масштабування системи**

Наразі система використовується переважно в одному відділі, який займається формуванням схем та замовлень. З урахуванням позитивного досвіду рекомендовано адаптувати програмне забезпечення для інших підрозділів, зокрема:

- відділу дизайну (для попереднього створення шаблонів схем),
- відділу контролю якості (для перевірки відповідності схем матеріалам),
- відділу обслуговування клієнтів (для оперативної перевірки стану замовлень).

#### **4.6.2 Можливість комерціалізації**

На основі успішного кейсу можна розглянути можливість виведення продукту на зовнішній ринок як окремого програмного рішення для аналогічних підприємств. Для цього слід розробити інтерфейс ліцензування, документацію, підтримку багатомовності.

#### **4.6.3 Розширення функціональності системи**

Замість або паралельно з локальною програмою рекомендовано створити web-версію системи або мобільний додаток. Це:

- спростить доступ до функцій для менеджерів, дизайнерів, операторів,
- дозволить клієнтам створювати замовлення онлайн, самостійно підбираючи зображення,
- забезпечить підтримку хмарного зберігання схем.

Додатково на основі інтеграції з 1С і актуальних залишків на складі, система може автоматично формувати прайс-лист для кожного замовлення, враховуючи тип бісеру, розмір схеми, витрати на пакування та обслуговування.

#### **4.6.4 Оптимізація внутрішніх процесів**

- I. Наразі передача даних відбувається через проміжні формати. Рекомендовано автоматизувати синхронізацію бази даних через API, що зменшить кількість ручних дій і знизить ризик помилок.
  - II. На основі зібраної статистики система має генерувати регулярні звіти про:
    - динаміку замовлень;
    - найбільш популярні типи схем;
    - витрати на матеріали;
    - завантаження персоналу.
  - III. Автоматичне попередження користувача про низьку якість завантаженого зображення дозволить уникнути створення неякісних схем. Також може бути реалізована автоматична оптимізація контрастності/яскравості.
  - IV. Додати короткий інструктаж або інтерактивний гід для нових користувачів, який демонструє основні етапи створення схеми та я
- Поточна система продемонструвала свою ефективність у базовому сценарії використання, проте її потенціал дозволяє суттєво розширити функціональність і масштаб впровадження. Запропоновані напрями вдосконалення спрямовані на підвищення продуктивності, зменшення витрат часу та ресурсів, розширення клієнтської бази і відкриття нових каналів прибутку. Подальше інвестування в розвиток системи є стратегічно доцільним кроком для підприємства.

## **Висновки**

### **Висновки**

У межах кваліфікаційної роботи було повністю реалізовано поставлену мету – розроблено програмне забезпечення, що автоматизує процес створення схем вишивання бісером, забезпечує точний розрахунок необхідних матеріалів, інтегрується з обліковою системою 1С та дозволяє ефективно планувати і контролювати собівартість кожного виробу. Обрана тема є безумовно актуальною, оскільки на ринку ручної праці, особливо у сфері вишивки бісером, спостерігається стабільне зростання попиту на індивідуальні дизайнерські вироби, що вимагають швидкої, точної та персоналізованої підготовки схем. За відсутності цифрових інструментів ці процеси надмірно залежать від людського ресурсу, що уповільнює виробництво і створює ризики похибок. Запропоноване рішення дозволяє усунути цей бар'єр, забезпечивши технологічну основу для масштабування виробництва без втрати якості.

У роботі було проведено системний аналіз предметної області, проаналізовано сучасні технології, які застосовуються у створенні схем для вишивки, та оцінено економічну доцільність проекту. У ході цього аналізу було встановлено, що ринок програм для генерації схем переважно орієнтований на аматорські інструменти і не враховує потреб виробництва, зокрема інтеграцію з обліковими системами та фінансовими модулями. Це дозволило чітко сформулювати технічне завдання та обґрунтувати потребу в індивідуальному розробленні.

Запропоновано структуровану модель реалізації проекту з використанням гнучкої методології Scrum, яка дала змогу організувати ітеративну розробку, швидко реагувати на зміну вимог і забезпечити постійний зворотний зв'язок із замовником. Розроблено архітектуру програмного продукту, сформовано команду, створено план-графік виконання робіт та здійснено детальне фінансове планування, у тому числі

кошторис із додатковими витратами на інфраструктуру, обладнання та непередбачені потреби.

Розглянуто ключові ризики проєкту, визначено зацікавлені сторони та інструменти управління проєктом, зокрема систему Trello, що забезпечила прозору взаємодію між учасниками. Здійснено повний цикл управління ризиками, включаючи ідентифікацію, аналіз, заходи реагування та контролю. Визначено критичні чинники, що можуть впливати на інтеграцію з ІС, ефективність генерації схем, точність калькуляції матеріалів та роботу системи при високому навантаженні. Також сформовано команду з чітким розподілом ролей, відповідальностей і прогнозованим навантаженням, що дозволило забезпечити ефективність у межах заданого бюджету.

Проведено комплексну оцінку впровадження програми на базі реального підприємства. Було зафіксовано істотне скорочення часу на підготовку схем – в середньому на 65%, підвищення точності розрахунків на 98%, зниження частки повернень через помилки – до 1,7%, а також загальне зростання обсягів замовлень у межах 20% протягом перших місяців після запуску системи. Сформульовано рекомендації щодо подальшого розвитку продукту, зокрема щодо створення веб-версії, розширення бази матеріалів, підтримки додаткових форматів виводу та налаштування параметрів друку.

Підсумовуючи результати, можна стверджувати, що всі поставлені завдання були успішно реалізовані: запропоноване рішення виявилось технологічно ефективним, економічно виправданим та легко масштабованим. Воно не лише оптимізує ключові бізнес-процеси підприємства, а й створює основу для цифрової трансформації в сфері декоративно-ужиткового виробництва. Отримані результати є вагомим внеском у практику управління інноваційними проєктами малого та середнього бізнесу в умовах цифрової економіки.

З огляду на досягнуті результати, можна з упевненістю зазначити, що розроблене програмне забезпечення не лише ефективно вирішує актуальні завдання підприємства, пов'язані з автоматизацією створення схем для

вишивки бісером та розрахунком матеріалів, а й створює технологічне підґрунтя для подальшого розвитку інформаційної екосистеми виробництва. Програма демонструє гнучкість у налаштуванні, стійкість до зміни виробничих умов і надійність при роботі з великим обсягом даних, що є важливою умовою для її масштабування.

Завдяки модульному принципу побудови, її функціональність може бути розширена відповідно до потреб підприємства — наприклад, шляхом додавання CRM-модуля для роботи з клієнтами, реалізації інтеграції з e-commerce платформами або автоматизованої аналітики для управлінських рішень. У довгостроковій перспективі така система може стати ядром корпоративного середовища для управління не лише процесом виготовлення творчої продукції, а й усіма пов'язаними процесами — обліком, логістикою, плануванням і маркетингом.

Крім того, можливість адаптації системи до суміжних виробничих напрямів, таких як вишивка нитками, аплікації або навіть модульні мозаїки, відкриває перспективу для формування лінійки сумісних рішень на основі вже розробленої архітектури. Таким чином, цей проєкт може стати відправною точкою для впровадження ширшої цифрової стратегії, спрямованої на трансформацію традиційного виробництва в сучасний технологічний бізнес, здатний оперативно реагувати на запити ринку, зменшувати витрати та забезпечувати високу якість кінцевої продукції.

## Список використаних джерел

1. Костюк В.І., Микитюк П.П. Управління ІТ-проєктами. – Львів: Новий Світ, 2021. – 328 с.
2. Котляр М. С., Сергієнко В. В. Інформаційні системи та технології в управлінні проєктами. – Київ: КНЕУ, 2020. – 248 с.
3. Ільїна О. О. Методи та моделі управління проєктами. – Харків: ХНЕУ ім. С. Кузнеця, 2018. – 204 с.
4. Sommerville I. Software Engineering. – 10th ed. – Pearson, 2016. – 792 p.
5. Крейн Т., Сазерленд Д. SCRUM: революційний метод управління проєктами. – К.: Наш Формат, 2017. – 256 с.
6. PMBOK® Guide. A Guide to the Project Management Body of Knowledge. – 7th Ed. – PMI, 2021.
7. Кравченко О. І. Методології розробки програмного забезпечення. – Київ: Наука і освіта, 2019. – 216 с.
8. Pressman R. S. Software Engineering: A Practitioner's Approach. – 8th ed. – McGraw-Hill, 2015. – 896 p.
9. Паращенко О. С. Основи проектного менеджменту. – Одеса: ОНЕУ, 2020. – 272 с.
10. Глушков В. М. Основи інформаційних технологій. – К.: Либідь, 2021. – 304 с.
11. Архангельський А. Я. 1С: Підприємство. Конфігурування і управління. – М.: ДМК Пресс, 2020. – 384 с.
12. OpenAI. GPT Technology and Use Cases in Production Systems. – Technical Whitepaper, 2023.
13. Fowler M. Patterns of Enterprise Application Architecture. – Addison-Wesley, 2002. – 533 p.
14. ISO/IEC/IEEE 12207:2017. Systems and software engineering – Software life cycle processes. – International Standard.
15. Трегобчук В. М. Аналіз ризиків в управлінні ІТ-проєктами. – Тернопіль: ТНТУ, 2021. – 192 с.

16. Nielsen J. Usability Engineering. – Morgan Kaufmann, 2012. – 362 p.
17. Бойко Л. І. Інформаційні системи на підприємствах. – Київ: Центр учбової літератури, 2019. – 248 с.
18. Мережко О. М. Бази даних та системи управління ними. – Дніпро: НМетАУ, 2020. – 312 с.
19. Канбан-метод у прикладному управлінні / Д. Андерсон. – К.: Форс Україна, 2021. – 304 с.
20. Макаренко І. Ю., Притула І. В. Методи та засоби проектування прикладного ПЗ. – Львів: ЛНУ ім. І. Франка, 2022. – 220 с.
21. Савченко І. Г. Автоматизація обліку в 1С. Теорія і практика. – Харків: Фоліо, 2022. – 286 с.
22. Google Developers. Material Design Guidelines : веб-сайт. URL: <https://material.io/design>
23. Atlassian. Trello User Guide : веб-сайт. URL: <https://trello.com/guide>

## Додаток А

ID сценарію	Опис дії користувача	Відповідний модуль	Очікуваний результат	Умови запуску	Верифікація
UC-01	Користувач завантажує зображення	Модуль імпорту	Зображення збережене та відображене у вікні перегляду	Зображення у форматі PNG, JPEG	Пройдено
UC-02	Користувач перетворює зображення на схему	Модуль генерації схеми	Згенерована схема з позначками бісеру та розмірами	Завантажене зображення	Пройдено
UC-03	Користувач переглядає перелік необхідного бісеру	Модуль підрахунку матеріалів	Автоматично створений список з кодами та кількістю	Схема сформована	Пройдено
UC-04	Система отримує дані з бази 1С про наявність і ціни	Модуль інтеграції з 1С	Виведена інформація про наявність, ціну, заміну	Активне з'єднання з базою 1С	Пройдено
UC-05	Менеджер експортує фінальний звіт для друку та архівації	Модуль експорту	Збережено PDF з деталізацією розрахунків	Схема, матеріали та ціни обчислені	Пройдено частково
UC-06	Користувач редагує палітру вручну	Модуль кольорокорекції	Змінено відображення кольорів та перераховано бісер	Вибрана схема	Пройдено

Варіант архітектури	Переваги	Недоліки	Складність реалізації	Масштабованість	Обрана альтернатив
Монолітна	Простота реалізації, швидкий старт	Важко оновлювати, неможливо масштабувати окремі модулі	Низька	Низька	-
Клієнт-серверна ( <b>2-tier</b> )	Відокремлення інтерфейсу та логіки, підтримка декількох користувачів	Високі вимоги до серверної частини, менше гнучкості	Середня	Середня	-
Модульна (багаторівнева)	Гнучкість у підтримці, можливість масштабування, незалежна розробка модулів	Більші витрати часу на проектування, потреба в чіткому менеджменті	Висока	Висока	✓
Мікросервісна	Незалежне масштабування, висока надійність, гнучкість	Занадто складна для невеликої команди, потребує <b>DevOps</b> та <b>CI/CD</b>	Дуже висока	Дуже висока	-

## Додаток Б

```
import numpy as np
import cv2
import matplotlib.pyplot as plt

def calculate_dominant_color(square):
    # Перетворюємо квадрат у формат, який можна використовувати для
    обчислення середнього кольору
    pixels = np.float32(square.reshape(-1, 3))
    # Обчислюємо середнє значення кольору
    mean_color = np.mean(pixels, axis=0)
    return mean_color

def get_bead_color(image, height_in_beads):
    height, width, _ = image.shape
    scale_factor = height_in_beads / height
    width_in_beads = int(scale_factor * width)

    # Розмір одного квадрата в пікселях
    square_width = width / width_in_beads
    square_height = height / height_in_beads

    bead_colors = []

    for y in range(height_in_beads):
        row_colors = []
        for x in range(width_in_beads):
            # Визначення координат квадрата в пікселях
            start_x = int(x * square_width)
```

```

start_y = int(y * square_height)
end_x = int((x + 1) * square_width)
end_y = int((y + 1) * square_height)

# Вирізання квадрата із зображення
square = image[start_y:end_y, start_x:end_x]

# Визначення переважаючого кольору
dominant_color = calculate_dominant_color(square)
row_colors.append(dominant_color)

bead_colors.append(row_colors)

return bead_colors, width_in_beads

def plot_grid(image, bead_colors, width_in_beads, height_in_beads):
    fig, ax = plt.subplots()
    ax.imshow(image)
    height, width, _ = image.shape
    square_width = width / width_in_beads
    square_height = height / height_in_beads

    for y in range(height_in_beads):
        for x in range(width_in_beads):
            rect = plt.Rectangle((x * square_width, y * square_height), square_width,
square_height,
                                linewidth=1, edgecolor='r', facecolor='none')
            ax.add_patch(rect)
            dominant_color = bead_colors[y][x]

```

```

        ax.text(x * square_width + square_width / 2, y * square_height +
square_height / 2,
                f'{int(dominant_color[0])}, {int(dominant_color[1])},
{int(dominant_color[2])}',
                color='white', ha='center', va='center', fontsize=8, weight='bold')
plt.show()

```

```
def main():
```

```
    # Завантаження зображення
```

```
    image_path = "path_to_your_image.jpg"
```

```
    image = cv2.imread(image_path)
```

```
    image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
```

```
    # Висота картини в бісеринках (задана користувачем)
```

```
    height_in_beads = 100
```

```
    # Отримання кольорів бісеру для кожного квадрата
```

```
    bead_colors, width_in_beads = get_bead_color(image, height_in_beads)
```

```
    # Відображення зображення з накладеною сіткою і кольорами
```

```
    plot_grid(image, bead_colors, width_in_beads, height_in_beads)
```

```
if __name__ == "__main__":
```

```
    main()
```