



Наталія ІЩУК

ДИЗАЙН МОБІЛЬНИХ ЗАСТОСУНКІВ



Київський національний університет імені Тараса Шевченка
Навчально-науковий інститут журналістики
Кафедра онлайн-медіа

Наталія ІЩУК

ДИЗАЙН МОБІЛЬНИХ ЗАСТОСУНКІВ

Навчальний посібник

Київ
ННІЖ
2025

УДК 004.42:658.512.23

I 98

*Рекомендовано вченою радою
Навчально-наукового інституту журналістики
Київського національного університету імені Тараса Шевченка
протокол № 14 від 8 квітня 2025 р.*

Рецензенти:

КРАЙНІКОВА Т. С., докторка наук із соціальних комунікацій, професорка кафедри редакційно-видавничих технологій і продюсування Навчально-наукового інституту журналістики

ЗЕЛІНСЬКА А. А., кандидатка наук із соціальних комунікацій, асистентка кафедри онлайн-медіа Навчально-наукового інституту журналістики

ІЩУК Н. М.

I 98 **Дизайн мобільних застосунків: навчальний посібник.** Київ, Навчально-науковий інститут журналістики, 2025. 104 с.

Посібник розроблено відповідно до навчальної програми з дисципліни «Основи медіавиробництва», яка викладається для студентів бакалаврату спеціальності 061 (С 7) «Журналістика». У посібнику містяться теоретичні відомості про дизайн, структуру, компоненти мобільних застосунків для платформ IOS та Android, рекомендації щодо створення каркасів, макетів і прототипів, а також огляд методів просування та монетизації мобільних застосунків.

Цей посібник може бути корисним для студентів-журналістів, дизайнерів, розробників мобільних застосунків, фахівців з UX/UI дизайну та всіх, хто цікавиться створенням зручних, сучасних та конкурентоспроможних цифрових продуктів.

УДК 004.42:658.512.23

© Іщук Н. М., 2025

© Навчально-науковий інститут журналістики, 2025

Зміст

Вступ	8
Розділ 1. Вступ до дизайну мобільних застосунків	9
<i>1.1. Еволюція мобільних операційних систем</i>	<i>9</i>
Коротка історія розвитку мобільних ОС (від перших смартфонів до сучасних IOS та Android)	9
Вплив операційних систем на дизайн мобільних застосунків	13
Основні тенденції у розвитку мобільних платформ і застосунків	15
<i>1.2. Основи мобільного дизайну</i>	<i>17</i>
Визначення дизайну мобільних застосунків і його роль у створенні продукту	17
Відмінності між веб-дизайном та мобільним дизайном	18
Тренди та сучасні підходи в мобільному UX/UI	19
<i>1.3. Психологія мобільних користувачів</i>	<i>23</i>
Як люди взаємодіють із мобільними застосунками ..	23
Поведінкові патерни мобільних користувачів	23
<i>1.4. Процес розробки мобільного дизайну</i>	<i>26</i>
Етапи створення мобільного застосунку з погляду дизайну	26
Взаємодія дизайнера з командою (продакт-менеджер, розробники, тестувальники) ...	28
Основні інструменти для роботи над дизайном мобільних застосунків	30
<i>1.5. Основи UX/UI для мобільних застосунків</i>	<i>33</i>
Що таке UX (user experience) і чому він важливий ...	33

Основи UI (user interface) та його роль у взаємодії користувача із застосунком	34
Принципи зручного та інтуїтивного дизайну	35
<i>Запитання до розділу 1</i>	<i>38</i>
<i>Завдання для самостійної роботи</i>	<i>38</i>

Розділ 2. Структура та компоненти дизайну мобільних застосунків для IOS/Android

2.1. Основи дизайну мобільних інтерфейсів

Відмінності між платформами IOS та Android

Гайдлайни та рекомендації Apple Human

Interface Guidelines і Google Material Design

2.2. Навігація та архітектура інтерфейсу

Основні патерни навігації (tab bar/bottom navigation, navigation drawer, tabs)

Ієрархія екранів та логіка переходів

UX-підхід до організації структури застосунку

2.3. Основні компоненти UI

Кнопки, поля введення, перемикачі, чекбокси

Меню, списки, картки

Діалогові повідомлення

2.4. Анімації та мікрвзаємодії

Роль анімації у покращенні UX

Принципи створення плавних переходів

2.5. Колір, типографіка та іконографія

Вибір кольорової схеми відповідно до UX/UI

Основи мобільної типографіки (шрифти, розміри, контраст)

Використання іконок у мобільних інтерфейсах

Створення іконки мобільного застосунку

Запитання до розділу 2	75
Завдання для самостійної роботи	75

Розділ 3. Каркас, макет, прототип мобільного

застосунку	76
<i>3.1. Основи проєктування мобільних інтерфейсів</i> ...	76
Відмінності між каркасом, макетом і прототипом	76
Етапи створення інтерфейсу: від ідеї до інтерактивного прототипу	76
Інструменти для проєктування	77
<i>3.2. Каркас (wireframe): структура майбутнього застосунку</i>	78
Що таке каркас і чому він важливий	78
Види каркасів: низькодеталізований (low-fidelity) та високодеталізований (high-fidelity)	79
Створення базової структури екранів та взаємодій ..	80
<i>3.3. Макет (mockup): візуальне оформлення</i>	80
Роль макету у процесі дизайну	80
Відповідність гайдлайнам IOS та Android	81
Типові помилки в макетах	82
<i>3.4. Прототип (prototype): інтерактивність та тестування</i>	84
Прототип як інструмент перевірки UX	84
Основи створення інтерактивних прототипів	85
Тестування та внесення змін перед розробкою	85
<i>3.5. Юзер-тестування та ітеративне покращення</i> .	86
Методи тестування прототипів	86
Аналіз зворотного зв'язку від користувачів	86
Ітеративне покращення мобільного застосунку	88
Запитання до розділу 3	90
Завдання для самостійної роботи	90

Розділ 4. Просування мобільних застосунків та монетизація	91
4.1. Стратегії просування мобільних застосунків ...	91
ASO (App Store Optimization): як підняти застосунок у пошуку	91
Оптимізація для Google Play	93
Оптимізація для App Store	94
4.2. Залучення та утримання користувачів	95
Гейміфікація як спосіб підвищення взаємодії	95
Сповіднення та персоналізація контенту	95
Аналітика користувацької поведінки	96
4.3. Моделі монетизації мобільних застосунків	97
Безкоштовні застосунки з рекламою	97
Підписки та внутрішні покупки	98
Платні застосунки та преміальні функції	98
<i>Запитання до розділу 4</i>	100
<i>Завдання для самостійної роботи</i>	100
Література	101

Вступ

Використання смартфона у щоденному житті стало настільки звичним, що користувачі навіть не замислюються, як часто тримають його в руках, аби перевірити останні новини, відповісти чи відреагувати на повідомлення, переглянути відео або сповіщення. Функція дзвінку перестала бути єдиною основною й відкрила можливості для контентних та сервісних мобільних застосунків. Конкуренція серед останніх продовжує зростати, витісняючи неактуальні, неестетичні чи навіть незручні у своєму використанні програми для смартфонів. Враховуючи всеохопну диджиталізацію, мало хто зможе назвати хоча б деякі основні елементи мобільного інтерфейсу: чекбокси, перемикачі, таби, тости, степери чи пікери.

Однак застосування кросплатформних підходів вимагає від журналістів розуміння не лише естетичного складника мобільних застосунків, а й психології мобільних користувачів, впровадження актуальних стратегій утримання їх уваги, доцільних практик зовнішньої мотивації та заохочення до інтеракцій.

Створення мобільного застосунку не належить до компетенцій майбутнього журналіста чи редактора, проте вміння реалізувати його концепцію від ідеї до інтерактивного прототипу є неабияк важливою для створення комунікаційної платформи та взаємодії з аудиторією, яка перебуває онлайн та використовує смартфон навіть під час взаємодії з іншими каналами комунікації.

Від автора



РОЗДІЛ 1. ВСТУП ДО ДИЗАЙНУ МОБІЛЬНИХ ЗАСТОСУНКІВ

1.1. Еволюція мобільних операційних систем

Коротка історія розвитку мобільних ОС (від перших смартфонів до сучасних IOS та Android)

З часу появи перших мобільних пристроїв, які не вирізнялися ані дизайном, ані функціоналом і до нинішнього дня операційні системи зазнали чималих змін і продовжують свій розвиток. Від примітивних платформ і до розвинених екосистем, таких як IOS та Android, вони динамічно змінюються як у результаті технологічного поступу, так і вимог користувачів. Сучасному користувачеві вже важко уявити смартфон без сенсорного дисплею чи відсутністю інтернет-зв'язку. Знадобився час, амбітні ідеї розробників та електротехніків, аби технології дійшли до тих, які для нас уже вважаються звичними.

BlackBerry як лідер бізнес-комунікацій

Історія BlackBerry як виробника електроніки розпочалася з пейджерів, а її засновник Майк Лазарідіс зробив сміливий крок, який визначив подальший шлях компанії. У 1984 році, за кілька місяців до випуску з Університету Ватерлоо, він вирішив покинути навчання. На той час йому залишало-



ся небагато до отримання ступеня бакалавра з електротехніки, але Лазарідіс вирішив зосередитися на розвитку компанії Research In Motion (RIM), яку заснував того ж року. Для Лазарідіса рішення кинути університет було ризикованим, але він бачив у цьому шанс реалізувати своє бачення нових технологій і створити революційний продукт.

Тому у 1996 році канадський стартап презентував свій перший пейджер Inter@ctive Pager 900. Особливого комерційного успіху новинка не мала, не в останню чергу через чималі габарити. Втім, за два роки RIM випустила більш компактний гаджет Inter@ctive Pager 950. Його головною перевагою стала клавіатура, на якій можна було друкувати та надсилати повідомлення.

Варто зазначити, що для просування своїх перших гаджетів компанія застосовувала доволі відомий на сьогодні маркетинговий прийом. Безкоштовні екземпляри пристроїв роздавали лідерам думок, знаменитостям та ентузіастам з Америки та Канади, які згодом стали амбасадорами бренду.

Таким чином гаджети RIM поступово набували популярності, а зрештою потрапили на Уолл Стріт та почали завойовувати прихильність бізнесменів.

Після ребрендингу нові пристрої почали називатися BlackBerry — цей крок був пов'язаний із клавіатурами, чий дизайн нагадував ягоди ожини.

Проте, коли на ринок вийшли iOS і Android, BlackBerry не змогла адаптуватися до трендів сенсорних екранів, що призвело до зниження її популярності.

Прорив в інноваціях від Apple

Цікаво, що Стів Джобс — генеральний директор Apple — був прихильником фруктових дієт, який надихнувся яблуком як символом простоти та доступності. Та чи був перший айфон таким же доступним за ціною?



9 січня 2007 року Apple презентував перший мобільний телефон компанії — iPhone 2G. Завдяки цьому гаджету Apple стала одним із лідерів на ринку та ввела в моду сенсорне управління.

Стів Джобс вважав, що сучасний смартфон повинен поєднувати у собі всі можливості телефона, музичного плеєра і кишенькового комп'ютера. На той час це було абсолютно нове явище: жодних фізичних кнопок, лише сенсорний екран.

У 2007 році фізична клавіатура була популярніша за сенсорну, і компанії не квапилися від неї відмовлятися. Однак, у липня цього ж року, коли iPhone надійшов у продаж, за перші 30 годин продали 270 тисяч айфонів. А через 3 місяці — було продано 1 млн гаджетів. Вартість iPhone 2G становила 499 доларів. Стало зрозуміло, що сенсорні екрани будуть стандартом.

Незважаючи на всі плюси, у першому iPhone було багато недоліків. Серед них: відсутність 3G; занадто тихий динамік; не було App Store та популярних застосунків; підтримки шпалер для робочого столу та можливості копіювати і вставляти текст.

Однак із кожним новим оновленням Apple покращувала свою систему, додаючи функції, на зразок FaceTime, Siri та потужних інструментів для безпеки.

Android — відкритість та адаптивність

Історія компанії Android Inc. розпочалася у 2003 році, коли 4 програміста Річ Майнер, Нік Сірс, Кріс Вайт та Енді Рубін вирішили заснувати компанію в Пало-Альто (Каліфорнія), яка мала виготовляти програмне забезпечення для цифрових камер. Колеги жартували, називаючи Енді Рубіна «Андроїдом» через його захоплення робототехнікою. Так і виникла назва андроїд.

Основним джерелом фінансування стали гроші Енді Рубіна, проте вони швидко закінчилися. І зрештою, засно-



вники зрозуміли, що ринок цифрових камер не має великого потенціалу, тому перейшли до розробки операційних систем для смартфонів, які могли б конкурувати з популярними на той час.

У 2005 році, розгледівши потенціал невеликої компанії, її придбала Google, а сума угоди склала близько 50 мільйонів доларів. Основні співробітники в рамках угоди залишилися і продовжили займатися розробкою.

Цікаво, що перший Android-смартфон 2008 року, HTC Dream, мав фізичну клавіатуру, оскільки Google спочатку орієнтувалася на стиль BlackBerry. Проте під впливом iPhone компанія швидко переробила концепцію на сенсорний екран.

Android її відкритість, дозволила численним виробникам створювати різноманітні пристрої, працюючи під управлінням цієї системи.

Android також зробила популярною ідею персоналізації, надаючи користувачам більше свободи налаштувань, ніж iOS.

Нині iOS і Android продовжують домінувати на ринку мобільних операційних систем. iOS залишає за собою лідерство в естетичному дизайні, стабільності та екосистемному підході, надаючи користувачам продукти, що тісно інтегровані між собою. Натомість Android залишається найбільш універсальною й відкритою платформою, яка підтримує різноманітні пристрої від численних виробників, що робить її доступною для ширшого кола споживачів.

За останні кілька років операційні системи продовжували еволюціонувати, додаючи нові функції, покращуючи безпеку, підтримку новітніх технологій і застосунків. Наприклад, з появою Android 12 і iOS 15, обидві платформи отримали значні оновлення в контексті персоналізації інтерфейсу, покращеної підтримки штучного інтелекту та інноваційних можливостей для розробників (Google, 2021; Apple, 2021).



Вплив операційних систем на дизайн мобільних застосунків

Операційні системи мобільних пристроїв відіграють визначальну роль у формуванні дизайну мобільних застосунків. Кожна з систем пропонує свої унікальні принципи, які впливають на зовнішній вигляд інтерфейсів, взаємодію користувачів із застосунками та їх функціональні можливості, які мають враховувати і дизайнери, і розробники. Незнання чи ігнорування принципів і правил систем негативно впливатиме на подальший користувацький досвід та ефективність і зручність взаємодії з застосунком.

Одним з основоположних чинників, що визначають дизайн мобільних застосунків, є гайдлайни операційних систем. Apple пропонує Human Interface Guidelines (HIG) та позиціонує простоту, чіткість й уніфікацію елементів інтерфейсу. Ці гайдлайни встановлюють правила для кожного елемента — від кнопок до анімацій — що дає змогу створити послідовний і зрозумілий інтерфейс для користувача. Згідно з дослідженнями професора Б. Дж. Фогга, пристосування до цих стандартів дає змогу не тільки покращити зручність використання, але й збільшити лояльність користувачів до бренду.

На противагу Google пропонує Material Design, який значно відрізняється від HIG, роблячи акцент у наближенні дизайну до реального фізичного світу з тінями і глибиною. Компоненти інтерфейсу детально прописані та забезпечують інтуїтивно зрозумілу навігацію, естетичність та адаптивність.

Візуальні особливості мобільних операційних систем також суттєво впливають на дизайн. У IOS акцент робиться на чіткість та естетику. Наприклад, система прагне до мінімалізму та акцентує увагу на створенні «порожнього простору», що допомагає користувачу зосередитись на осно-



вному контенті застосунку. Елементи навігації в IOS часто зменшують свою роль в інтерфейсі, надаючи більше уваги для контенту.

Натомість, Android використовує багатофункціональний підхід, який дає змогу створювати інтерактивні, насичені елементи для навігації. Це може бути досягнуто завдяки підтримці Material Design, що створює можливість використовувати динамічні анімації і переходи між екранами. Це забезпечує більш інтуїтивне сприйняття навігаційних процесів на платформі, але також вимагає більших зусиль для адаптації інтерфейсу під різні розміри екранів і дозволи пристроїв.

Вибір операційної системи також визначає технології, які використовуються для створення застосунків. Для IOS основними інструментами є мови програмування Swift і Objective-C, що дають змогу максимально ефективно використовувати ресурси пристроїв. Розробка застосунків для iOS орієнтована на забезпечення високої продуктивності та зручності користувача через інтеграцію з іншими продуктами Apple, такими як iCloud або Apple Watch (Apple, 2020).

Для Android основними мовами є Java і Kotlin, а також інструменти на основі Android SDK, що дають можливість створювати застосунки, здатні працювати на різноманітних пристроях з різними характеристиками. Це означає, що дизайн Android-застосунків має бути адаптованим до широкого спектру екранів, розмірів, що вимагає більшої гнучкості і пристосованості.

Операційні системи суттєво впливають на дизайн мобільних застосунків. Кожна платформа має свої гайдлайни, які визначають візуальні елементи, стилі навігації, технології та способи взаємодії з користувачем. Варто відзначити, що обидві платформи враховують принципи доступності для людей з обмеженими можливостями. НІГ ще й робить



акцент на інклюзивності, аби дизайн був зрозумілим для людей із різними когнітивними особливостями. Отже, врахування специфіки кожної ОС є ключовим для створення зручного, функціонального та ефективного дизайну мобільних застосунків.

Основні тенденції у розвитку мобільних платформ і застосунків

Розвиток мобільних платформ не стоїть на місці. Кожен рік привносить нові технології, функції та покращення, які змінюють підхід до розробки мобільних застосунків. Сучасні мобільні платформи, такі як iOS та Android, активно адаптуються до нових вимог користувачів і технологічних новинок. Щороку фахівці обговорюють тренди, які формуватимуть підходи до реалізації нових мобільних застосунків чи впровадженні поліпшень до вже наявних на ринку.

Інтеграція штучного інтелекту (AI)

Стрімке поширення штучного інтелекту (AI) та невпинного машинного навчання дало поштовх замислитися над впровадженням його в застосунках різного типу та призначення. Від покращених голосових асистентів до персоналізованих рекомендацій у застосунках — AI допомагає зробити мобільний досвід інтуїтивнішим і персоналізованішим. Алгоритми машинного навчання дають змогу застосункам краще передбачати потреби користувачів та оптимізувати їхній досвід на основі попередніх даних і збереженої історії.

Швидкість передачі даних і розвиток 5G

Швидкість передачі даних свого часу дала змогу відійти від текстового контенту, який пропонували перші новинні мобільні застосунки до мультимедійного. Згадується GPRS та EDGE, коли очікування на завантаження вважалися нормою і не давали змогу легко отримувати та обмінюватися контентом. Відтак, низька затримка мережі допомагають за-



стосункам працювати ще ефективніше та швидше, що відкриває нові горизонти для мобільних ігор в реальному часі, відеозв'язку та потокового передавання даних.

Популяризація доповненої реальності (AR)

Доповнена реальність не є чимось надто новим, однак її перспективи все ще обговорюються та поступово впроваджуються, залучаючи усе більшу аудиторію. Поєднання реального світу із віртуальними об'єктами, створює новий рівень взаємодії з користувачем. Платформи iOS та Android активно розвивають свої фреймворки для AR — ARKit для iOS і ARCore для Android. Наприклад, застосунки, що допомагають застосувати майкап, приміряти одяг, аксесуари чи предмети інтер'єру в реальному часі, вже стали популярними завдяки AR-технологіям. Можливість віртуальної подорожі до картинної галереї чи навіть отримання додаткової візуалізації під час футбольного матчу за допомогою смартфона стають новою реальністю і продовжуватимуть свій розвиток.

Покращення безпеки та конфіденційності

Оскільки мобільні пристрої стають основними засобами для обміну особистою інформацією, безпека і конфіденційність даних набувають усе більшої ваги. Як Android, так і iOS постійно вдосконалюють свої засоби захисту даних. Це включає впровадження більш складних методів автентифікації, таких як біометричні технології (відбитки пальців, розпізнавання обличчя) та вдосконалення механізмів шифрування даних. Натомість мобільні застосунки змінюють підходи до реєстрації та пропонують автентифікацію через Google-акаунти, OTP коди, соціальні мережі чи за допомогою відбитків пальців.

Гейміфікація у застосунках

Тенденції, які привнесли ігрові форми в неігрових контекстах продовжують розвиватися в застосунках медіа.



Оскільки ігри мають одну з найбільших аудиторій, це стимулює заохочувати користувачів долучатися до квестів, міні-ігор, тестів, конкурсів чи навіть пропонувати гру, коли користувач втратив інтернет-з'єднання і не може продовжити користуватися мобільним застосунком.

Інтеграція з інтернетом речей (IoT)

Смарт-пристрої, поєднані за допомогою інтернет-зв'язку до смартфона дають змогу користувачам керувати ними безпосередньо через свої мобільні застосунки (роботи-пилососи, фітнес-трекери, розумні розетки і лампи тощо). Завдяки цим можливостям користувачі можуть автоматизувати свою оселю, керувати різними пристроями з одного інтерфейсу та створювати інтегровані сценарії.

Розвиток голосових функцій

Голосові інтерфейси стали невід'ємною частиною сучасних мобільних платформ. Завдяки вдосконаленим голосовим асистентам, таким як Siri і Google Assistant, користувачі можуть з легкістю взаємодіяти з пристроєм, використовуючи лише голосові команди. Саме під час пандемії COVID-19 питання голосових функцій почало обговорюватися більш активно, аби забезпечити зменшення зайвого доторкання до смартфона й керувати ним за допомогою голосу (з'явилося більше голосового пошуку товарів, голосові команди в застосунках відеоконференцій, розширене голосове керування авто, наприклад, з Android Auto тощо).

1.2. Основи мобільного дизайну

Визначення дизайну мобільних застосунків і його роль у створенні продукту

Дизайн мобільних застосунків є одним із ключових етапів у процесі створення програмного продукту. Це передбачає розробку зручного та естетично привабливого інтерфейсу, що сприяє ефективній взаємодії користувача з застосунком.



Дизайн мобільних застосунків — це комплекс процесів, які охоплюють:

UI-дизайн (User interface) — розробку зовнішнього вигляду застосунку, зокрема кнопок, іконок, шрифтів, кольорової гами та загального стилю.

UX-дизайн (User experience) — створення логічної та інтуїтивно зрозумілої структури взаємодії користувача з продуктом.

Дизайн впливає не лише на зовнішній вигляд застосунку, але й на його функціональність, зручність використання та рівень залучення користувачів.

Якісний дизайн мобільного застосунку:

- покращує досвід його використання — логічно організований контент та інтуїтивний інтерфейс роблять використання застосунку простішим і приємнішим;
- утримує свою аудиторію — добре продуманий дизайн допомагає уникнути розчарування користувачів та мотивує їх повертатися до застосунку;
- формує кокурентну перевагу — привабливий і функціональний інтерфейс може стати ключовою перевагою над аналогічними продуктами на ринку;
- оптимізує процеси — зручний дизайн сприяє швидшому виконанні мети, що підвищує ефективність роботи користувачів.

Таким чином, дизайн мобільного застосунку є важливим компонентом у створенні цифрового продукту, що впливає на його успіх та популярність серед користувачів.

Відмінності між веб-дизайном та мобільним дизайном

І хоча веб та мобільний дизайн мають спільні принципи UX/UI, між ними є суттєва різниця, яку потрібно враховувати, створюючи новий продукт чи забезпечуючи кросплатформність.



1. Адаптація до різних пристроїв

Веб-дизайн орієнтований на великі екрани (настільні ПК, ноутбуки, планшети). Мобільний дизайн створюється для менших екранів смартфонів, де важливі компактність і зручність взаємодії.

2. Способи взаємодії

Веб-дизайн розрахований на використання клавіатури та миші. Мобільний дизайн враховує сенсорне керування, жести, натискання та свайпи.

3. Навігація та структура

Веб-дизайн часто використовує меню, що розгортається, або верхні панелі навігації. Мобільний дизайн має мінімалістичні меню, часто приховані в «гамбургер/бургер/бен-то/донер/кебаб/мітбол»-іконці або нижній панелі.

4. Продуктивність і швидкість

Веб-сайти завантажуються через браузер, що може впливати на швидкість роботи. Мобільні застосунки оптимізовані для швидкої роботи на смартфонах, можуть працювати офлайн.

5. Обмеження в дизайні

Веб-дизайн дає змогу використовувати більше графіки, складні макети та інтерактивні елементи. У мобільному дизайні потрібно уникати дрібних елементів та складних анімацій, щоб не перевантажувати інтерфейс.

Отже, дизайн мобільних застосунків відрізняється своєю простотою, адаптивністю та орієнтацією на користувачів, які взаємодіють із продуктом через сенсорний екран.

Тренди та сучасні підходи в мобільному UX/UI

Сучасний мобільний дизайн постійно змінюється, враховуючи нові технології та потреби користувачів. Щороку фахівці обговорюють основні тренди у сфері мобільного UX/UI, але зрештою тенденції продовжують залишатися такими:



1. Мінімалізм та простота

Менше складних елементів — більше простору та зручності. Чистий дизайн із простими формами, великою типографікою та мінімальним використанням тексту забезпечує зручність користування.

The screenshot displays the New York Times mobile application interface. At the top, the logo "The New York Times" is centered. Below it, a "LIVE" indicator and "12m ago" timestamp precede the main article title: "Markets Tumble After Trump's Talk of a Recession". The article text states that stock markets fell after President Trump's comments. A "See more updates" link is provided. Below the article is a line chart titled "S&P 500 Index" showing market fluctuations from March 6 to March 10. The chart's y-axis ranges from 5,650 to 5,800. Below the chart, a navigation bar includes icons for Today, Play, Sections, and You. To the right, a vertical list of other articles is shown, including "Art Schallock, Oldest Surviving Major Leaguer, Dies at 100", "Flo Fox, 79, Dies; Street Photographer Overcame Blindness and Paralysis", "As Utility Bills Soar, New Yorkers Face the Cost of a Greener Future", and "I Started a Conversation With a Woman Sitting to My Right". Each article includes a thumbnail image, a timestamp, and a share icon.

The New York Times

2h ago

LIVE 12m ago

Markets Tumble After Trump's Talk of a Recession

Stock markets around the world fell a day after President Trump did not rule out the possibility that his trade policies could cause a recession this year.

See more updates >

S&P 500 Index

Date	Index Value
March 6	~5,750
March 7	~5,700
March 10	~5,650

Data delayed at least 15 minutes • By The New York Times • Source: Factset

Trump's Tactics Lead Americans to Question Role on World Stage

Among supporters and detractors alike, President Trump's transactional approach to foreign policy has upended old notions about the U.S. as a global leader.

Today Play Sections You

Art Schallock, Oldest Surviving Major Leaguer, Dies at 100

A pitcher, he played for the Yankees and the Orioles. When Mickey Mantle was sent to the minors in 1951, Schallock was called up.

18h ago

Flo Fox, 79, Dies; Street Photographer Overcame Blindness and Paralysis

She was legally blind and used a motorized wheelchair, but she managed to capture what she called the "ironic reality" of New York City on film.

12h ago

As Utility Bills Soar, New Yorkers Face the Cost of a Greener Future

The utility that serves New York City and Westchester County has filed a request to raise its rates to help pay for the shift to cleaner energy, sparking dismay among residents.

1d ago

I Started a Conversation With a Woman Sitting to My Right'

Seats up close for a Broadway show, a friendly question on the 7 and more reader tales of New York City in this week's Metropolitan Diary.

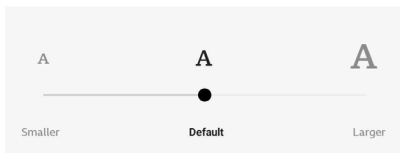
1d ago 281 Comments

Рис. 1–2. Мінімалізм застосунку «New York Times»

2. Персоналізація інтерфейсу

Багато застосунків нині пропонують вибір кольорової гами, темного/світлого режиму та розміру шрифту.

Text size



Preview

Headline 1

At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga.

Settings

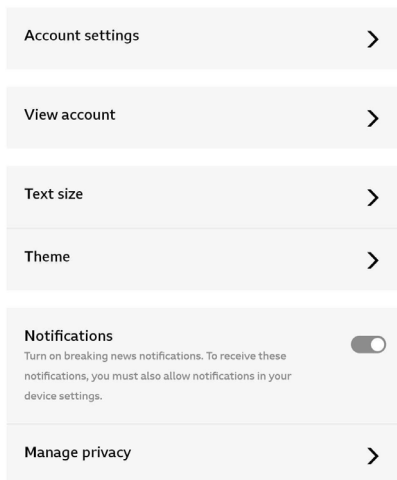


Рис. 3–4. Персоналізація теми, кеглю та сповіщень застосунку «BBC»

3. Неоморфізм

Цей стиль поєднує плаский дизайн із реалістичними тінями та світлом, що створює ефект фізичних об'єктів.

4. Адаптивність інтерфейсу

Сучасні мобільні застосунки автоматично підлаштовуються під різні типи пристроїв, розширень екрану та навіть гнучкі дисплеї.

5. Мікроанімації та мікровзаємодії

Невеликі анімації допомагають користувачам розуміти зв'язок між діями та їх результатом (наприклад, зміна кольору кнопки при натисканні).

6. Голосові інтерфейси та штучний інтелект

Дедалі більше застосунків інтегрують голосові команди та AI-рекомендації для персоналізації досвіду. Нестача часу для набирання тексту стає поштовхом для голосових функцій.

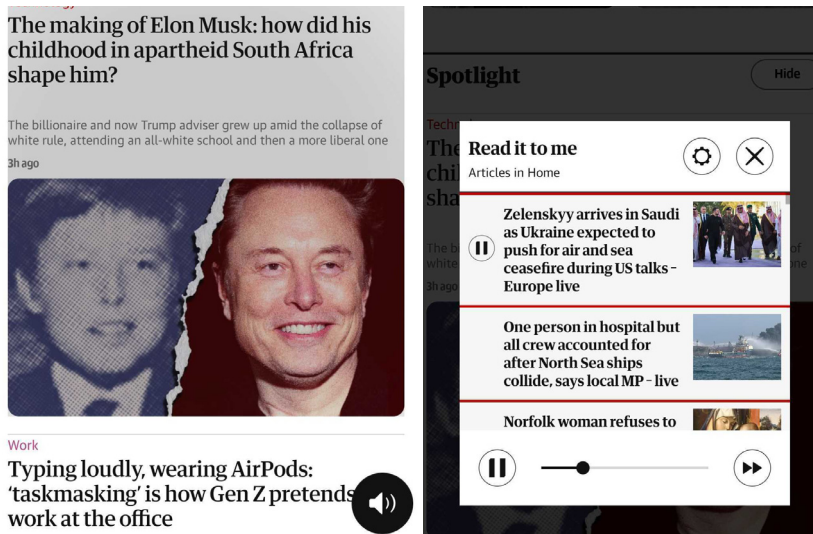


Рис. 5–6. Функція прочитання новини в застосунку «Guardian»

7. Гейміфікація

Впровадження ігрових елементів (досягнень, винагород, рівнів) робить користування застосунком захопливішим.

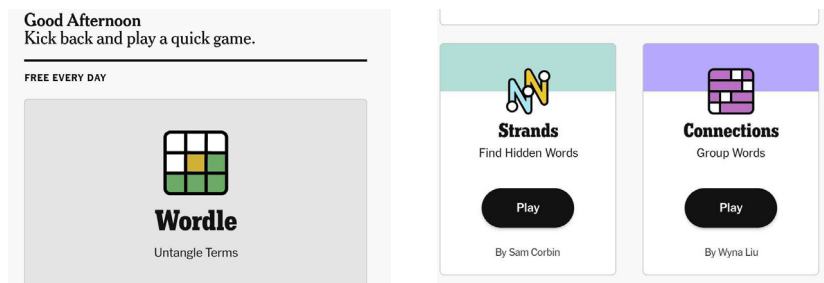


Рис. 7–8. Міні-ігри в «New York Times»

8. Біометрична автентифікація

Використання відбитків пальців, розпізнавання обличчя та інших методів для швидкого входу в застосунки замість традиційних мейла та пароля.

9. Інтеграція з іншими сервісами

Синхронізація з календарем, соціальними мережами чи розумним будинком для покращення користувацького досвіду.

Відтак, сучасні тенденції мобільного UX/UI спрямовані на створення інтуїтивних, швидких та персоналізованих застосунків, що відповідають потребам користувачів.

1.3. Психологія мобільних користувачів

Як люди взаємодіють із мобільними застосунками

Мобільні застосунки стали невід'ємною частиною повсякденного життя, і їхня взаємодія з користувачем визначає, наскільки успішним буде продукт. Девальвація часу, місця і способу отримання інформації стають вкрай актуальними для мобільних користувачів, які відкривають застосунки у транспорті, в черзі, на роботі, вдома та навіть перед сном. І ця динаміка суттєво впливає на те, як потрібно проектувати інтерфейси. Якщо застосунок складний, перевантажений зайвою інформацією або змушує користувача довго розбиратися в його функціоналі, швидше за все сприйтиме подальшому його видаленню та встановленню іншого застосунку серед численних конкурентів. Саме тому розуміння поведінкових патернів мобільних користувачів є ключовим для створення зручного й привабливого дизайну.

Поведінкові патерни мобільних користувачів

Люди взаємодіють із мобільними застосунками не так, як із веб-сайтами на комп'ютері. На мобільних пристроях вони очікують швидких реакцій, мінімальної кількості дій та інтуїтивного розташування елементів. Багато користува-



чів утримують телефон однією рукою, тому найважливіші кнопки мають бути доступні великим пальцем без зайвих зусиль. Також вони звикли до жестів: свайпи, довге натискання, подвійні тапи — усе це має працювати природно, відповідно до звичних сценаріїв (Fogg, 2009).

Для утримання користувачів та їх уваги, бажання повертатися до мобільного застосунку знов і знов — він повинен швидко вирішувати їх потреби. Люди відкривають застосунки з конкретною метою — знайти інформацію, купити товар, замовити їжу, переглянути новини. Якщо шлях до виконання цих дій занадто довгий або незрозумілий, користувачі просто залишають застосунок. Тому важливо спростувати взаємодію, залишаючи тільки найнеобхідніше. Наприклад, якщо застосунок передбачає реєстрацію, краще дати можливість увійти через пошту, ID чи соціальні мережі, аніж змушувати заповнювати довгі форми.

Дизайнери також повинні враховувати емоційний складник. Користувачі відчувають задоволення від добре продуманих мікроанімацій, плавних переходів та інтерактивних елементів, які підтверджують їхні дії. Наприклад, кнопка, що злегка змінює колір або розмір при натисканні, дає миттєву реакцію і робить UX більш «живим».

Ще один важливий аспект — передбачуваність інтерфейсу. Люди не люблять несподіваних змін або складних елементів, які потребують роз'яснень. Якщо навігація в застосунку працює так само, як у популярних застосунках (наприклад, іконка «лупа» завжди відповідає за пошук, а «кошик» — за покупки), користувачам не доведеться вчитися заново. Однак, вірто враховувати і питання креативного підходу — якщо всі мобільні застосунки використовуватимуть зрозумілі іконки, це буде не цікаво, тож, якщо говорити про нижнє меню і є потреба додати креативу — іконки повинні мати текстівку, аби користувач міг зрозуміти меню.



Приваблибий і зручний інтерфейс — це баланс між функціональністю, простотою та швидкістю. Якщо дизайн допомагає людям швидко отримати бажаний результат, створює позитивний емоційний досвід і відповідає їхнім звичкам, застосунок має всі шанси стати улюбленим та постійно використовуваним.

Однією з найвідоміших моделей, що пояснює поведінку користувачів, є модель поведінки Фогга (Fogg Behavior Model), розроблена Бі Джей Фоггом у Стенфордському університеті. Відповідно до цієї моделі, для виникнення певної поведінки необхідна одночасна наявність трьох ключових чинників: мотивації, можливості та тригера (Fogg, 2009).

Мотивація — бажання користувача виконати певну дію. Чим вища мотивація, тим більше ймовірність, що користувач виконає потрібну дію.

Можливість — наскільки просто користувачеві виконати дію. Якщо інтерфейс складний або дії потребують зайвих зусиль, користувач може відмовитися від виконання дії.

Тригер — зовнішній чи внутрішній сигнал, що спонукає користувача до дії (наприклад, push-сповіщення).

Якщо хоча б один із цих компонентів відсутній, поведінка не відбудеться. Це можна метафорично порівняти із такою ситуацією — у вас є велике бажання взяти участь у велоперегонах, про подію ви дізналися з медіа, але є один нюанс — у вас немає велосипеда. Тож, відсутність можливості не дасть змоги взяти участь у марафоні.

Розглянемо цю ситуацію з іншого боку — є велосипед і бажання взяти участь у веломарафоні, але на вулиці йде дощ. І тут, вирішальним є тригер. Ви отримали сповіщення, в якому дізналися, що ваші друзі збираються взяти участь теж. Тож, це підштовпує головного героя не зважати на погоду. Наявність можливості, мотивації та тригеру переборять погодні умови.

Подібна ситуація і з використанням застосунку: навіть якщо у користувача висока мотивація ним скористатися, але інтерфейс перевантажений або він не отримує жодного нагадування (тригера), бажана дія може не відбутися. Пригадуються тригери: товар завершується, тимчасовий купон а покупку, який збіжить через 48 годин, кількість друзів, які збираються взяти участь у події/ стежать за сторінкою.

1.4. Процес розробки мобільного застосунку

Етапи створення мобільного застосунку з погляду дизайну

Процес створення мобільного застосунку з погляду дизайну проходить через кілька ключових етапів, кожен з яких відіграє важливу роль у забезпеченні зручності використання, привабливого вигляду та ефективної взаємодії з користувачем.

Перший етап — це дослідження та аналіз, контекстуальне опитування, під час якого визначається цільова аудиторія, її потреби, проблеми та очікування. Аналіз ринку та конкурентів допомагає виявити сильні та слабкі сторони подібних застосунків, а також зрозуміти, які підходи працюють найкраще.

Другим етапом стає створення концепції майбутнього проекту, коли важливо сформулювати основну мету продукту, визначивши його функціональні можливості та унікальну цінність на основі потреб аудиторії.

Наступним етапом є розробка інформаційної архітектури, яка визначає логіку розташування контенту, навігацію між екранами та основні сценарії взаємодії. Для цього створюється карта застосунку (user flow), яка відображає можливі шляхи користувача, а також здійснюється логічне групування інформації, що сприяє кращому сприйняттю контенту та зручному використанню інтерфейсу.

Коли структура застосунку визначена, розробляються wireframe-макети — базові каркаси, що відображають роз-



ташування елементів на екранах без деталізації. Спочатку створюються спрощені ескізи (low-fidelity wireframes), а потім деталізовані варіанти (high-fidelity wireframes), які охоплюють основні компоненти інтерфейсу. Це допомагає ще на ранніх етапах проектування оцінити зручність навігації та оптимізувати користувацький досвід.

Наступний крок — розробка візуального оформлення, або UI-дизайн. Визначається загальний стиль, кольорова гама, шрифти, кнопки, іконки та інші графічні елементи. Важливо враховувати рекомендації Apple Human Interface Guidelines та Google Material Design, аби інтерфейс відповідав вимогам платформ iOS чи Android, якщо мова йде про нативні мобільні застосунки. На цьому етапі створюються деталізовані макети (mockups), які максимально наближені до його фінального вигляду.

Після цього розробляється інтерактивний прототип, який дає змогу оцінити, як працюватиме застосунок на практиці. Завдяки спеціальним інструментам, створюються клікабельні моделі застосунку, які дають можливість тестувати взаємодію користувача з інтерфейсом ще до початку програмування. Проведення юзер-тестування допомагає отримати зворотний зв'язок та внести необхідні коригування, щоб усунути можливі недоліки.

Коли дизайн остаточно затверджений, починається підготовка до передачі макетів розробникам. Важливо надати всі графічні ресурси у відповідних форматах, а також детальні специфікації щодо розмірів, кольорів, відступів та поведінки UI-елементів. Співпраця дизайнера з командою розробників на цьому етапі є важливою, оскільки допомагає уникнути розбіжностей між задумом та реалізацією.

Завершальним етапом є тестування дизайну після його імплементації у код. Проводиться перевірка відповідності розробленого інтерфейсу затвердженим макетам, оцінюють-



ся юзабіліті, а також здійснюється тестування адаптивності на різних пристроях. Лише після усунення всіх недоліків застосунок готовий до випуску в App Store та Google Play.

Таким чином, створення дизайну мобільного застосунку — це багатовітупний процес, що вимагає глибокого аналізу, продуманого проектування, ретельного тестування та тісної співпраці з розробниками. Дотримання цього підходу допомагає створити зручний, привабливий і конкурентоспроможний продукт.

Взаємодія дизайнера з командою (продакт менеджер, розробники, тестувальники)

Дизайнер мобільного застосунку — це не лише творець візуального складника продукту, а й активний учасник командної роботи. Для успішної реалізації проекту важливо, аби дизайнер ефективно взаємодіяв з іншими членами команди: продакт менеджером, розробниками та тестувальниками — лише ефективна командна взаємодія приносить добрі результати.

Взаємодія з продакт менеджером (*product manager, PM*)

Роль PM: визначає критерії продукту чи його частини, керує розробкою, окреслює довготривалі чи короткотривалі (наприклад, сезонні чи проміжні) бізнес-цілі, пріоритети та ключові функціональні можливості застосунку в цілому чи його частин.

Як дизайнер співпрацює з PM?

- Обговорює загальну концепцію продукту, його цілі та призначення.
- Визначає ключові завдання UX/UI, які повинні вирішувати проблеми користувачів.
- Узгоджує пріоритети дизайну відповідно до бізнес-стратегії (поетапність створення макетів та завдання).



- Спільно аналізує результати користувацьких тестувань та формує макети з поліпшення.

PM може надати аналітичні дані про поведінку користувачів у застосунку, а дизайнер на основі цих даних оновить інтерфейс для покращення юзабіліті. Для аналізу поведінки користувачів можуть братися до уваги такі сервіси: Amplitude, LogRocket, Instabug тощо.

Взаємодія з розробниками (developers, dev team)

Роль розробників: перетворюють дизайн у робочий код та забезпечують технічну реалізацію застосунку.

Як дизайнер співпрацює з розробниками?

- Передає макети та гайдлайни, використовуючи, наприклад, Figma.

- Надає специфікації для UI-компонентів (кольори, розміри, анімації, адаптивність).

- Враховує технічні обмеження платформи (IOS/Android) при розробці дизайну.

- Регулярно проводить синхронізаційні зустрічі для уточнення деталей та внесення коригувань.

Якщо дизайнер пропонує складну анімацію для екрану завантаження, розробники можуть пояснити, що її реалізація займе багато часу, і тоді буде знайдено альтернативне рішення.

Взаємодія з тестувальниками (quality assurance)

Роль тестувальників: перевіряють коректність роботи застосунку та виявляють баги, зокрема пов'язані з UI/UX.

Як дизайнер співпрацює з тестувальниками?

- Разом з QA-інженерами перевіряє відповідність дизайну фактичній реалізації.

- Аналізує звіти про помилки, пов'язані з неправильним відображенням елементів або некоректною роботою інтерфейсу.

- Допомогає уточнювати вимоги для тест-кейсів, що стосуються користувацького досвіду.



- Пропонує UX-поліпшення на основі реальних сценаріїв використання, знайдених під час тестування.

Тестувальник виявив, що, за певних умов, користувачеві повертається помилка і її потрібно оформити. Дизайнер переглядає випадок і пропонує, скажімо, снєкбар для кращої візуалізації, оновлює макет і передає варіант розробникам.

Загальні принципи ефективної взаємодії

Комунікація та взаємодія команди є основною, тож зустрічі-обговорення (refinement мітинги, дизайн-рев'ю, демо готових результатів або після імплементації розробником для обговорення в команді).

Гнучкість — готовність до змін та адаптації дизайну відповідно до технічних або бізнес-вимог.

Документація — детальні дизайн-гайди для команди розробки та QA.

Користувацьке тестування/внутрішнє тестування членами команди чи компанії — спільний аналіз з PM, Dev і QA, щоб покращити якість продукту.

Злагоджена взаємодія дизайнера з продакт менеджером, розробниками та тестувальниками — ключовий чинник успішного створення мобільного застосунку. Взаєморозуміння, спільна робота над ітераціями та постійний обмін інформацією, бізнес-логікою та юзер флоу допомагають створити якісний продукт, який відповідає очікуванням користувачів.

Основні інструменти для роботи над дизайном мобільних застосунків

Розробка дизайну мобільного застосунку потребує використання різних інструментів для створення макетів, прототипування, передання дизайну розробникам та тестування інтерфейсу. Нижче розглянемо основні інструменти, які використовуються у процесі роботи.

Ці програми допомагають дизайнерам створювати візуальні макети, налаштовувати стиль та компоненти інтерфейсу.

Figma — один із найпопулярніших інструментів для UI/UX-дизайну, що підтримує спільну роботу в реальному часі. Використовується для створення макетів, прототипів та дизайн-систем.

Adobe XD — гнучкий інструмент для створення макетів та інтерактивних прототипів, інтегрований з іншими продуктами Adobe.

Sketch — популярний серед дизайнерів iOS, використовується для створення інтерфейсів та дизайн-систем.

Інструменти для прототипування та тестування UX

Прототипування дає змогу оцінити, як користувачі взаємодіють із застосунком до початку розробки.

InVision — створення інтерактивних прототипів, що допомагає дизайнерам і розробникам тестувати UX.

Marvel — позиціонує себе як сервіс швидкого прототипування, тестування та передачі (дизайну) для сучасних дизайнерських команд.

Axure — інструмент, який надає UX-фахівцям можливість створювати реалістичні, функціональні прототипи та поширювати для спільного доступу команди.

ProtoPie — потужний інструмент для створення складних анімацій та інтерактивних прототипів.

Maze — сервіс для тестування прототипів на реальних користувачах та аналізу їх поведінки.

Інструменти для взаємодії з розробниками (design handoff)

Ці сервіси допомагають передавати готові макети дизайну команді розробників з усіма специфікаціями.

Zeplin — дозволяє експортувати макети з Figma, Adobe XD та Sketch у вигляді зрозумілої документації для розробників.



Figma Dev Mode — вбудований режим у Figma для передавання дизайну розробникам із доступом до всіх властивостей елементів.

Інструменти для створення дизайн-систем

Дизайн-системи забезпечують єдиний стиль інтерфейсу для всього застосунку. Це пришвидшує подальшу роботу над проектом і дає змогу перевикористовувати компоненти.

Figma Libraries — допомагає створювати та керувати глобальними компонентами, такими як кнопки, іконки, стилі шрифтів.

Storybook — використовується розробниками та дизайнерами для створення та документування UI-компонентів.

Zeroheight — сервіс для створення та ведення дизайн-документації.

Інструменти для створення іконок, графіки та анімації

Окрім стандартних UI-елементів, дизайн мобільного застосунку включає іконки, ілюстрації та анімації.

Adobe Illustrator — використовується для створення векторної графіки та іконок.

Lottie — інструмент для створення та інтеграції анімованої графіки у мобільні застосунки (формат JSON-анімацій).

Principle — застосовується для створення анімованих інтерфейсів та інтерактивних переходів між екранами.

Інструменти для спільної роботи та організації процесу

Ефективна співпраця між дизайнерами, розробниками та менеджерами вимагає використання платформ для комунікації та управління завданнями.

Slack — корпоративний месенджер для обговорення дизайну та взаємодії з командою.

Trello/Jira/Shortcut/Linear — платформи для управління завданнями, що допомагають відстежувати прогрес роботи на всіх етапах, починаючи від дизайну.



Notion — використовується для зберігання документації, нотаток та ідей щодо дизайну чи бізнес-логіки.

Вибір інструментів для роботи над дизайном мобільного застосунку залежить від конкретного проекту та процесів у команді. Figma є універсальним рішенням для більшості дизайнерських задач, але у складних проектах можуть використовуватися додаткові інструменти для тестування, прототипування та взаємодії з розробниками. Використання цих сервісів допомагає створювати якісні та ефективні мобільні застосунки.

1.5. Основи UX/UI для мобільних застосунків

Що таке UX (user experience) і чому він важливий

UX, або досвід користувача, — це загальне враження людини від взаємодії з мобільним застосунком. Це не лише про естетику інтерфейсу, а й про зручність, ефективність і комфорт користування. Хороший UX означає, що людина може легко виконати потрібні дії, не витрачаючи зайвий час на роздуми чи пошуки функцій.

Важливість UX важко переоцінити. Якщо застосунок інтуїтивний, зручний і приємний у використанні, аудиторія з більшою ймовірністю залишатиметься в ньому, повертатиметься та рекомендуватиме іншим. Навпаки, якщо інтерфейс незрозумілий, а процеси складні, люди швидко закриють застосунок і видалять його. Якісний UX допомагає зменшити рівень відмов, підвищує лояльність користувачів і навіть впливає на бізнес-показники, адже зрозумілий процес оформлення покупки чи замовлення напряму впливає на дохід компанії.

UX також формує конкурентну перевагу. У світі мобільних застосунків, де багато аналогічних сервісів, саме досвід користувача може стати вирішальним чинником. Наприклад, якщо один банківський застосунок дає змогу зробити



переказ у два кліки, а інший змушує проходити кілька складних кроків, користувач швидше обере перший.

Окрім того, гарний UX зменшує витрати на підтримку, оскільки користувачі менше звертаються до служби підтримки через проблеми з навігацією чи функціоналом. Це також знижує витрати на доопрацювання застосунку в майбутньому, оскільки добре продуманий дизайн не потребує значних змін.

Отже, UX — це не просто питання зручності, а й стратегічний елемент успіху будь-якого мобільного застосунку. Він впливає на лояльність користувачів, фінансові показники компанії та позицію продукту на ринку.

Основи UI (user interface) та його роль у взаємодії користувача з застосунком

UI (User Interface), або користувацький інтерфейс, — це графічний складник мобільного застосунку, з яким взаємодіє користувач. Він охоплює всі візуальні елементи: кнопки, іконки, текстові поля, меню, кольорову палітру, анімації та інші компоненти. Якісний UI має бути не лише естетично привабливим, а й зрозумілим, щоб користувач міг інтуїтивно знаходити потрібні функції та легко виконувати необхідні дії.

Головне завдання UI — зробити навігацію та використання застосунку максимально зручними та зрозумілими. Якщо інтерфейс складний або нелогічний, навіть найкорисніший функціонал може залишитися непоміченим, що зменшить ефективність застосунку. Наприклад, якщо важливі кнопки розташовані в незвичних місцях або недостатньо помітні, користувачі можуть витратити більше часу на пошук потрібних функцій, що може їх розчарувати.

Щоб UI був ефективним, він має відповідати кільком ключовим принципам:



Візуальна ієрархія — важливі елементи мають бути помітнішими за другорядні.

Читабельність — текст повинен легко сприйматися, а кольори мають забезпечувати достатній контраст.

Єдність стилю — усі елементи дизайну повинні дотримуватися загальної концепції, щоб користувач не плутався.

Інтерактивність — елементи інтерфейсу мають реагувати на дії користувача (анімації, зміна кольору кнопок, ефекти при натисканні).

Сучасні UI-дизайни активно використовують мікровзаємодії та плавні анімації, щоб покращити сприйняття застосунок. Наприклад, коли кнопка змінює вигляд після натискання або коли сторінка оновлюється із плавним переходом, це робить взаємодію більш природною та комфортною.

Отже, UI є не просто декоративним елементом, а ключовою частиною користувацького досвіду. Добре продуманий інтерфейс допомагає людям швидко освоїти застосунок, ефективно з ним працювати та отримувати задоволення від використання, що напряму впливає на успішність продукту.

Принципи зручного та інтуїтивного дизайну

Зручний дизайн мобільного застосунок забезпечує користувачам легкість у навігації, швидке виконання завдань і позитивний досвід використання. Головна його мета — зробити інтерфейс настільки інтуїтивним, щоб люди могли взаємодіяти з застосунком без зайвих пояснень чи навчання.

Одним із ключових принципів є простота. Дизайн має бути зрозумілим, без зайвих елементів, які можуть відволікати чи ускладнювати використання. Чим лаконічніше подано інформацію, тим легше користувачам знайти те, що їм потрібно.



«Проблема з простим дизайном полягає не в тому, що він занадто простий, а в тому, що він часто не достатньо простий», — Д. Норман «The Design of Everyday Things»

Важливу роль відіграє логічна навігація. Користувачі повинні швидко знаходити необхідні функції та не губитися у структурі застосунку. Елементи керування мають бути розташовані у звичних місцях, а кнопки та значки повинні зрозуміло вказувати на свої функції.

«Навігація повинна бути легкою та логічною, дозволяючи користувачам швидко й ефективно знаходити те, що їм потрібно», — Б. Шнайдерман, К. Плезант, М. Коен та С. Джейкобс «Дизайн інтерфейсу користувача: стратегії ефективної взаємодії людини з комп'ютером»

Ще один принцип — передбачуваність. Д. Норман детально описує важливість інтуїтивно зрозумілих інтерфейсів, зручності та передбачуваності в дизайні. Дії користувачів повинні давати очікуваний результат.

У книзі «Дизайн інтерфейсу користувача: стратегії ефективної взаємодії людини з комп'ютером» колектив авторів також наголошують на візуальній ієрархії, яка допомагає направляти увагу користувача. Важливі кнопки та ключові функції мають бути більш помітними, аніж другорядні елементи. Це досягається за допомогою контрасту, розміру шрифту, кольорових акцентів та відступів.

Послідовність у дизайні забезпечує єдиний стиль взаємодії на всіх екранах застосунку. Використання однакових кольорів, шрифтів і компонентів допомагає створити зрозумілий інтерфейс, який не вимагає від користувача щоразу адаптуватися до нових умов.

Не менш важливим є зворотний зв'язок. Коли користувач виконує певну дію, застосунок має підтверджувати її



виконання — це може бути коротке повідомлення, анімація або вібрація, що сигналізує про зміну стану.

Дизайн також повинен бути доступним для всіх категорій користувачів. Високий контраст, можливість збільшення шрифту та підтримка альтернативних методів навігації (наприклад, голосові команди) роблять застосунок зручним для ширшої аудиторії.

«Доступність — це основний принцип дизайну, який забезпечує, щоб продукти та системи були доступними для всіх людей, незалежно від їхніх можливостей або обмежень», — Б. Шнайдерман, К. Плеазант, М. Коен та С. Джейкобс «Дизайн інтерфейсу користувача: стратегії ефективної взаємодії людини з комп'ютером».

Використання звичних UX-патернів спрощує адаптацію до застосунку. Якщо елементи керування розташовані так само, як у популярних застосунках, користувачі інтуїтивно розуміють, як ними користуватися.

Нарешті, тестування на реальних користувачах допомагає знайти слабкі місця інтерфейсу та виправити їх до фінального релізу. Регулярне вдосконалення UX/UI на основі зворотного зв'язку сприяє створенню дійсно зручного продукту.

«Дизайн — це ітеративний процес. Завжди є місце для вдосконалення», — Дж. Гарретт «Елементи користувацького досвіду: дизайн, орієнтований на користувача для вебу та не тільки»

Зручний та інтуїтивний дизайн допомагає користувачам легко взаємодіяти з мобільним застосунком, мінімізує можливість помилок і створює приємний досвід використання. Використання принципів простоти, логіки, передбачуваності та доступності робить інтерфейс не лише красивим, а й функціональним, що напряду впливає на популярність і успіх застосунку.



Запитання до розділу 1

1. Як змінилися мобільні операційні системи з моменту появи перших пристроїв?
2. Чому BlackBerry втратила популярність після появи IOS та Android?
3. Чим перший iPhone відрізнявся від популярних на той час смартфонів?
4. Чому відкритість Android стала її ключовою перевагою на ринку?
5. Яка різниця між Human Interface Guidelines від Apple та Material Design від Google?
6. Як впливають сучасні мобільні операційні системи на дизайн застосунків?
7. Що таке UX/UI дизайн і чому це важливо для мобільних застосунків?
8. Які сучасні тренди впливають на мобільний дизайн?
9. Як психологія користувачів впливає на проектування мобільних інтерфейсів?
10. Які основні етапи створення мобільного дизайну від ідеї до випуску застосунку?

Завдання для самостійної роботи

Аналіз трендів: потрібно обрати будь-який мобільний застосунок (Instagram, TikTok, або інший на вибір) та проаналізувати його UX/UI, визначити тренди, використані в дизайні, та описати, як вони впливають на зручність використання застосунку.



РОЗДІЛ 2. СТРУКТУРА ТА КОМПОНЕНТИ ДИЗАЙНУ МОБІЛЬНИХ ЗАСТОСУНКІВ ДЛЯ IOS/ANDROID

2.1. Основи дизайну мобільних інтерфейсів

Відмінності між платформами IOS та Android

IOS та Android — дві найпопулярніші мобільні платформи, і хоча вони виконують однакову функцію — забезпечують роботу мобільних пристроїв — між ними є суттєві відмінності. Ці відмінності впливають не лише на користувацький досвід, а й на процес розробки та дизайну застосунків.

Перше, що їх відрізняє — це візуальний стиль і підхід до дизайну. Apple використовує Human Interface Guidelines, які спрямовані на мінімалізм, чистоту інтерфейсу та використання великої кількості білого простору. Кнопки, іконки та шрифти у IOS більш округлі, легкі та естетично стримані. Android, натомість, базується на Material Design, де ключову роль відіграють тіні, шари та анімації, що роблять інтерфейс більш реалістичним і динамічним.



Ще одна важлива відмінність — навігація. На IOS користувачі переважно взаємодіють через нижню панель навігації та свайпи. Наприклад, жест «назад» на iPhone працює проведенням від лівого краю екрана. В Android існує кнопка «Назад», яка є частиною системної навігації, а в деяких версіях — свайп-навігація. Це впливає на розташування елементів у застосунках: те, що логічно для користувача IOS, може виглядати незвично для власника Android-пристрою.

Щодо адаптації під різні пристрої, тут Android має значну перевагу в різноманітності, але й викликає більше складнощів у розробників. Існує величезна кількість Android-смартфонів із різними розмірами екранів, співвідношенням сторін і технічними характеристиками, що потребує додаткових зусиль для забезпечення коректного відображення інтерфейсу на всіх моделях.

Ще один цікавий момент — типографіка та кнопки. У IOS шрифтом за замовчуванням є San Francisco, а в Android — Roboto. Розміри кнопок та інтерактивних елементів у Android традиційно більші, що частково пов'язано з більшою кількістю пристроїв із великими екранами.

Також варто згадати про оновлення та підтримку старих версій. Apple активно змушує користувачів переходити на нові версії IOS, тому більшість пристроїв оновлюються швидко. У світі Android все складніше: через велику кількість виробників деякі пристрої можуть отримувати оновлення із затримкою або взагалі залишатися на старих версіях. Це впливає на розробку застосунків, оскільки вони повинні підтримувати одразу кілька поколінь операційної системи.

Щодо монетизації, вважається, що користувачі IOS частіше витрачають гроші на платні застосунки та покупки всередині застосунків, тоді як власники Android частіше використовують безкоштовні застосунки з рекламою.

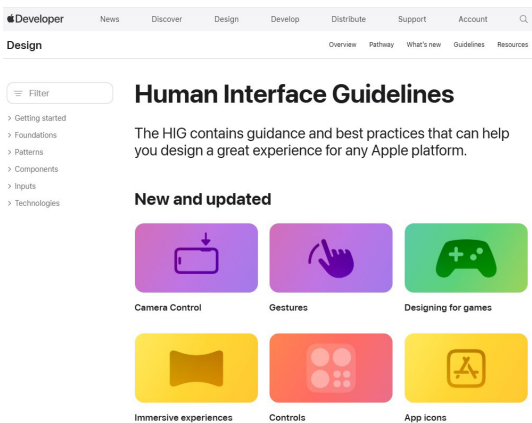


IOS і Android мають принципово різний підхід до дизайну, взаємодії з користувачем та підтримки пристроїв. Дизайнери та розробники повинні враховувати ці відмінності, аби створювати зручні та зрозумілі застосунки для кожної платформи. Вибір між ними залежить від цільової аудиторії, функціональності застосунку та бізнес-стратегії.

Гайдлайни та рекомендації Apple Human Interface Guidelines і Google Material Design

Якщо ви коли-небудь працювали з мобільним дизайном або просто порівнювали застосунки на iPhone і Android, то, напевно, помічали, що вони виглядають і працюють по-різному. Це не випадково, адже кожна платформа має свої гайдлайни дизайну — набір рекомендацій, які визначають, як має виглядати та працювати застосунок. Apple дотримується Human Interface Guidelines (HIG) (<https://developer.apple.com/design/human-interface-guidelines>), а Google розробив Material Design (<https://m3.material.io>). Обидва підходи мають спільну мету — зробити інтерфейси зручними та зрозумілими, але їх реалізація відрізняється.

Apple Human Interface Guidelines: мінімалізм і чіткість



Apple прагне до простоти та елегантності. Human Interface Guidelines (HIG) фокусується на чистому, лаконічному дизайні, де кожен елемент має своє місце й не відволікає користувача від основної функції. Важливий акцент робиться на контент — текст, зображення та головна дія



мають бути в центрі уваги, а інші елементи повинні залишатися на задньому плані.

Основні принципи HIG:

Глибина та прозорість — Apple активно використовує розмиття та тіні, щоб створити ефект багатoshаровості інтерфейсу (наприклад, панель сповіщень із напівпрозорим фоном).

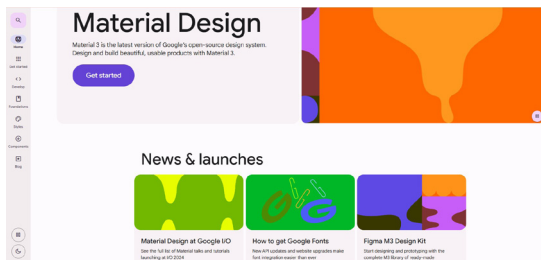
Жести та анімації — свайпи та плавні переходи повинні виглядати природно й нагадувати реальні фізичні рухи.

Чіткі розміри та відступи — у iOS є чітка система сітки та відстаней між елементами, що допомагає зберігати гармонію в дизайні.

Текст і типографіка — шрифт San Francisco використовується за замовчуванням і адаптується до розміру екрана та стилю застосунку.

Однією з особливостей iOS-дизайну є нижня навігаційна панель (tab bar), яка використовується для основних розділів застосунку, а також кнопка «Назад» у вигляді жесту (проведення від лівого краю екрана).

Google Material Design: структура і динаміка



На відміну від Apple, Google робить ставку на структурність і інтерактивність. Material Design побудований за принципом реальних матеріалів — віртуальні елементи інтерфейсу імітують поведінку паперу,

який може підніматися, змінювати форму, накладатися на інші шари.

Основні принципи Material Design:

Реалістична глибина — елементи інтерфейсу мають чіткі тіні та зміну рівнів, що створює ефект багатoshаровості.



Живі кольори та акценти — Google заохочує використовувати насичені кольори та контрастні елементи, що допомагає створювати виразний візуальний стиль.

Анімація та відповідь на дії — кнопки та картки мають «відгук» на натискання, розширюючись або змінюючи колір, що робить взаємодію зрозумілішою.

Гнучкість та адаптивність — Material Design розроблений для роботи на різних пристроях, тому дизайн повинен бути масштабованим і адаптуватися до будь-якого екрану.

В Android навігація часто реалізується через бокове меню (hamburger menu), а кнопка «Назад» є системною і завжди розташовується в нижній частині екрана.

Основні відмінності між HIG та Material Design

Платформа	Головні риси	Навігація	Анімації	Структура	Шрифти
iOS (HIG)	Мінімалізм, акцент на контенті	Tab bar, жести	Природні, плавні	Чистий дизайн, без зайвих деталей	San Francisco
Android (Material Design)	Структурність, яскраві кольори	Бокове меню, системна кнопка «Назад»	Динамічні, з ефектами натискання	Використання тіней і рівнів	Roboto

Якщо ви створюєте застосунок для iPhone, слід дотримуватися Human Interface Guidelines, щоб користувачі отримали звичний досвід. Якщо ж розробка ведеться для Android, важливо дотримуватися Material Design, аби елементи інтерфейсу відповідали стилю платформи.

Якщо застосунок розробляється для обох платформ одночасно, варто адаптувати дизайн для кожної з них. Це не означає, що потрібно створювати два окремих інтерфейси, але важливо враховувати особливості взаємодії, навігації та стилю, щоб користувачі IOS та Android почувалися комфортно.

Гайдлайни Apple та Google — це не просто рекомендації, а основа дизайну мобільних застосунків, яка допомагає створювати зручний та зрозумілий інтерфейс. HIG підкреслює



простоту, мінімалізм і плавність взаємодії, тоді як Material Design надає перевагу структурі, яскравості та динаміці. Дизайнерам та розробникам важливо розуміти ці відмінності, щоб створювати якісні застосунки, які відповідають очікуванням користувачів кожної з платформ.

2.2. Навігація та архітектура інтерфейсу

Основні патерни навігації (tab bar/bottom navigation, navigation drawer, tabs)

Навігація — це основа зручного мобільного застосунку. Від того, як організовані кнопки, меню та переходи між екранами, залежить швидкість і комфорт користувача під час роботи із застосунком. Існує кілька основних патернів навігації, які використовуються в мобільних інтерфейсах. Найпопулярніші з них — tab bar/bottom navigation (нижня панель вкладок), navigation drawer (бічне меню) і tabs (вкладки). Кожен із цих підходів має свої переваги та застосовується залежно від особливостей застосунку.

Tab bar — це один із найпоширеніших способів навігації в IOS-застосунках. Він розташований у нижній частині екрана і містить кілька основних розділів застосунку, які можна перемикати одним натисканням. Вкладки завжди знаходяться в полі зору користувача, що спрощує перемикання між основними секціями.



Рис. 9. Tab bar (нижня панель вкладок) у гайдлайні HIG



Використовується фіксована кількість вкладок (зазвичай 3–5), що зменшує ризик інформаційного перевантаження. Кожна вкладка має власний значок та підпис за потреби, що допомагає інтуїтивно зрозуміти її функцію. Панель вкладок в IOS не містить анімаційних ефектів переходу.

Tab bar зустрічається практично у всіх застосунках, де є кілька ключових розділів, наприклад, у Instagram (Головна, Пошук, Створити пост, Reels, Профіль) або The New York Times (Home, Play, You).

Bottom navigation — це варіант tab bar, який прописаний у Material Design (Android). Він також може містити кнопку головної дії (FAB), якої не існує в гайдлайнах IOS.

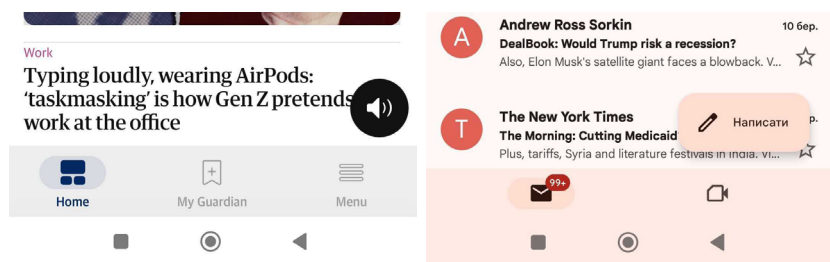


Рис. 10–11. Нижня навігація з кнопкою головної дії (FAB) у Guardian та Gmail

Bottom navigation дає змогу використовувати до 5 основних вкладок. Елементи можуть мати динамічну поведінку, наприклад, анімації при перемиканні. Використовується в застосунках, де важливо швидко перемикатися між основними розділами без прихованих меню.

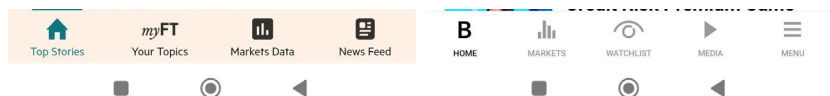


Рис. 12–13. Bottom navigation у Financial Times та Bloomberg



Tab bar і bottom navigation — ідеальні для застосунків із 3–5 основними вкладками нижнього меню, які користувачі часто перемикають (соцмережі, музичні сервіси, медіа).

Navigation drawer — це приховане бічне меню, яке відкривається свайпом або натисканням на іконку «гамбургера» (три горизонтальні лінії у верхньому куті екрана). Якщо застосунок містить багато розділів, і неможливо розмістити їх усі в нижній панелі.

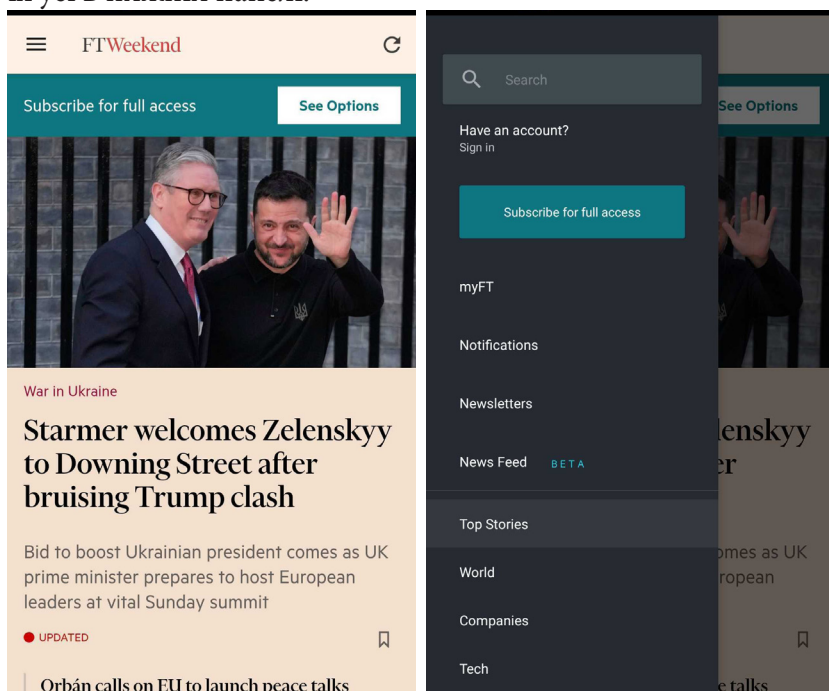


Рис. 14–15. Бічне меню із гамбургер-іконкою у Financial Times

Приховане меню варто використовувати, коли потрібна навігація до менш важливих сторінок, які не повинні займати постійне місце на екрані, а також у випадках, коли важливо залишати більше простору для основного контенту.



Прикладом такого типу навігації є Gmail. Натискаючи на «гамбургер»-меню або свайпаючи з лівої сторони — відкриваються категорії листів (на місці, додати статус, усі скриньки, вхідні, із зірочкою тощо), у застосунку Fianancial Times (myFT, Notifications, Newsletters, Top Stories, World, Companies, Tech, Markets, Climate тощо), а також у багатьох корпоративних та бізнес-застосунках, де є багато підрозділів.

Головний мінус Navigation Drawer — він прихований, і користувачам потрібно зробити додаткову дію, щоб отримати доступ до меню. Це робить навігацію менш швидкою у порівнянні з tab bar чи bottom navigation.

Navigation drawer — підходить для складних застосунків із великою кількістю розділів (поштові клієнти, карти, бізнес-застосунки). Якщо потрібно, можна поєднувати bottom navigation та navigation drawer, як це робить Fianancial Times: головні вкладки внизу (Top Stories, Your Topics, Markets Data, News Feed), а додаткові налаштування у бічному меню.

Таби (tabs) також є елементами навігації. Вони використовуються для швидкого перемикання між різними розділами в межах одного екрана або сторінки. Вкладки (tabs) працюють на одному рівні навігації. Вони не ведуть на інші сторінки чи екрани, а просто перемикають контент на поточному рівні. Вкладки бувають фіксовані (Fixed Tabs), коли всі вкладки видно одночасно чи прокручувані (Scrollable Tabs) — вкладок багато, і вони прокручуються горизонтально.

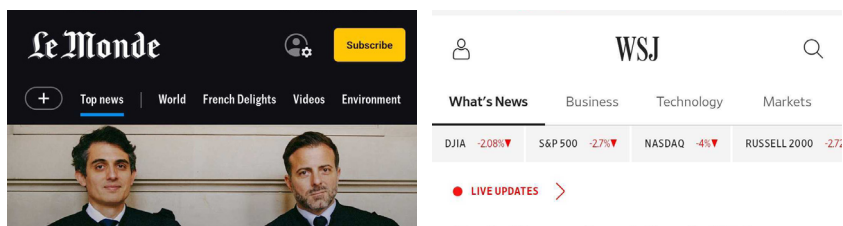


Рис. 16–17. Вкладки у «Le Monde» та «Wall Street Journal»

Навігація — це критично важливий елемент UX, оскільки саме вона визначає, наскільки швидко та зручно користувачі будуть переміщуватися між розділами застосунку. Вибір між *tab bar*, *navigation drawer* і *bottom navigation* залежить від кількості вкладок, важливості розділів і платформи, під яку розробляється продукт. Найкраще рішення — спростувати взаємодію та дотримуватися стандартів iOS і Android, аби користувачам було зручно з першої секунди використання застосунку.

Ієрархія екранів та логіка переходів

Коли ми відкриваємо мобільний застосунок, то рідко замислюємося про те, як саме організовані екрани та чому певні кнопки ведуть нас у потрібне місце. Але насправді за цим стоїть добре продумана система ієрархії та логіка переходів, яка визначає, як користувач пересувається застосунком. Від її правильного налаштування залежить зручність використання та швидкість досягнення потрібної мети.

Ієрархія екранів — це структура застосунку, яка визначає, які сторінки є основними, які допоміжними, а які — вторинними. Це свого роду «карта», за якою користувач орієнтується в інтерфейсі.

У більшості мобільних застосунків можна виокремити три рівні ієрархії:

Головний екран (*Top-Level Screen*) — це основна сторінка, яка відкривається після запуску застосунку. Наприклад, у соціальних мережах — це стрічка новин, у маркетплейсах — список товарів, у банківських застосунках — баланс рахунку.

Рівень другорядних розділів — це основні функціональні розділи, до яких можна перейти з головного екрану (наприклад, «Пошук», «Профіль», «Повідомлення»).

Детальні екрани та додаткові налаштування — це екрани, які відкриваються всередині певних розділів (напри-



клад, перегляд конкретного товару, картки користувача або сторінка редагування профілю).

Добре спроектована ієрархія дає користувачеві інтуїтивно зрозуміти, де він перебуває в застосунку і як повернутися назад без зайвих зусиль. Логіка переходів — це спосіб, яким користувач потрапляє з одного екрану на інший. Існує кілька основних підходів до навігації, які залежать від типу застосунку та його функціоналу.

Лінійна навігація — коли користувач проходить послідовний шлях, наприклад, у формах реєстрації або кроках оформлення замовлення. Це схоже на послідовність слайдів у презентації — користувач рухається вперед і назад без можливості «перескочити» деякі етапи.

Ієрархічна (деревоподібна) структура — користувач починає з головного екрану, переходить у категорії, потім у підкатегорії, а звідти — до конкретного елемента (наприклад, маркетплейси: «Головна» → «Товари» → «Категорія» → «Картка товару»).

Модальна навігація (Pop-up/Overlay) — коли користувач бачить додатковий екран або спливаюче вікно поверх поточного інтерфейсу. Це часто використовується для швидких дій, таких як підтвердження видалення файлу або перегляд детальної інформації без повного виходу з основного екрану.

Глобальна навігація — коли користувач може швидко переключатися між ключовими розділами застосунку (наприклад, через tab bar або bottom navigation).

Жестова навігація — використання свайпів, натискань або інших жестів для переміщення між екранами (наприклад, свайп вліво для повернення назад у iOS або жестова система у Android).

Погано спроектована структура призводить до того, що користувач губиться у застосунку, не розуміє, як повернутися назад або знайти потрібну функцію. Це може викликати



роздратування, а в найгіршому випадку — змусити людину взагалі відмовитися від використання застосунку.

Ось кілька ключових принципів, які допомагають зробити навігацію зручною:

1. Мінімум кроків до цільової дії — користувач повинен дійти до потрібної функції за 2–3 переходи, а не заглиблюватися у безкінечні меню.

2. Зрозумілість шляхів повернення — кнопка «Назад» або жестова навігація повинні працювати логічно, щоб користувач не губився у структурі застосунку.

3. Візуальні підказки — активні вкладки, хлібні крихти або заголовки сторінок повинні чітко вказувати, де саме перебуває користувач.

4. Послідовність — навігація повинна бути однаковою у всьому застосунку: якщо в одному розділі діє свайп «назад», він повинен працювати так само і в інших.

Ієрархія екранів і правильна логіка переходів — це основа вдалого UX. Вона дає змогу зробити застосунок інтуїтивним, зрозумілим та зручним для користувачів. Головне — продумати структуру так, аби люди могли легко переміщатися між розділами, не губилися та швидко знаходили необхідні функції. Добре продуманий дизайн — це коли користувачеві не потрібно замислюватися, як знайти потрібний розділ чи повернутися назад.

UX-підхід до організації структури застосунку

Структура мобільного застосунку — це, по суті, його скелет, який визначає, як користувач буде переміщуватися між екранами, знаходити потрібні функції та виконувати ключові дії. Якщо структура добре продумана, користувач не замислюється, як щось знайти — він просто інтуїтивно це робить. Саме тому UX-підхід до організації структури застосунку є критично важливим для створення зручного продукту.



Усе починається з дослідження. Перш ніж проектувати екрани, потрібно розуміти, хто буде користуватися застосунком, які завдання він допомагає вирішувати та як люди очікують взаємодіяти з ним. Для цього UX-дизайнери проводять опитування, аналізують поведінку користувачів у подібних застосунках і визначають типові сценарії використання.

На основі цього формується інформаційна архітектура — своєрідна карта всіх екранів і розділів застосунку. Вона допомагає зрозуміти, як організувати контент, які розділи зробити основними, а які — другорядними.

Уявімо, що ми створюємо мобільний застосунок для бронювання гри в гольф. UX-дизайнер має спроектувати структуру так, щоб користувач якомога швидше міг забронювати/оплатити гру.

Головна сторінка — це клуб, в якому гравець весь час робить бронювання гри в гольф, відкритий на вкладці забронювати та перелік гольф полів із часовими слотами.

Після вибору гольф поля — вибір дати та часу гри та кількості гравців із подальшим підтвердженням замовлення. Меню Бронювання показує майбутні та минулі ігри. Процес бронювання містить мінімальну кількість кроків (гольф поле, день/час, кількість гравців, підтвердження/оплата).

Якщо структура добре спроектована, користувачеві не доведеться довго шукати меню або витратити час на зайві переходи — усе буде зрозуміло з першого використання.

UX-підхід до організації структури мобільного застосунку базується на логіці, простоті та ефективності. Застосунок має бути зрозумілим без пояснень, а користувач — легко знаходити потрібні функції. Гарний UX — це не тільки стильний дизайн, а й правильно продумана структура, яка робить взаємодію швидкою, зручною та приємною.



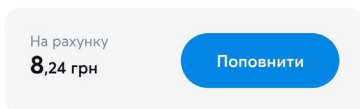
2.3. Основні компоненти UI

Кнопки, поля введення, перемикачі, чекбокси

Коли ми користуємося мобільними застосунками, більшість дій виконуємо за допомогою інтерактивних елементів — кнопок, полів введення, перемикачів і чекбоксів. Саме ці компоненти є основою взаємодії з інтерфейсом, і від їх зручності залежить те, наскільки швидко та комфортно користувач зможе виконати свої завдання.

Кнопки

Кнопки — це один із найважливіших елементів взаємодії користувача з інтерфейсом. Вони повинні бути зручними, зрозумілими та легко помітними. Що робить кнопку справді ефективною?



Помітність і зрозумілість. Користувач має з першого погляду розуміти, яку дію виконає натискання. Наприклад: кнопка «Оплатити» має бути яскравою та розташованою поруч із сумою оплати. Натомість кнопка «Скасувати» або «Повернутися» не повинна бути такого ж кольору, як основна дія — краще зробити її менш яскравою (наприклад, сірою або білою з рамкою).

Невдалий приклад, якщо кнопки «Видалити» та «Зберегти» мають однаковий вигляд, користувач може випадково натиснути не на ту дію, що призведе до помилки. Застосунок фітнес-трекера «RunKeeper» також має цей недолік, тому легко сплутати кнопки «Стоп» і «Пауза», адже вони мають яскраві кольори, а основною дією є кнопка «Стоп» червоного кольору. Якщо користувач їде на велосипеді і зупинився на світлофорі, то може випадково завершити поїздку, натиснувши на стоп замість паузи.

Натомість вдалим прикладом є подібний застосунок «Map my run», який пропонує провести по кнопці для за-



вершення трекінгу, що унеможливорює помилкове її використання. Також кнопка має сірий колір.

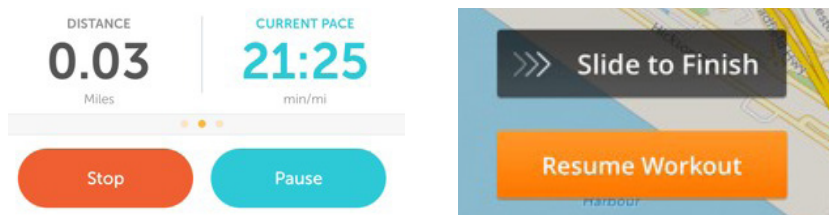


Рис. 18–19. Кнопки застосунку «RunKeeper» та застосунку «Map my run»

Зручний розмір. Кнопки повинні бути достатньо великими, аби їх легко можна було натиснути пальцем. Мінімальна висота — 44–48 рх. На мобільних пристроях великі кнопки, наприклад, «Замовити» допомагають швидко ухвалювати рішення без зайвих зусиль.

Контрастність. Колір кнопки має відповідати її важливості:

1) головна дія — яскравий колір (синій, зелений, червоний), наприклад, кнопка «Підтвердити покупку» на екрані інтернет-магазину;

2) другорядні дії — менш виразні кольори (сірий, білий), як-от кнопка «Скасувати» або «Пропустити».

Невдалою є ситуація, коли всі кнопки на екрані одного кольору, і користувачеві важко визначити, яка з них основна.

Ефект натискання. Після кліку користувач має отримати зворотний зв'язок — зміну кольору, легку анімацію або завантажувальний індикатор. Наприклад, після натискання кнопки «Купити» з'являється коротка анімація завантаження, яка показує, що замовлення обробляється.

Поля введення та їх зручність

Поля введення використовуються для введення інформації: реєстрації, пошуку, коментарів тощо. Щоб вони були



Your email address *

Continue

інтуїтивно зрозумілими, потрібно дотримуватись кількох правил:

Чіткий контур і зрозумілий підпис. Вдалим прикладом

буде замість простого «Email» додати підказку: «Введіть ваш email, наприклад, name@gmail.com». Якщо поле виглядає як декоративний елемент і не має підпису, користувач може не зрозуміти, що саме потрібно ввести.

Підказки та приклади. Додатковий текст допомагає користувачеві правильно ввести дані. Наприклад, під полем «Номер телефону» можна додати підказку «+380 XX XXX XX XX», щоб користувач одразу розумів формат введення.

Автозаповнення та підказки. Якщо система може запам'ятати дані або підказати варіанти, це значно полегшує життя користувачам. Наприклад, користувач має кілька електронних пошт і вийшов з акаунту, варто залишити в полі останню використану пошту, аби полегшити його авторизацію знов. Інший приклад — Google Maps. Під час введення адреси достатньо написати кілька перших літер, і з'явиться перелік можливих варіантів.

Помітні повідомлення про помилки. Якщо користувач ввів неправильний email, під полем має з'явитися червоне повідомлення «Введіть ко-

First name *

Please enter a valid name with at least 2 characters

Last name *

Please enter a valid name with at least 2 characters

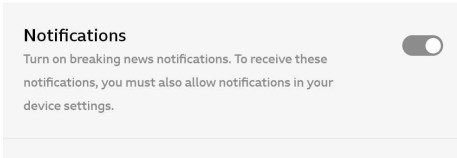
ректну адресу електронної пошти». Ще краще — додати автоматичну перевірку на фронтенді та підказати правильний формат ще до відправки форми.

Невдалим буде варіант, коли після заповнення форми просто з'являється повідомлення «Помилка», без пояснення, що саме пішло не так.



Перемикачі

Перемикачі дають змогу вмикати або вимикати певні налаштування (наприклад, «Отримувати сповіщення»). Вони працюють як світловий вимикач: є два стани — увімкнено/вимкнено. Щоб перемикачі були ефективними:



1) Вони повинні мати зрозумілий стан — різниця між «включено» і «вимкнено» має бути очевидною (зазвичай це зміна кольору, наприклад, синій — увімкнено, сірий — вимкнено).

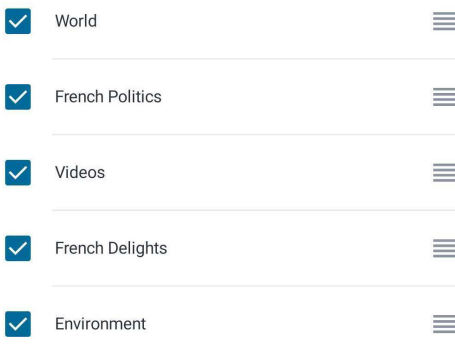
2) Супроводжуватися текстовим описом — поруч із перемикачем має бути чітка назва (наприклад, «Автоматичне оновлення»).

3) Зміни повинні бути миттєвими — перемикач не повинен вимагати натискання кнопки «Зберегти», зміни застосовуються одразу.

Чекбокси

Чекбокси дають змогу користувачеві вибрати одну або зазвичай кілька опцій із переліку (обрати тематику розділів новин, які читач планує відстежувати).

Sections



Щоб чекбокси були ще зрозумілішими, вони повинні мати підпис (наприклад, «Отримувати новини», «Погоджуюся з умовами використання»), достатній розмір, оскільки користувач натискає чекбокси пальцем, вони повинні бути



не менше 24x24 px для зручної взаємодії. Доречним буде групування, якщо чекбоксів багато, щоб користувач міг швидко знайти потрібні опції.

Ці компоненти часто використовуються разом: кнопки можуть підтверджувати вибір чекбоксів, а перемикач може вмикати додаткове поле введення. Важливо, аби усі елементи були візуально узгодженими, розташовувалися логічно та не створювали хаос в інтерфейсі.

Кнопки, поля введення, перемикачі та чекбокси — це основні будівельні блоки будь-якого мобільного застосунку. Від того, наскільки вони інтуїтивні та зручні, залежить, чи буде користувачеві комфортно працювати з інтерфейсом. Чітке оформлення, правильний розмір і логічне розташування допомагають зробити взаємодію простою та приємною, що напряму впливає на успіх продукту.

Меню, списки, картки

Коли ми відкриваємо мобільний застосунок, то бачимо не просто текст і кнопки — інформація організована у вигляді меню, списків та карток. Ці елементи допомагають структурувати контент так, аби він був легко зрозумілим і доступним. Від того, як саме вони оформлені, залежить, наскільки швидко користувач зможе знайти потрібні дані та виконати потрібні дії.

Меню — один із найголовніших інструментів навігації в застосунку. Воно містить основні розділи та допомагає користувачеві швидко перемикатися між ними.

Основні види меню:

Головне меню (main menu) — зазвичай розташовується в нижній частині екрану (tab bar/bottom navigation) або відкривається збоку (navigation drawer).

Контекстне меню (context menu) — з'являється після довгого натискання на елемент або натискання на три крап-



ки, допомагаючи виконати додаткові дії (редагування, видалення тощо).

Спадне меню (dropdown menu) — використовується для вибору однієї з кількох опцій, наприклад, у фільтрах або налаштуваннях.

Щоб забезпечити зручне меню — не варто перевантажувати його великою кількістю пунктів. Головне меню має містити лише ключові розділи, а менш важливі — ховатися у додаткових налаштуваннях. Користувач повинен одразу розуміти, що означає кожен пункт. Наприклад, іконка профілю повинна вести саме до профілю, а не до налаштувань. Меню повинно бути легко досяжним однією рукою (особливо у сучасних смартфонах з великими екранами).

Списки

Списки — один із найефективніших способів подачі великої кількості інформації у мобільному застосунку. Вони присутні у месенджерах (перелік чатів), у маркетплейсах (товари), у банківських застосунках (транзакції), у медіа тощо.

Вони поділяються на:

- прості списки — текстові або графічні елементи, розташовані вертикально (наприклад, список контактів);
- списки з іконками — містять не лише текст, а й графічні позначки або аватарки (наприклад, у Telegram контакти мають фото);
- списки з прев'ю — коли кожен елемент містить зображення, заголовки, підзаголовки (наприклад, список статей або картки товарів)/

Щоб зробити списки зручними, потрібно дотримуватися таких рекомендацій:

- забезпечити достатню відстань між елементами, щоб уникнути випадкових натискань. Кожен пункт списку має бути достатньо великим;



- групувати за категоріями — якщо список довгий, його краще розбити на категорії або додати заголовки розділів;
- застосовувати логічне сортування — інформація має бути розташована в порядку, який найбільше відповідає потребам користувачів (наприклад, у списку транзакцій останні платежі мають бути зверху).

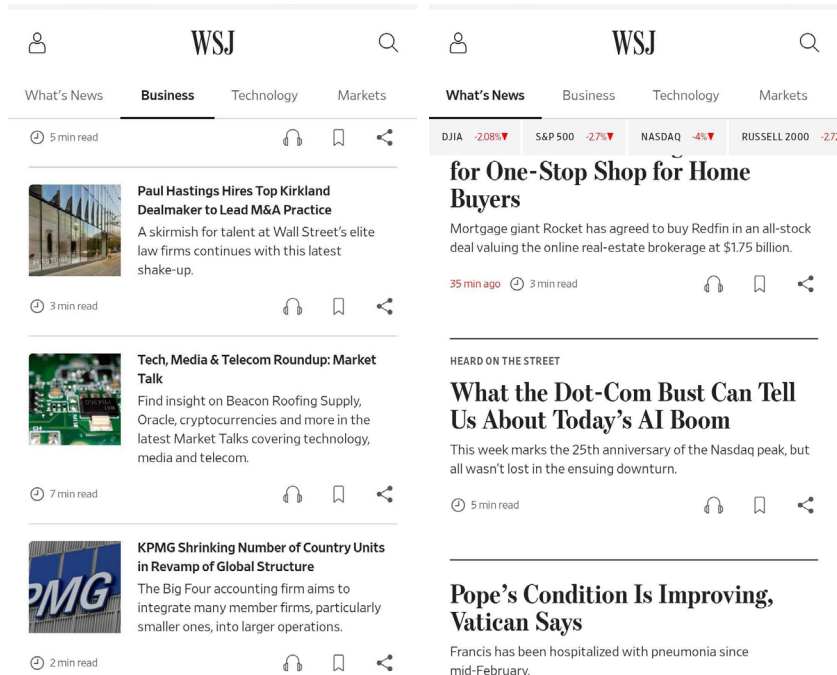


Рис. 20–21. Списки новин «Wall Street Journal»

Картки

Картки — це окремі блоки, які містять певну інформацію (зображення, заголовок, текст, кнопку дії). Вони часто використовуються у стрічках новин, маркетплейсах, рекомендаціях та інших застосунках, де потрібно подати багато даних у візуально привабливому форматі. Вони універсальні та підходять як для списків, так і для окремих елементів (на-

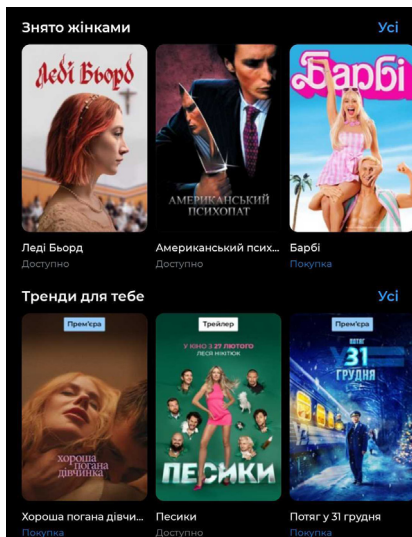


Рис. 22. Фільми у вигляді карток в додатку «Київстар ТВ»

приклад, стаття у блозі, товар у магазині). Користувач одразу бачить, що міститься в картці, завдяки поєднанню тексту та зображення. У картках можна розміщувати кнопки, рейтинги, позначки «Обране» та інші інтерактивні елементи.

Щоб використовувати цей компонент правильно, варто звернути увагу на таке:

1) картка повинна містити лише найважливішу інформацію, не перевантажувати користувача зайвими деталями;

2) текст, зображення та кнопки повинні гармонійно розташовуватися, щоб картка була естетичною та легко сприймалася;

3) якщо картка веде на окремий екран, вона має містити достатньо інформації, аби користувач міг ухвалити рішення ще до відкриття деталей.

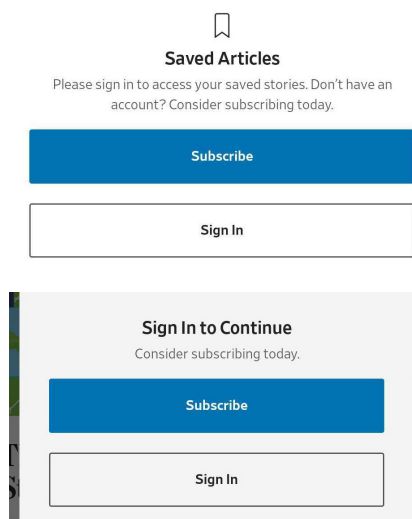
Отже, підсумуємо: меню використовується для навігації та переходу між розділами застосунку, списки — для відображення великої кількості даних у компактному вигляді, а картки — для візуально естетичної подачі контенту з інтерактивними елементами. У багатьох випадках ці елементи працюють разом. Наприклад, у маркетплейсі меню допомагає перемикатися між категоріями, списки відображають товари, а картки показують їх деталі та ціну.

Меню, списки та картки — основні елементи, які допомагають структурувати інформацію у мобільному застосунку.



ку. Головне завдання дизайнера — зробити їх максимально зручними та зрозумілими, щоб користувачі могли швидко знаходити потрібний контент і легко взаємодіяти з інтерфейсом. Простота, логіка та візуальна привабливість — ось три ключові принципи, які допомагають створювати ефективні меню, списки та картки.

Діалогові повідомлення



Ці елементи інтерфейсу допомагають звернути увагу користувача на важливу інформацію або підтвердити певну дію. Вони можуть бути корисними, коли потрібно повідомити про помилку, отримати підтвердження чи надати користувачеві додаткові варіанти вибору. Але якщо їх використовувати неправильно, вони дратуватимуть, відволікатимуть або навіть змусять людину закрити застосунок.

Діалогові повідомлення з'являються поверх основного контенту і блокують взаємодію із застосунком, поки користувач не виконає потрібну дію (наприклад, закриє вікно або підтвердить дію). Вони використовуються, наприклад, у випадках підтвердження важливої дії («Ви впевнені, що хочете видалити файл?»), запиту дозволів (доступ до камери чи геолокації), оновлення чи акції («Новий функціонал доступний! Спробуйте зараз»), під час помилки або попередження («Невірний пароль. Спробуйте ще раз»), як результат успішної дії («Ваш пароль успішно змінено»).



Основні види:

Alert (спливне повідомлення), описане в гайдлайні IOS — наприклад, «Помилка! Невірний пароль». Такі повідомлення з'являються в центрі екрана та потребують натискання кнопки для закриття.

Toast (тимчасове повідомлення) — короткий текст, який зникає сам по собі через кілька секунд (наприклад, «Файл збережено»).

Snackbar (нижнє повідомлення у Material Design) — схоже на toast, але може містити кнопку дії (наприклад, «Повідомлення надіслано — Скасувати»).

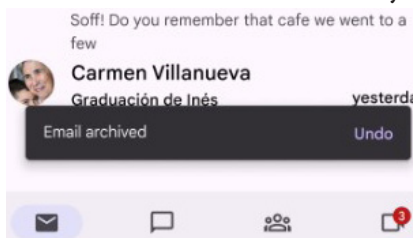


Рис. 23. Зразок снейкбару в Material Design

Варто правильно використовувати діалогові повідомлення: не зловживати ними — користувач не повинен постійно бачити повідомлення після кожної дії; добре візуалізувати — помилки мають бути червоними, успішні дії — зеленими,

інформаційні — синіми; не вимагати зайвих дій — toast-повідомлення гарні тим, що зникають самі, і користувачеві не потрібно натискати «ОК».

Діалогові повідомлення повинні допомагати, а не дратувати. Вони мають бути доречними, інформативними та не перевантажувати інтерфейс. Якщо користувачеві доводиться натискати «ОК» або закривати модальні вікна занадто часто, варто переглянути їхнє використання, щоб не псувати UX.

2.4. Анімації та мікровзаємодії

Роль анімації у покращенні UX

Анімація в мобільних застосунках — це не просто красивий ефект, а важливий інструмент, який робить взаємодію



з інтерфейсом більш природною, зрозумілою та естетично доповненою. Вона допомагає користувачеві орієнтуватися в застосунку, надає зворотний зв'язок на дії та створює відчуття плавності. Без анімації інтерфейс здавався б статичним і менш інтерактивним.

Коли користувач відкриває бічне меню, і воно плавно виїжджає з лівого боку екрана, це створює логічний зв'язок між дією та її результатом. Без анімації зміна могла б здаватися раптовою, що викликало б дезорієнтацію. Найпростіше уявити її у застосунках із гамбургер-меню (не обов'язково натискати на іконку, можна просто провести пальцем із лівого боку до правого). У реальному житті об'єкти не з'являються миттєво і не зникають без причини. Хороша анімація імітує фізику реального світу, роблячи інтерфейс інтуїтивно зрозумілим.

У IOS, коли користувач повертається назад, екран не просто змінюється — він плавно «зсувається» праворуч, створюючи ефект, ніби користувач гортає сторінку.

Анімація допомагає підкреслити ключові дії або помилки. Якщо користувач вводить невірний пароль, поле може злегка «струснутися», ніби кажучи: «Щось не так». Це набагато ефективніше, ніж просто червоний текст із підказкою. У застосунках для банкінгу, таких як Monobank, при спробі авторизації з неправильним паролем поле пароля «трясється», сигналізуючи про помилку.

Користувач повинен отримувати миттєвий візуальний зворотний зв'язок, щоб бути впевненим, що його дія спрацювала. Однак, варто зауважити, що люди звикають до цих дій і сприймають їх як належне. Варто придивитися і зрозуміти, що анімації скрізь, наприклад, у соціальних мережах, коли відкривається непрочитане повідомлення. У Facebook при реагуванні на пост, смайлик спочатку збільшується, також динамічно відтворює емоцію, додаючи приємний ві-



зуальний ефект. Анімація може створювати приємний користувачький досвід, роблячи інтерфейс більш дружнім та живим. У застосунку Duolingo після правильної відповіді анімований персонаж підстрибує від радості, що додає користувачеві мотивації. Ефект скролінгу можуть спостерігати користувачі TikTok або Instagram, де сторінки змінюються «перелистуванням», що створює природне відчуття гортання контенту.

Анімація також використовується під час завантаження застосунку та в переходах між екранами. Особливої уваги заслуговує анімація на стартовому екрані (splash screen). Вона зазвичай з'являється на кілька секунд після запуску застосунку і може охоплювати анімацію логотипу, фонових елементів, тексту або інших графічних компонентів. Якщо застосунок завантажує великий обсяг даних або ініціалізує функції при запуску, анімація може служити як «тимчасова завіса», що дає змогу користувачеві очікувати завершення завантаження, не відчуваючи паузу чи затримку. Анімація на стартовому екрані також допомагає підкреслити бренд, встановлюючи візуальний контакт зі споживачем. Наприклад, плавна поява логотипу компанії підвищить упізнаваність бренду. Такий ефект на стартовому екрані буде більш захопливим, надаючи користувачеві приємне візуальне враження на початку взаємодії із застосунком. Однак, варто пам'ятати, що анімація на стартовому екрані не повинна займати більше 3–5 секунд.

Попри всі переваги, занадто багато анімацій може дратувати. Якщо застосунок змушує чекати через надто довгі ефекти, це погіршує UX. Тому важливо дотримуватися балансу між естетикою та швидкістю взаємодії.

Анімації не тільки покращують загальний вигляд застосунків, але й можуть стати частиною персоналізованого досвіду користувача. Наприклад, в деяких застосунках можна



спостерігати анімаційні ефекти, що змінюються залежно від часу доби, місця перебування користувача або навіть його попередніх взаємодій із застосунком. У мобільному застосунку Spotify можна побачити, як анімації відображають вибір користувача на основі його слухових уподобань або створених плейлистів. Обираючи пісню або під час перемикання між різними типами музичних жанрів, застосунок може створювати анімації, що супроводжують цей перехід, надаючи відчуття безшовної та органічної зміни контенту. Це не лише покращує взаємодію, а й допомагає створити персоналізований досвід, який адаптується до звичок користувача.

Анімація також може підсилювати відчуття нагороди та мотивації. Наприклад, у Memrise, анімації, які з'являються після правильно виконаних завдань, можуть викликати відчуття досягнення та підвищити інтерес до подальших дій. Це створює відчуття прогресу, що мотивує користувача продовжувати навчання.

Анімації можуть також слугувати важливим інструментом для зміцнення брендингу та створення відчуття унікальності застосунку. Замість стандартних або застарілих візуальних елементів, бренди можуть використовувати анімацію, аби зробити свій продукт упізнаванішим і запам'ятовуваним. Наприклад, у застосунках таких як Airbnb чи Uber, анімаційні ефекти допомагають створити особливу атмосферу, підкреслюючи унікальність бренду і його місце на ринку.

Важливим аспектом є врахування доступності анімацій для всіх категорій користувачів. Вони можуть бути недоступними для людей з обмеженими можливостями, наприклад, для осіб із порушеннями зору або тих, хто страждає від епілепсії. Багато застосунків нині враховують ці чинники, даючи змогу користувачам налаштувати відображення анімацій чи повністю відключати їх. Наприклад, система iOS



має вбудовану функцію «Зменшення руху», яка допомагає зменшити або прибрати анімації для людей, що мають проблеми з вестибулярним апаратом або іншими медичними станами.

Один з важливих аспектів, коли мова йде про анімації, це їхня роль у підвищенні ефективності роботи застосунку. Анімація може бути використана для індикації прогресу в обробці даних, завантаженні контенту чи підвантаженні додаткових ресурсів, даючи користувачу зрозуміти, що процес триває, а застосунок не «завис». Наприклад, у застосунках для покупок, таких як Amazon або eBay, анімації на екранах кошика або під час оформлення замовлення можуть слугувати не лише для візуального оформлення, а й для покращення сприйняття швидкості застосунку, зменшуючи сприйняття затримки в часі.

Мікроваємодії — це не лише анімації в переходах між екранами, а й ефекти, які відбуваються при взаємодії з окремими елементами інтерфейсу, такими як кнопки, перемикачі або іконки. Наприклад, у застосунках для організації завдань, таких як Todoist або Microsoft To-Do, при натисканні на завдання або його позначенні як виконаного може бути

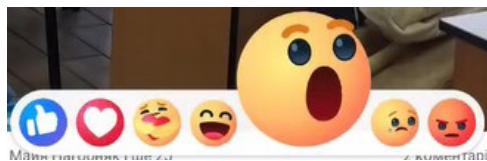


Рис. 24. Анімовані реакції у Facebook рухаються та змінюють розмір

додано анімаційний ефект — наприклад, іконка «галочки» може анімовано з'являтися чи ставати зеленого кольору, додаючи приємну візуальну зворотну реакцію.

Анімація — це не просто ефект, а функціональний інструмент UX, який допомагає користувачам краще розуміти інтерфейс, полегшує навігацію та робить взаємодію із застосунком більш природною. Вона має бути не переванта-



женою, не заважати користувачеві та не уповільнювати процеси, а працювати на покращення досвіду.

Принципи створення плавних переходів

Перехід між екранами або відкриття нового контенту не повинні бути миттєвими та грубими. Різка зміна може дезорієнтувати користувача. Натомість, варто додати плавне затемнення, зсув або масштабування, щоб користувач бачив, що саме змінилося.

У застосунку Instagram при відкритті сторінки профілю контент плавно зсувається вбік, а не з'являється раптово. У реальному житті об'єкти не зупиняються миттєво — вони мають інерцію. Тому при створенні переходів важливо враховувати прискорення та уповільнення анімацій. Якщо користувач натискає на картку, яка відкриває повноекранний перегляд, важливо показати логічний зв'язок між цими станами. Картка може плавно розширюватися, а не просто зникати, відкриваючи новий екран. У Google Photos, коли користувач натискає на мініатюру зображення, вона плавно розгортається на весь екран, а не просто перемикається на інший вигляд.

Навіть якщо анімація робить перехід красивим, вона не повинна затримувати користувача. Ідеальна тривалість анімації — 200–300 мс. Це достатньо швидко, щоб не здавалося повільним, але й достатньо плавно, щоб не виглядало різко.

У material design більшість анімацій побудовані так, що користувач не помічає затримки, але все одно бачить плавний ефект.

При переміщенні між екранами користувач повинен розуміти, де він знаходиться. Якщо анімація допомагає йому орієнтуватися (наприклад, екран «з'їжджає» вбік, а не просто змінюється), це робить UX зручнішим. У YouTube, коли відкривається відео, воно плавно розширюється, а при згор-



танні — зменшується в правий нижній кут, показуючи, що його все ще можна переглядати.

Плавні переходи найчастіше використовуються:

- як перемикання між вкладками — наприклад, в Instagram між «Головною», «Пошуком» і «Профілем» немає різкого обриву, а екрани змінюються плавно;

- в результаті відкриття меню та модальних вікон — у Google Maps панель із деталями місця виїжджає плавно знизу, а не з'являється раптово;

- під час підвантаження контенту — у Twitter нові твіти не просто з'являються, а поступово розсувають вже наявний список;

- при гортання списків — у TikTok при зміні відео не відбувається миттєвої різкої зміни, а є легкий ефект перетікання.

Плавні переходи — це ключовий елемент UX, який допомагає зробити взаємодію із застосунком логічною, природною і приємною. Вони створюють відчуття безперервності, покращують навігацію та допомагають користувачеві не втрачати контекст. Головне — дотримуватися балансу, щоб анімація була швидкою, але не різкою, ефектною, але не перевантаженою.

2.5. Колір, типографіка та іконографія

Вибір кольорової схеми відповідно до UX/UI

Кольори відіграють вирішальну роль у дизайні мобільного застосунку, впливаючи на емоції, зручність використання та загальне сприйняття продукту. Вибір правильного кольорового рішення — це не просто питання естетики, а й стратегічний крок, який може вплинути на взаємодію користувачів із застосунком, їхню залученість і навіть рівень конверсії. Розуміння психології кольору допомагає створити мобільний застосунок, що викликає правильні почуття у користувачів і відповідає його основним цілям.



Добираючи кольорову схему, врахуйте так:

1) Визначення цільової аудиторії — дитячі застосунки можуть бути яскравішими, бізнес-застосунки — стриманішими.

2) Використання не більше 3–4 основних кольорів — за багато відтінків можуть зробити інтерфейс хаотичним.

3) Тестування контрастності — важливо, щоб усі елементи були добре видимими.

4) Дотримання єдиної стилістики — кольори мають бути узгодженими з загальним дизайном.

5) Перевірка сприйняття кольорів людьми з порушеннями зору — для цього можна використати симулятори кольорової сліпоти (наприклад, Stark — <https://www.getstark.co/>).

Брендові кольори повинні бути впізнаваними та асоціюватися із застосунком. Наприклад, Facebook — синій, що символізує довіру та спокій; Instagram — градієнтний перехід від рожевого до жовтого, що створює енергійний образ; WhatsApp — зелений, що передає дружелюбність і спілкування. Важливо підтримувати єдність кольорової гами в логотипі, іконках та інтерфейсі.

Кольори повинні передавати сенс: зелений — успішна дія, червоний — помилка/незворотня дія, синій — довіра. Головні СТА (call to action) мають виділятися та бути контрастними. Згідно з WCAG (Web Content Accessibility

WebAIM web accessibility in mind services articles

ACCESSIBLE COLORS

My text color is #747474 at 18 px and regular weight

My background color is #EEEEEE

My design must be AA compliant

Contrast Checker

Home > Resources > Contrast Checker

Foreground Hex Value #0000FF Color Picker Alpha 1

Background Hex Value #FFFFFF Color Picker

Fails AA Required contrast ratio: 4.5 Your contrast ratio: 4.03

Passes AA if you change background color to #BFBFBF New contrast ratio: 4.52

Passes AA if you change text color to #666666 New contrast ratio: 4.53

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do...

Guidelines), контрастність тексту та фону є важливим критерієм доступності:

- мінімальний контраст для тексту — 4.5:1 (для великих шрифтів — 3:1). Перевірити можна на сайті <https://accessible-colors.com/> або <https://webaim.org/resources/contrastchecker>
- кнопки та інтерактивні елементи мають бути добре видимими на тлі;
- темний текст на світлому тлі читається краще, ніж світлий на темному.

Найкраще працює правило 60–30–10: 60% — основний фон і нейтральні кольори, 30% — вторинні кольори (допоміжні акценти), 10% — головні акцентні кольори (СТА-кнопки, важливі елементи).

Багато застосунків підтримують темну тему, тому важливо адаптувати кольорову схему для обох варіантів, зберігаючи читабельність. Рекомендується використовувати наступні схеми кольорів:

- Темний текст (#000000) на світлому фоні (#FFFFFF) — класичний варіант для читабельності.
- Світло-сірий текст (#757575) для другорядної інформації.
- Основний текст у темному режимі — білий (#FFFFFF) або світло-сірий (#E0E0E0) на темному фоні (#121212).

Основи мобільної типографіки (шрифти, розміри, контраст)

Типографіка — це основа читабельності та естетики в мобільному дизайні. Навіть найкращий контент буде марним, якщо текст важко читати або він занадто дрібний. Варто враховувати, що IOS та Android мають різні характеристики щодо типографіки. Основний шрифт IOS — це San Francisco, тоді як в Android — Roboto. Використовуйте чіткі та прості шрифти — вони мають бути зручними для зчитуван-



ня навіть на маленьких екранах. Найпопулярніші мобільні шрифти: San Francisco, Roboto, Open Sans, Inter, Lato. Бажано не використовувати більше двох шрифтів у застосунку — основний і акцентний.

Для основного тексту андроїду пропонується 14 sp (normal), для заголовку — 20 sp (medium), для кнопок і табів — 14 sp (medium), для тексту у формах — 16 sp (normal), текст помилки — 12 sp (normal), для короткого модального вікна може збільшуватися до 16 sp (normal), у переліку сайдбару — 14 sp (medium), а в простому переліку — 16 sp (normal). Для другорядного тексту може викоритосуватися 14 sp (normal), але світліший за кольором, аби додати йому другорядності.

Розмірні характеристики IOS визначають основним — 17 pt, вторинний текст — 15 pt, третинний текст також 15 pt, але світліший. Для заголовку рекомендується 34 pt (bold) і мінімальним кеглем може бути 10 pt (regular) у tab bar до назв вкладок.

Важливо залишати достатньо простору між рядками (line-height 1.4–1.6), аби текст не здавався зім'ятим. Між заголовками та текстом має бути чіткий візуальний розрив для зручного сприйняття.

Базуючись на офіційних гайдлайнах від Google Material Design та Apple Human Interface Guidelines, а також на загальноприйнятих принципах UX-дизайну, потрібно враховувати такі оптимальні відступи та вирівнювання:

Відступ між заголовком та основним текстом — 8–12 dp.

Відступ між абзацами — 4–8 dp.

Відступ між рядками в абзаці (line-height) — 1.4–1.6.

Поля з боків текстових блоків — 16–24 dp.

Основний текст зазвичай вирівнюється по лівому краю (left-aligned), що полегшує зчитування. Центроване вирівнювання використовується для заголовків або коротких



текстів (наприклад, підписів до зображень). Вирівнювання по ширині (justified) рідко застосовується у мобільних застосунках, оскільки воно може створювати нерівномірні пробіли між словами.

Доцільна мобільна типографіка — це баланс між естетикою, читабельністю та доступністю. Використання правильних шрифтів, розмірів, відступів і контрасту дає змогу створити зручний і зрозумілий інтерфейс. Залишайте достатньо простору між елементами, уникайте надмірного використання кольорів у тексті та підтримуйте динамічне масштабування для кращої доступності.

Використання іконок у мобільних інтерфейсах

Іконки — це візуальні орієнтири, які допомагають користувачам швидко знаходити функції без необхідності читати текст. Вони роблять UX зрозумілішим, інтуїтивним та естетично привабливим.



Рис. 25. Правило 5 секунд (якщо користувач не розуміє іконку впродовж цього часу, її потрібно підписувати), фото — [uxplanet](#)

Користувач повинен інтуїтивно розуміти значення іконки без пояснень. Наприклад, лупа — це пошук, кошик — вміст покупок, сердечко — обране. Іконки повинні мати однаковий стиль (контурні, заповнені, мінімалістичні або детальні).

Використання різних стилів іконок у одному застосунку виглядає хаотично. Мінімальний розмір інтерактивних іконок — 24x24 px (щоб їх легко було натискати). У заголовках і кнопках зазвичай використовуються 32–48 px.

Іконки повинні мати достатній відступ від тексту, щоб не зливатися з ним. Використання кольорів може підсилювати розуміння (наприклад, червона іконка — це попередження



або помилка). Але іконки не повинні бути занадто яскравими, якщо вони виконують допоміжну роль. Невеликі анімації іконок можуть покращити взаємодію (наприклад, іконка завантаження, що крутиться, або лайк, який збільшується при натисканні). У застосунку WhatsApp використовується набір простих і зрозумілих іконок — кнопка камери, смайлики, голосові повідомлення — усе інтуїтивно знайоме.

Отже, кольорова схема повинна не лише виглядати красиво, а й допомагати користувачеві орієнтуватися у застосунку. Важливо підтримувати контрастність, баланс і акценти. Типографіка визначає, наскільки легко користувачі сприймають текст. Правильний вибір шрифтів, розмірів та міжрядкових інтервалів значно покращує UX. Іконки допомагають швидко сприймати функціонал застосунку, але їх потрібно використовувати зрозуміло, лаконічно та в єдиному стилі.

Гарний UI — це не просто красиві кольори, шрифти та іконки, а продуманий баланс між естетикою та зручністю.

Створення іконки мобільного застосунку

Іконка мобільного застосунку — це перше, що бачить користувач, тому вона має бути унікальною, впізнаваною і відповідати загальному стилю застосунку.

Основні принципи створення іконки:

1. Простота та мінімалізм — іконка повинна бути легко впізнаваною навіть у маленькому розмірі. Уникайте дрібних деталей та складних форм.

2. Збалансований дизайн — використовуйте чіткі геометричні форми, рівномірний розподіл елементів і гармонійне поєднання кольорів.

3. Адаптивність — іконка має добре виглядати у різних розмірах (від 48px до 512px) та підтримувати темний і світлий режими.



4. Відповідність стилю платформи — Apple і Google мають свої рекомендації щодо дизайну іконок:



Рис. 26. Apple App Icon Guidelines (<https://developer.apple.com/design/human-interface-guidelines/app-icons>)

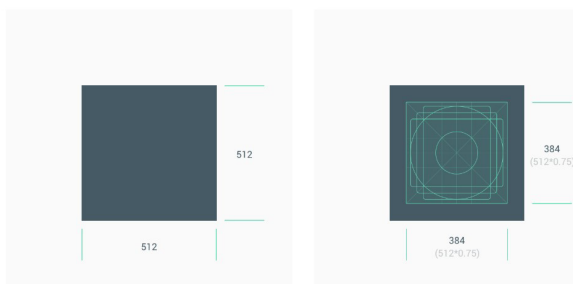


Рис. 27. Google Play Icon Design (<https://developer.android.com/distribute/google-play/resources/icon-design-specifications>)

5. Використання палітри кольорів — контрастні кольори допомагають зробити іконку помітною, але варто уникати надто яскравих поєднань, що можуть драгувати користувачів.

Варто розуміти, що іконка мобільного застосунку — це не те саме, що й логотип і вона може бути динамічною — змінюватися відповідно до трендів чи пір року.

Іконка мобільного застосунку:

- використовується виключно для представлення застосунку на мобільних пристроях;
- має квадратний або круглий формат із заданими розмірами;



- повинна бути простою та легко впізнаваною навіть у маленьких розмірах;
- часто є спрощеним або адаптованим варіантом логотипу.

Іконки для застосунку бувають трьох типів: графічні, текстові, комбіновані.



Рис. 28. Графічні іконки додатків Reuters News та The Independent



Рис. 29. Текстові іконки додатків Le Monde, The Wall Street Journal, та Euronews



Рис. 30. Комбінована іконка додатку Le Figaro

Технічні вимоги:

Формат: PNG або SVG (для масштабованості).

Розміри: Android: 512x512 px (основний), адаптація для різних dpi (48x48, 72x72, 96x96 тощо). IOS: 1024x1024 px (основний), адаптація для різних пристроїв (180x180, 120x120 тощо).

Тло: IOS вимагає повної заповненості іконки, Android використовує адаптивні іконки з можливістю створення шару переднього плану та тла.

Для тестування іконки можна скористатися емуляторами, а також онлайн-сервісами, які допоможуть підготувати як саму іконку, так і різні варіації її форматів:

<https://easyappicon.com/>

<https://makeappicon.com/>

<https://icon.kitchen/>

Високоякісна іконка підвищує привабливість застосунку, сприяє його впізнаваності та створює позитивне перше враження.



Запитання до розділу 2

1. Які основні чинники потрібно враховувати дизайнеру, щоб створити зручний інтерфейс одночасно для IOS і Android?
2. Як навігація в застосунках на IOS відрізняється від Android?
3. У чому переваги та недоліки використання navigation drawer у порівнянні з tab bar?
4. Чому важливо адаптувати дизайн застосунку окремо під IOS і Android?
5. Які існують основні рівні ієрархії екранів у мобільних застосунках?
6. Що таке UX-підхід до структури мобільного застосунку?
7. Які основні компоненти UI використовуються в мобільних застосунках?
8. Наведіть приклади використання анімацій у мобільних застосунках для покращення UX.
9. Які принципи важливо враховувати під час створення плавних переходів між екранами?
10. Якими є рекомендації щодо типографіки та вибору кольорової схеми для мобільних застосунків?

Завдання для самостійної роботи

Редизайн іконки мобільного застосунку. Обрати невдалий варіант іконки та створити альтернативну, враховуючи рекомендації Apple та Google. Описати концепцію і пояснити дизайнерські рішення.





РОЗДІЛ 3. КАРКАС, МАКЕТ, ПРОТОТИП МОБІЛЬНОГО ЗАСТОСУНКУ

3.1. Основи проектування мобільних інтерфейсів Відмінності між каркасом, макетом і прототипом

Каркас, макет і прототип — це різні етапи в розробці мобільного застосунку, кожен із яких має свою роль. Каркас — це базова структура інтерфейсу, без деталей. Макет — візуалізоване оформлення з кольорами, шрифтами і графікою. Прототип — інтерактивна модель, яка дає змогу тестувати функціональність до того, як його реалізують.

Етапи створення інтерфейсу: від ідеї до інтерактивного прототипу

На початковому етапі, коли ідея лише формується, дизайнери створюють *wireframe* — це базовий скелет майбутнього інтерфейсу. Він не містить кольорів, зображень чи складного оформлення, а лише показує, де будуть розташовані основні елементи: кнопки, текстові поля, зони контенту. *wireframe* нагадує чорно-білу схему, яка дає змогу швидко протестувати логіку розташування елементів без витрачання часу на деталізацію.



Коли структура визначена, приходить черга москур — більш деталізованої версії дизайну, де додаються кольори, шрифти, графічні елементи. На цьому етапі макет наближається до того, що користувач побачить у фінальному продукті, але ще не має інтерактивних можливостей. Він допомагає команді узгодити візуальний стиль і зробити необхідні коригування перед створенням реального функціонального прототипу.

Наступний етап — створення прототипу. На відміну від статичного макету, прототип інтерактивний: він імітує поведінку реального застосунку, дає змогу тестувати переходи між екранами, натискати кнопки та перевіряти зручність навігації. Це допомагає ще до розробки оцінити, наскільки інтуїтивно користувач може взаємодіяти із застосунком і чи немає проблем у бізнес-критеріях чи логіці UX.

Інструменти для проектування

Увесь цей процес не був би можливим без сучасних інструментів для проектування, які допомагають створювати всі ці етапи у зручному цифровому середовищі. Найпопулярнішими серед дизайнерів є Figma, Sketch, Penpot, UXPin, Framer та Lunacy.

Figma має велику перевагу завдяки хмарному зберіганню та можливості одночасної роботи кількох дизайнерів у реальному часі, що робить її ідеальним вибором для командної роботи.

Sketch довгий час був стандартом у UI/UX-дизайні, особливо серед користувачів macOS, хоча зараз його популярність поступово знижується.

Penpot — це відкритий (open-source) інструмент, що працює в браузері та підтримує SVG, що робить його чудовою альтернативою для тих, хто шукає безкоштовне рішення без підписки.



UXPin орієнтований на створення інтерактивних прототипів та дизайн-систем, що особливо зручно для команд, які співпрацюють із розробниками.

Framer більше підходить для створення анімаційних та інтерактивних прототипів, дозволяючи швидко демонструвати складні інтерфейси.

Lunacy — це безкоштовний офлайн-редактор від Icons8, який підтримує файли Sketch та підходить для тих, хто не хоче залежати від хмарних сервісів.

Кожен із цих інструментів має свої особливості, тому вибір залежить від конкретних потреб дизайнера чи команди. Робота над інтерфейсом починається з ідеї, яка поступово проходить через стадії структурування (wireframe), деталізації (mockup) і функціонального тестування (prototype), перш ніж перейти до розробки. Інструменти, такі як Figma та інші роблять цей процес гнучким, ефективним і швидким, допомагаючи дизайнерам не просто створювати гарні екрани, а й проектувати зручний, продуманий і логічний користувацький досвід.

3.2. Каркас (wireframe): структура майбутнього застосунку

Що таке каркас і чому він важливий

Перед тим як мобільний застосунок набуває свого фінального вигляду, кожен його екран проходить через стадію wireframe — каркасного макета, який визначає структуру інтерфейсу без графічного оформлення. Wireframe — це візуальне представлення скелету застосунку, що показує, як будуть розташовані кнопки, текстові блоки, зображення та інші елементи. Це один із перших і найважливіших кроків у розробці UI/UX, оскільки саме на цьому етапі формується логіка навігації та розташування компонентів.

Головна цінність wireframe у тому, що він дає змогу швидко перевірити, чи є структура інтуїтивною, чи зручно



користувачеві знаходити потрібну інформацію та виконувати дії. Це допомагає уникнути помилок ще до детального опрацювання дизайну, які могли б ускладнити взаємодію з застосунком.



Рис. 31. Сайт uistencils.com пропонує трафарети для створення каркасів

Види каркасів: низькодеталізований (low-fidelity) та високодеталізований (high-fidelity)

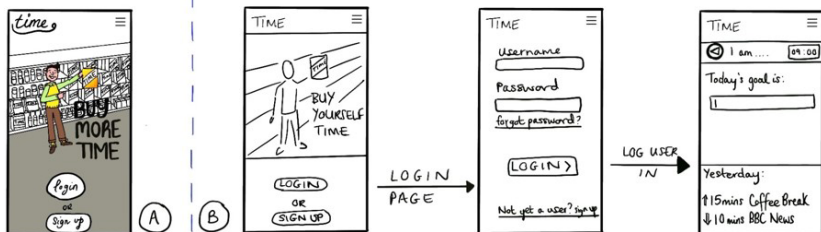


Рис. 32. Приклад каркасу (фото — Anil Muppalla, github.com)

Каркасні макети бувають різних рівнів деталізації. Низькодеталізований каркас — це найпростіші схеми, зазвичай створені в чорно-білому кольорі без дрібних деталей. Вони використовуються на перших етапах, коли потрібно швидко перевірити загальну логіку інтерфейсу. Тут елементи представлені у вигляді простих блоків, а тексти замінені на умовні написи або просто прямокутники, що позначають май-



бутній контент. Відтак високодеталізовані каркаси містять реальні тексти, більш точні розміри елементів та можуть включати базові стилістику й відступи, наближаючи каркас до майбутнього дизайну.

Створення базової структури екранів та взаємодій

Процес створення wireframe зазвичай починається з аналізу основних сценаріїв використання застосунку. Дизайнер визначає, які екрани будуть ключовими, які дії виконуватимуть користувачі та як має працювати навігація. Далі будується базова структура екранів, де позначаються основні зони інтерфейсу: розташування меню, кнопок, форм введення та інших елементів.

Створення wireframe — це ефективний спосіб швидко протестувати концепцію перед тим, як розпочинати детальний дизайн. Оскільки цей етап не потребує складної графіки, він допомагає команді легко вносити зміни, обговорювати варіанти та покращувати структуру ще до того, як почнеться розробка фінального інтерфейсу. Використання wireframes у процесі проектування допомагає розробити різні варіанти дизайну, легко вносити зміни та зробити взаємодію із застосунком логічною й підготувати якісну основу для подальшої роботи.

3.3. Макет (mockup): візуальне оформлення

Роль макету у процесі дизайну

Після того як структура застосунку визначена за допомогою wireframe, настає етап створення макету (mockup) — деталізованого візуального представлення інтерфейсу, яке демонструє, як виглядатиме застосунок у фінальному вигляді. Якщо wireframe є лише схемою, то макет додає всі необхідні деталі: кольори, шрифти, іконки, графічні елементи та загальну стилістику, яка формує враження від продукту.



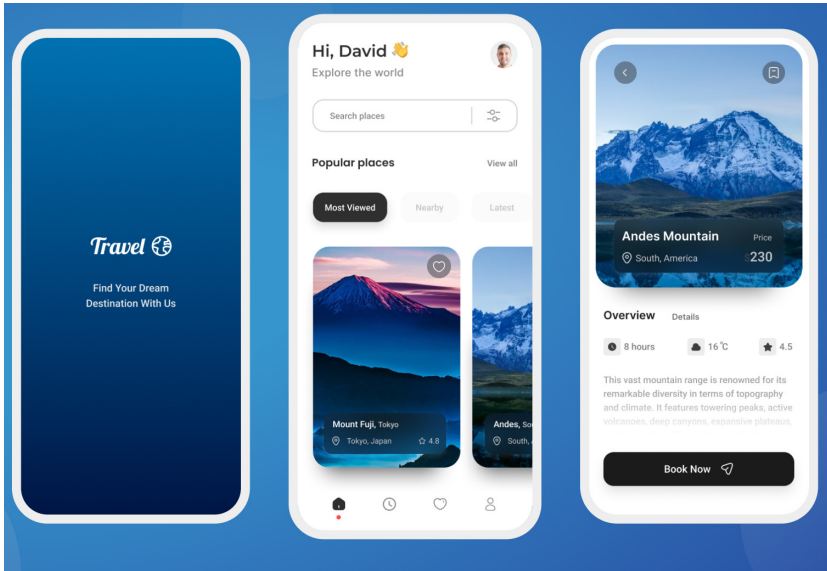


Рис. 33. Шаблони макетів у Figma

Це дуже важливий етап, адже саме тут дизайн починає виглядати як реальний застосунок. Кольорова схема визначає настрій і візуальний стиль: яскраві відтінки можуть створити динамічний ефект, пастельні — викликати відчуття спокою, а темний режим зробить інтерфейс комфортним для нічного використання. Шрифти також відіграють ключову роль: вони мають бути не тільки гармонійними, а й читабельними, особливо на невеликих екранах смартфонів. Додавання іконок допомагає користувачам швидше орієнтуватися в застосунку, а графічні елементи, такі як ілюстрації або тіні, роблять інтерфейс більш сучасним і приємним для взаємодії.

Відповідність гайдлайнам IOS та Android

Важливим аспектом є відповідність гайдлайнам IOS та Android. Обидві платформи мають свої дизайн-системи:



Apple Human Interface Guidelines для IOS та Google Material Design для Android. Вони визначають, якими мають бути розміри кнопок, правила навігації, використання тіней та ефектів взаємодії. Наприклад, у IOS переважає мінімалістичний стиль із плавними переходами та відсутністю явних меж між елементами, тоді як Material Design робить акцент на багаторівневості, використанні тіней і чітких кнопках.

Дотримання цих гайдлайнів не лише полегшує роботу розробникам, а й робить застосунок зрозумілим для користувачів. Люди звикли до певних поведінкових патернів інтерфейсу у своїх операційних системах, тому відхилення від цих стандартів може викликати дискомфорт або складнощі у використанні.

Макет — це фінальна точка перед створенням інтерактивного прототипу. Він дає змогу клієнтам і команді побачити, як виглядатиме продукт, протестувати зручність візуального стилю та узгодити всі деталі перед розробкою. На цьому етапі можна ще внести зміни без значних витрат часу та ресурсів, тож важливо добре опрацювати всі аспекти макету, аби зробити інтерфейс не тільки естетично привабливим, а й максимально функціональним.

Типові помилки в макетах

Проте, навіть досвідчені дизайнери можуть припускатися помилок, які негативно вплинуть на зручність, функціональність та візуальне сприйняття продукту. Важливо обговорювати виконаний макет із продакт менеджером і командою перед початком його втілення. Такі мозкові штурми та командне оцінювання макету допоможе вчасно внести виправлення та удосконалити його.

Однак, деякі помилки все-ж-таки можуть зробити інтерфейс занадто складним, інші — знизити читабельність або зробити взаємодію незручною. Одна з найпоширеніших



помилка — ігнорування адаптивності та різних екранів. Дизайнер може створити естетичний макет, який ідеально виглядає на одному розмірі екрана, але абсолютно не підходить для інших пристроїв. Варто пам'ятати, що смартфони мають різні роздільні здатності, співвідношення сторін і щільність пікселів, тому макет повинен бути гнучким і правильно адаптуватися до всіх можливих варіантів.

Ще одна помилка — недостатній контраст між елементами. Якщо текст має занадто низький контраст із тлом, користувачам буде важко його читати, особливо в умовах яскравого освітлення. Це вкрай важливо для мобільних застосунків, адже користувачі взаємодіють із ними у різних ситуаціях — у транспорті, на вулиці чи в темних приміщеннях. Правильний вибір кольорів та контрастності покращує загальну читабельність і доступність інтерфейсу.

Не менш критичною є помилка перевантаженого дизайну. Якщо у макеті використано занадто багато кольорів, шрифтів, тіней та ефектів, це може створювати нагромадження компонентів та ускладнювати сприйняття. Чим простіший і зрозуміліший інтерфейс, тим легше користувачеві в ньому орієнтуватися. Принцип мінімалістичності стає актуальним саме в мобільному UI/UX, де розмір і компактність мають значення.

Також важливо уникати непослідовності в дизайні. Якщо кнопки на різних екранах матимуть різний стиль, розміри або кольори, це дезорієнтуватиме користувача. Дотримання єдиної дизайн-системи та логічної послідовності допомагає створити гармонійний і передбачуваний досвід взаємодії.

Варто додати, що більшість макетів залишаються на цій стадії, як остаточний і не стають інтерактивними прототипами, що заощаджує час дизайнера.

Створення макету — це баланс між естетикою та функціональністю. Помилки можуть звести нанівець навіть най-



кращі дизайнерські ідеї, тому важливо думати не лише про візуальну привабливість, а й про зручність, логіку та доступність інтерфейсу.

3.4. Прототип (prototype): інтерактивність та тестування Прототип як інструмент перевірки UX

Перш ніж дизайн мобільного застосунку дійде до етапу розробки, він має пройти ретельну перевірку. Це необхідно, аби переконатися, що всі елементи інтерфейсу працюють так, як задумано, а користувачі зможуть легко орієнтуватися в застосунку. Головний інструмент для цього — прототип. Він допомагає краще оцінити новий функціонал, ефектно представити його замовнику і ще на ранніх стадіях побачити, як виглядатиме і функціонуватиме застосунок, і виправити можливі недоліки до того, як почнеться процес розробки.

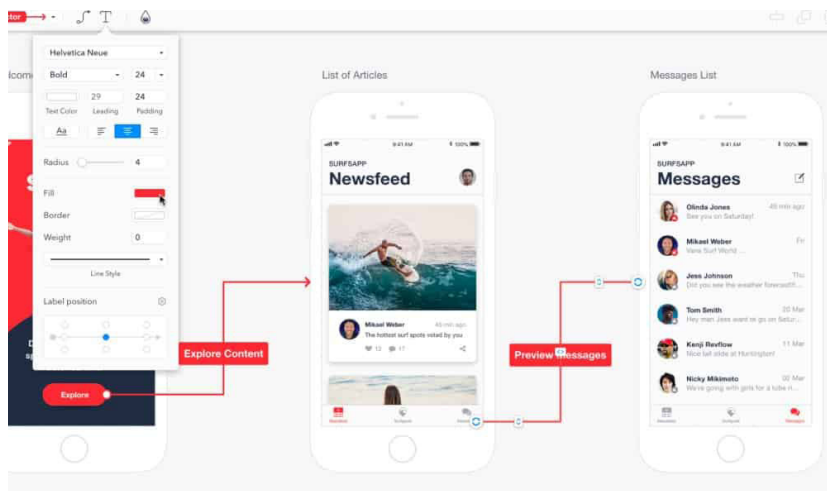


Рис. 34. Зразок прототипу у Proto.io

Прототипування — це створення інтерактивної моделі майбутнього застосунку, яка імітує його поведінку: натис-



кання кнопок, переходи між екранами, роботу меню тощо. На відміну від статичного макету, прототип дає можливість взаємодіяти з інтерфейсом у реальному часі, що допомагає оцінити зручність навігації, логіку сценаріїв і розташування елементів.

Основи створення інтерактивних прототипів

Основою якісного прототипу є інтерактивність. Це означає, що користувач повинен мати можливість виконувати основні дії: відкривати екрани, переходити між розділами, натискати кнопки та взаємодіяти з формами. Для створення інтерактивних прототипів використовуються спеціальні інструменти, такі як Figma, InVision, Axure, Moqups, Proto.io тощо. Вони допомагають не лише проєктувати дизайн, а й додавати анімації, логіку натискань і симуляцію реального використання застосунку.

Однак прототип не просто створюється і передається в розробку — перед цим його потрібно тестувати. Це ключовий момент, оскільки саме тестування допомагає знайти проблеми у зручності використання. Наприклад, якщо користувач не може швидко знайти потрібну кнопку або навігація здається заплутаною, це означає, що дизайн потребує доопрацювання.

Тестування та внесення змін перед розробкою

Тестування прототипу зазвичай проводиться в кілька етапів. Спочатку його перевіряє сама команда дизайнерів чи тих, хто працює над проєктом, аналізуючи, чи всі сценарії працюють правильно. Потім можуть залучатися реальні користувачі, які взаємодіють із прототипом і дають зворотний зв'язок. Важливо спостерігати, які дії викликають труднощі, і на основі цього вносити необхідні зміни.

Прототипування дає змогу зекономити час і ресурси, адже виправлення проблем на ранньому етапі набагато про-



стіше, аніж після розробки. Це не просто етап перед програмуванням, а інструмент для перевірки UX, тестування зручності та внесення покращень ще до того, як застосунок стане повністю функціональним. Якісний інтерактивний прототип — це запорука того, що фінальний продукт буде інтуїтивним, зручним і відповідатиме очікуванням користувачів.

3.5. Юзер-тестування та ітеративне покращення Методи тестування прототипів

Після створення інтерактивного прототипу важливо переконатися, що він справді зручний, логічний і відповідає очікуванням користувачів. Саме для цього проводиться тестування, яке допомагає виявити потенційні проблеми та покращити інтерфейс ще до того, як почнеться розробка. Без цього етапу навіть найкращий дизайн може виявитися незручним у реальному використанні, що призведе до втрат часу і грошей на виправлення вже у фінальному продукті.

Тестування прототипів може відбуватися різними методами. Юзабіліті-тестування передбачає, що реальні користувачі взаємодіють із прототипом і виконують певні завдання. Спостереження за тим, як вони пересуваються між екранами, шукають функції та реагують на інтерфейс, допомагає зрозуміти, чи є проблеми у логіці або навігації. Ще один метод — А/В тестування, коли порівнюються дві різні версії інтерфейсу, щоб визначити, яка з них працює краще. Наприклад, можна тестувати різні варіанти розташування кнопок або колірні акценти, аби дізнатися, що є більш зручним.

Аналіз зворотного зв'язку від користувачів

Отримання зворотного зв'язку від користувачів — це один із найцінніших результатів тестування. Часто дизайнери та розробники мають своє бачення, яке може не збігатися



з тим, як сприймають інтерфейс реальні користувачі. Саме тому варто уважно аналізувати, які труднощі виникли, які дії були незрозумілими, де користувачі губилися або витрачали більше часу, аніж очікувалося. Дуже важливо не просто отримати відгуки, а й правильно їх інтерпретувати, відокремлюючи реальні проблеми від суб'єктивних побажань окремих користувачів.

Після аналізу зібраних даних команда дизайнерів та UX-фахівців вносить необхідні коригування до прототипу. Це може бути зміна структури меню, спрощення деяких екранів, зміна розташування важливих кнопок або додавання підказок. Головна мета — зробити інтерфейс максимально інтуїтивним, щоб користувачам не доводилося витрачати час на роздуми про те, як виконати потрібну дію.

Завершальний етап — це підготовка макетів і прототипів до передачі в розробку. Дизайн-файли структуруються, додаються специфікації для розробників, а також пояснюється логіка взаємодії між елементами. Це важливий момент, адже чітко задокументований та добре протестований прототип значно зменшує ризик непорозумінь між дизайнерами та розробниками і допомагає створити кінцевий продукт, який відповідатиме як задуму, так і очікуванням користувачів.

Тестування та внесення правок — це не просто технічний етап, а процес, що робить застосунок зручнішим, функціональнішим та ефективнішим. Саме тому якісне UX-тестування перед передачею проекту в розробку допомагає уникнути помилок і зробити фінальний продукт таким, що задовольнить потреби користувачів із першого ж використання. Однак, жоден проект не створюється одразу, але розвивається й поліпшується з часом, реагуючи на потреби користувачів, тренди в дизайні чи нові бізнес-стратегії.



Ітеративне покращення мобільного застосунку

Ітеративне покращення є невід’ємним складником процесу UX-дизайну, яка забезпечує поступове наближення прототипу до ідеального стану. Будь-яка компанія, що займається розробкою інформаційних технологій, застосовує ітеративний підхід (традиційно, ітерація — це 2–4 тижні).

Цей підхід базується на принципі постійного аналізу та поступових удосконалень, що допомагає знизити ризики виникнення серйозних помилок на пізніх етапах розробки. Важливим є розуміння того, що кожна наступна ітерація не просто виправляє помилки, а й поглиблює знання про користувачів, їхні звички, потреби та очікування, що допомагає створити максимально релевантний та ефективний інтерфейс. Процес ітеративного покращення передбачає постійний зворотний зв’язок між дизайнерами, розробниками і користувачами. Такий безперервний цикл взаємодії дає змогу краще зрозуміти, як саме інтерфейс використовується на практиці. Вже після перших циклів тестування стає очевидно, які рішення працюють добре, а які потребують доопрацювання чи повної заміни. Важливо розуміти, що зміни не завжди будуть масштабними; часто це можуть бути точкові, але суттєві коригування, наприклад, уточнення формулювань, коригування розміру елементів або ж поліпшення мікроанімцій.

Ефективність ітеративного підходу також полягає у тому, що він допомагає розподілити ресурсні витрати рівномірніше протягом усього періоду розробки, а не концентрувати їх на фінальному етапі. Кожна проведена ітерація тестування та внесення змін економить час та кошти, які довелося б витратити на масштабні переробки вже створеного продукту. Крім того, регулярне тестування створює впевненість у тому, що фінальний продукт відповідатиме усім запитам користувачів і бізнесу, що передусім підвищує його конкурентоспроможність на ринку.



Також варто пам'ятати, що процес ітеративного покращення ніколи не є завершеним остаточно. Навіть після запуску продукту команда продовжує збирати відгуки, відслідковувати поведінку користувачів та впроваджувати регулярні оновлення. Таким чином, дизайн та функціональність постійно еволюціонують відповідно до змін у потребах користувачів, нових трендів і технологій. Саме завдяки такому гнучкому підходу інтерфейс стає не лише зручнішим, але й довготривало релевантним і затребуваним серед користувачів.



Запитання до розділу 3

1. Чим каркас відрізняється від макету та прототипу?
2. Для чого потрібен каркасний макет на початкових етапах проектування мобільного застосунку?
3. Які переваги дає використання інтерактивного прототипу перед етапом розробки?
4. Назвіть основні інструменти, що використовуються для створення каркасів, макетів та прототипів мобільних застосунків.
5. У чому перевага ітеративного підходу до UX-дизайну?
6. Які помилки часто допускають дизайнери під час створення макетів мобільних застосунків?
7. Чому важливо проводити юзабіліті-тестування ще на стадії прототипу?
8. У чому різниця низькодеталізованого та високодеталізованого каркасу та на якому етапі кожен із них використовується?
9. Як адаптивність макету впливає на зручність мобільного застосунку?
10. Які методи використовуються під час тестування прототипів мобільних застосунків?

Завдання для самостійної роботи

Використовуючи Figma, створити інтерактивний прототип стартового екрану з анімацією до власного мобільного застосунку.





РОЗДІЛ 4. ПРОСУВАННЯ МОБІЛЬНИХ ЗАСТОСУНКІВ ТА МОНЕТИЗАЦІЯ

4.1. Стратегії просування мобільних застосунків ASO (App Store Optimization): як підняти застосунок у пошуку

Створення якісного мобільного застосунку — це лише частина успіху. Щоб продукт дійшов до своєї аудиторії, його потрібно просувати, а значить, працювати над його видимістю у сторах, залучати користувачів через рекламу та налагоджувати ефективну комунікацію з аудиторією. Якщо не приділити цьому належної уваги, навіть найкращий застосунок може загубитися серед тисяч конкурентів.

Одним із найефективніших способів підвищити видимість застосунку є App Store Optimization (ASO) — оптимізація сторінки застосунку в Google Play та App Store для покращення його позицій у пошуку. Більшість користувачів знаходять нові застосунки саме через пошук у магазинах, тому важливо, щоб сторінка застосунку була оптимізована за ключовими словами, мала привабливий опис і якісні візуальні матеріали. Назва, підзаголовок, ключові слова, опис,



іконка, скріншоти та відеопрезентація — все це має бути налаштоване так, щоб одразу привернути увагу користувача і переконати його завантажити застосунок.

Проте однієї ASO недостатньо, особливо якщо застосунок тільки виходить на ринок. Саме тут важливу роль відіграє реклама, яка допомагає швидко залучити перших користувачів. Це можуть бути платні кампанії у соціальних мережах, таргетована реклама у Facebook, Instagram, TikTok, а також оголошення у Google Ads, які ведуть людей безпосередньо на сторінку застосунку у сторах. Додатково можна використовувати впливовий маркетинг, співпрацюючи з блогерами або лідерами думок, які можуть розповісти про застосунок своїй аудиторії.

Однак навіть успішне залучення користувачів — це лише початок. Щоб люди не тільки встановлювали застосунок, а й активно ним користувалися, важливо працювати з утриманням аудиторії та відгуками. Відгуки у сторах мають прямий вплив на рішення інших користувачів завантажити застосунок, а також на його рейтинг у пошуку. Якщо користувачі залишають позитивні коментарі, застосунок отримує більше довіри та привертає більше завантажень. Важливо швидко реагувати на негативні відгуки, пропонувати підтримку та постійно вдосконалювати продукт на основі фідбеку.

Успішна стратегія просування мобільного застосунку поєднує ASO, рекламу та активну комунікацію з користувачами. Без оптимізації сторінки застосунку важко досягти високих позицій у пошуку, без реклами — залучити нову аудиторію, а без роботи з відгуками — утримати наявних користувачів. Правильний підхід до цих процесів допомагає не просто вивести застосунок на ринок, а й зробити його популярним і успішним.



Критерій	ASO (App Store Optimization)	SEO (Search Engine Optimization)
Сфера застосування	Мобільні додатки (App Store, Google Play)	Вебсайти (Google, Bing та інші пошуковики)
Основна мета	Збільшення кількості встановлень додатка	Збільшення трафіку та відвідуваності сайту
Тип контенту	Опис додатка, заголовки, скріншоти, відгуки	Статті, блоги, сторінки сайту, метадані
Чинники впливу	Відгуки, рейтинг, кількість завантажень, утримання користувачів	Якість контенту, посилання, ключові слова, швидкість завантаження
Інструменти аналізу	App Annie, Sensor Tower, Mobile Action	Google Analytics, Ahrefs, SEMrush
Ключові метрики	Встановлення, рейтинг, конверсія на сторінці додатку	Органічний трафік, позиції в пошуку, CTR

Табл. 1. Відмінності ASO та SEO

Оптимізація для Google Play

Оптимізація застосунку в Google Play Market є важливим аспектом успіху мобільного застосунку, оскільки допомагає збільшити його видимість, кількість завантажень і залученість користувачів. Для досягнення найкращих результатів слід враховувати низку конкретних рекомендацій.

Експерти рекомендують використовувати назву, що містить до 30 символів. Найрелевантніші ключові слова мають розміщуватися ближче до неї. Назва повинна бути лаконічною, зрозумілою та передавати суть застосунку максимально точно. Короткий опис становить 80 символів і повинен коротко розкривати головну функцію та переваги застосунку, стимулюючи користувачів ознайомитися із застосунком детальніше.

Повний опис може містити до 4000 символів, де варто докладно розкрити функціонал, унікальні можливості, переваги та особливості застосунку. Рекомендується використовувати ключові слова, але уникати їх надмірної концентрації. Текст має бути чітко структурованим, з логічним поділом на абзаци для зручного читання.

Графічні матеріали, такі як іконка та скріншоти подаються високої якості та візуально привабливими. Рекомен-



дується розміщувати до 8 скриншотів, що демонструють основні функції та інтерфейс застосунку. Використання промо-відео тривалістю до 30 секунд може значно підвищити залученість користувачів.

Окрім цього, потрібно регулярно аналізувати відгуки, оперативно відповідати на них та враховувати зауваження для покращення застосунку. Постійні оновлення, удосконалення технічних характеристик та стабільності роботи також сприяють вищому рейтингу й кращій позиції у результатах пошуку Google Play.

Оптимізація для App Store

Оптимізація застосунку для App Store має свої особливості, які в деяких аспектах відрізняються від оптимізації для Google Play. Назва застосунку в App Store повинна містити не більше 30 символів. Важливо включити в назву найважливіші ключові слова, які відображають основну функцію та цінність застосунку. Крім того, в App Store доступне окреме поле для підзаголовка, також до 30 символів, де можна додатково уточнити призначення або ключові переваги застосунку.

Опис застосунку в App Store має бути максимально змістовним та привабливим, проте важливо врахувати, що лише перші три рядки (близько 170 символів) є найважливішими, адже вони відображаються одразу, без додаткових дій користувача. Повний опис може містити до 4000 символів, у яких слід детально та структуровано описати всі ключові функції та переваги застосунку.

Особливу увагу варто приділити візуальному оформленню. Іконка застосунку повинна бути яскравою, легко впізнаваною та відповідати дизайну самого застосунку. Apple рекомендує використовувати до 10 якісних скриншотів, які наочно демонструють інтерфейс та ключові функції застосунку. Використання короткого та інформативного промо-



відео тривалістю до 30 секунд також позитивно впливає на залучення користувачів.

Важливим чинником оптимізації є вибір релевантних ключових слів. App Store надає окреме поле (до 100 символів), де можна прописати основні ключові слова, за якими застосунок буде з'являтися у пошуку.

Apple враховує оцінки та кількість позитивних відгуків, тому варто регулярно взаємодіяти з аудиторією, реагувати на відгуки та оперативно вирішувати проблеми, що виникають.

4.2. Залучення та утримання користувачів

Гейміфікація як спосіб підвищення взаємодії

Користувачі мобільних застосунків щодня стикаються з десятками нових продуктів, але залишаються лише з тими, які захоплюють, мотивують і створюють індивідуальний досвід. Щоб підтримувати високу залученість, розробники використовують три ключові механізми: гейміфікацію, персоналізовані нотифікації та аналіз поведінки користувачів.

Гейміфікація — один із найпотужніших способів зробити взаємодію цікавою та втримати увагу користувачів. Вона працює за принципом впровадження ігрових підходів у неігрових контекстах. До таких елементів можна зарахувати нагороди за виконані дії, прогрес-бари, рівні, рейтинги та змагання. У медійних застосунках розміщуються міні-ігри, у фінтех-сервісах можуть з'являтися челенджі на тему економії, а у фітнес-застосунках — щоденні цілі, які приносять віртуальні винагороди. Це формує звичку взаємодії із застосунком і робить процес використання цікавішим.

Сповіщення та персоналізація контенту

Але просто залучити користувача — недостатньо. Щоб утримувати його інтерес, необхідна персоналізація контенту



та доцільна стратегія сповіщень. Сповіщення — це прямий канал зв'язку між застосунком і користувачем, але важливо не перевантажувати його непотрібними повідомленнями, які можуть викликати роздратування. Персоналізовані сповіщення, засновані на поведінкових даних, значно ефективніші. Вчасно відправлене повідомлення, яке відповідає інтересам користувача, підвищує ймовірність повернення до застосунку.

Аналітика користувацької поведінки

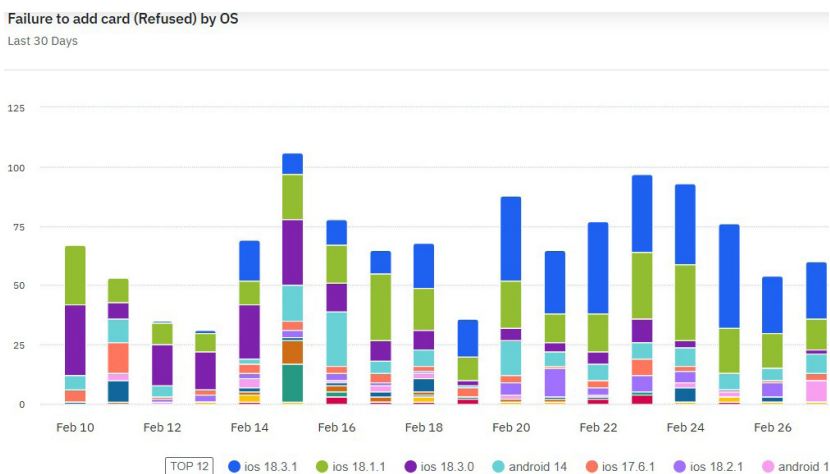


Рис. 34. Зразок аналітики в Amplitude

Однак ефективність гейміфікації та персоналізації неможливо оцінити без аналітики користувацької поведінки. Сучасні мобільні застосунки збирають дані про те, як люди взаємодіють із продуктом: які функції використовують найчастіше, де вони затримуються, на якому етапі можуть відмовитися від використання. Використовуючи аналітичні інструменти (Google Analytics for Firebase, Amplitude, Mixpanel), розробники можуть виявити проблемні місця в



UX та оптимізувати досвід користувачів. Наприклад, якщо видно, що більшість користувачів залишають застосунок на стадії реєстрації, варто спростити процес або додати альтернативні методи входу.

Поєднання гейміфікації, персоналізації та глибокого аналізу користувацької поведінки створює динамічний і адаптивний продукт, який не просто привертає увагу, а й утримує аудиторію, перетворюючи випадкових користувачів у лояльних прихильників застосунку.

4.3. Моделі монетизації мобільних застосунків

Безкоштовні застосунки з рекламою

Розробка мобільного застосунку — це не тільки створення корисного або розважального продукту, але й можливість отримувати з нього прибуток. У сучасному мобільному ринку існує кілька основних моделей монетизації: безкоштовні застосунки з рекламою, підписки та внутрішньоігрові покупки, а також платні застосунки з преміальними функціями. Вибір моделі залежить від типу продукту, його аудиторії та стратегії розвитку.

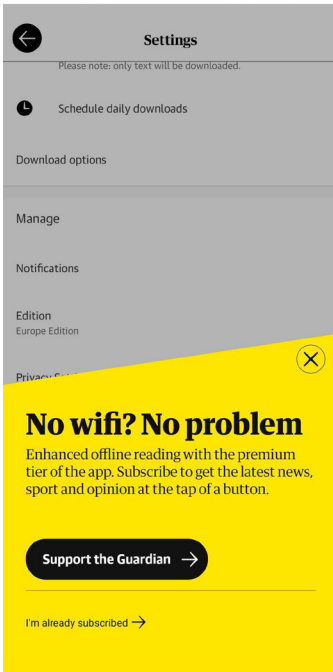
Безкоштовні застосунки з рекламою є найпоширенішим способом монетизації. Оскільки користувачі не завжди готові платити за встановлення застосунку, реклама допомагає отримувати прибуток без потреби змушувати їх купувати контент. У таких застосунках можуть бути банерна реклама, відеореклама, інтегровані рекламні блоки або нативна реклама, яка виглядає як частина контенту. Наприклад, у мобільних іграх часто показують рекламу після завершення рівня, а у безкоштовних застосунках для читання або перегляду відео з'являються рекламні вставки перед або під час використання. Важливо знайти баланс: якщо реклама буде надто нав'язливою, користувачі можуть відмовитися від застосунку або знайти альтернативу без реклами.



Підписки та внутрішні покупки

Підписки та внутрішні покупки — це модель, яка допомагає монетизувати застосунок через повторні платежі. Застосунки, що працюють за підпискою, надають користувачам доступ до преміального контенту або функцій за щомісячну або річну оплату. Внутрішні покупки (in-app purchases) — ще одна ефективна стратегія, особливо у мобільних іграх. Користувачі можуть купувати віртуальну валюту, додаткові функції або персоналізовані елементи для покращення досвіду гри чи функції. Ця модель працює особливо добре у фріміум-іграх (free-to-play), де базовий доступ безкоштовний, але розширені можливості відкриваються за додаткову оплату.

Платні застосунки та преміальні функції



Платні застосунки та преміальні функції — це класична модель монетизації, коли користувачі сплачують за встановлення застосунку або окремі функції всередині нього. Такий підхід менш популярний у сучасному мобільному ринку, оскільки аудиторія рідко погоджується платити за застосунок, не спробувавши його. Проте він добре працює для професійних інструментів, навчальних застосунків та продуктах для бізнесу, де якість контенту та функціональність важливіші за безкоштовний доступ. Крім того, деякі застосунки використовують модель «free trial», коли користувач



отримує можливість випробувати застосунок безкоштовно впродовж певного періоду, а потім оплачує доступ до повної версії.

Ефективна монетизація мобільного застосунку залежить від правильного вибору моделі, яка відповідає очікуванням користувачів і природно інтегрується у продукт. Успішні застосунки часто комбінують кілька стратегій, наприклад, поєднуючи рекламу з преміальними підписками або внутрішніми покупками. Головне — зберігати баланс між заробітком і комфортом користувача, щоб монетизація не заважала основному функціоналу та не знижувала залученість аудиторії.



Запитання до розділу 4

1. Що таке ASO і чому воно є важливим для просування мобільного застосунку?
2. У чому різниця між оптимізацією застосунку для Google Play і App Store?
3. Які основні чинники впливають на позиції застосунку в пошуку у сторсах?
4. Які інструменти використовуються для аналізу ефективності просування застосунків?
5. Як гейміфікація допомагає утримувати увагу користувачів?
6. Які переваги персоналізованих сповіщень над звичайними повідомленнями?
7. Чому аналіз користувацької поведінки є важливим етапом для утримання користувачів?
8. Назви основні моделі монетизації мобільних застосунків.
9. Які ризики має модель монетизації за допомогою реклами?
10. Для яких типів застосунків найкраще підходить модель підписки?

Завдання для самостійної роботи

Продумати та записати концепцію механізмів гейміфікації та персоналізації для покращення залучення користувачів до власного мобільного застосунку.



Література

1. Голембо, В. А. (2020). UX/UI дизайн: Основи проектування інтерфейсів. Київ: ArtHuss.
2. Еяль, Н., Гувер, Р. (2014). Покупец на гачку. Київ: Манн, Иванов і Фербер. Режим доступу: <https://psyfactor.org/lib/trigger2.htm>
3. Круг, С. (2021). Не змушуйте мене думати! Веб-дизайн з погляду користувача. Київ: Наш формат.
4. Малевич, В. (2022). Дизайн цифрових продуктів. Харків: Фабула.
5. Нільсен, Я. (2000). Веб-дизайн: книга Якоба Нільсена. Київ: Вільямс.
6. Психологія дизайну: Як розташування та зовнішній вигляд елементів можуть впливати на користувача. Genius Space. <https://genius.space/lab/psihologiya-dizajnu-yak-roztashuvannya-ta-zovnishnij-viglyad-elementiv-mozhut-vplivati-na-koristuvacha/>
7. Apple (2020). Apple Operating Systems: A History. Режим доступу: <https://www.apple.com/os-history/>
8. Benyon, D. (2019). Designing Interactive Systems: A Comprehensive Guide to HCI, UX & Interaction Design (4th ed.). Pearson.
9. Cooper, A., Reimann, R., & Cronin, D. (2007). About Face: The Essentials of Interaction Design (3rd ed.). Wiley.
10. DuBay, W. H. (2004). A Model of Text Readability. Impact Information.
11. Fogg, B. J. (2009). A Behavior Model for Persuasive Design. Stanford Persuasive Technology Lab. Режим доступу: <https://ps.journal.kspu.edu/index.php/ps/article/download/4519/3989>
12. Garrett, J. J. (2010). The Elements of User Experience: User-Centered Design for the Web and Beyond. New Riders.

13. Google (2021). Android Operating System: Evolution and Impact. РЕЖИМ ДОСТУПУ: <https://www.android.com/our-history/>
14. Hooper, S. (2017). Design for Fingers, Touch, and People: Part 1. UXmatters. РЕЖИМ ДОСТУПУ: <https://www.uxmatters.com/mt/archives/2017/03/design-for-fingers-touch-and-people-part-1.php>
15. Johnson, J. (2021). Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Guidelines (3rd ed.). Morgan Kaufmann.
16. Nielsen, J. (1993). Usability Engineering. Morgan Kaufmann Publishers.
17. Norman, D. A. (2013). The Design of Everyday Things (Revised and Expanded ed.). MIT Press.
18. Rieman, J., Franzke, M., & Redmiles, D. (1995). Consistency in User Interface Design: An Empirical Study. Behaviour & Information Technology, 14(3), 137–148.
19. Saffer, D. (2014). Microinteractions: Designing with Details. O'Reilly Media.
20. Shneiderman, B., Plaisant, C., Cohen, M., & Jacobs, S. (2016). Designing the User Interface: Strategies for Effective Human-Computer Interaction (6th ed.). Pearson.
21. Sullivan, M. (2019). The Evolution of Mobile Operating Systems: From Early Smartphones to Modern iOS and Android. Tech History Journal, 15(2), 42–56.
22. Tech Innovators (2020, January 10). A Brief History of Mobile Operating Systems. Tech Innovations. РЕЖИМ ДОСТУПУ: <https://www.techinnovations.com/history-of-os>
23. TechCrunch (2019). The Rise and Dominance of Mobile OS Platforms. РЕЖИМ ДОСТУПУ: <https://techcrunch.com/2019/08/21/the-rise-and-dominance-of-mobile-os/>
24. Tognazzini, B. (2014). First Principles of Interaction Design. Nielsen Norman Group.

25. Weinschenk, S. (2020). 100 Things Every Designer Needs to Know About People (2nd ed.). Pearson.

26. Yablonski, J. (n.d.). Laws of UX: Psychological Principles for Effective Design. Режим доступу: <https://lawsofux.com/>

Рекомендовані онлайн-ресурси

27. Apple Human Interface Guidelines. Режим доступу: <https://developer.apple.com/design/human-interface-guidelines/>

28. Google Material Design Guidelines. Режим доступу: <https://m3.material.io/>

Навчальне видання

Наталія Михайлівна Іщук

ДИЗАЙН МОБІЛЬНИХ ЗАСТОСУНКІВ

Посібник

для студентів спеціальності 061 (С7) «Журналістика»

Гарнітура Minion Pro.
Формат 57.8×84/16