

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЧЕРНІГІВСЬКА ПОЛІТЕХНІКА»
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЕЛЕКТРОННИХ ТА
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЛЮДИНО-МАШИНА ВЗАЄМОДІЯ

МЕТОДИЧНІ ВКАЗІВКИ

для виконання лабораторних робіт
для здобувачів вищої освіти
першого (бакалаврського)
рівня вищої освіти за спеціальністю
121 – "Інженерія програмного забезпечення"

Обговорено і рекомендовано
на засіданні кафедри
інформаційних технологій та
програмної інженерії
протокол № 1 від 31.08.2021 р.

Чернігів 2021

Людино-машинна взаємодія. Методичні вказівки для виконання лабораторних робіт для здобувачів вищої освіти першого (бакалаврського) рівня вищої освіти за спеціальністю 121 – "Інженерія програмного забезпечення" / Укл. Д.Я. Руднєв, М.М. Войцеховська. – Чернігів: НУ «Чернігівська політехніка», 2021. – 36 с., укр. мовою.

Укладачі: Руднєв Дмитро Ярославович, аспірант, викладач, навчальний ІТ-центр «Гранд»

Войцеховська Марія Михайлівна, д.ф., викладач кафедри інформаційних технологій та програмної інженерії

Відповідальний за випуск: Войцеховська М.М., д.ф., викладач кафедри інформаційних технологій та програмної інженерії

Рецензент: Базилевич Володимир Маркович, завідувач кафедри інформаційних та комп'ютерних систем, к.е.н., доцент

ЗМІСТ

Передмова	4
Лабораторна робота №1 Аналіз предметної області, написання сценаріїв використання, побудова Use Case діаграм та діаграм діяльності	6
Лабораторна робота №2 Створення фреймворків інтерфейсу мобільного додатку у Figma	13
Лабораторна робота №3 Проектування дизайну мобільного додатку у середовищі Figma	16
Лабораторна робота №4 Прототипування взаємодії користувача з інтерфейсом	22
Лабораторна робота №5 Шаблони сканування інформації та принципи візуального дизайну	28
Лабораторна робота №6 Структура презентації дизайн-рішень	32
Рекомендовані джерела.....	36

Передмова

Кожен день ми спілкуємося, шукаємо потрібну інформацію або просто проводимо час за спогляданням зображень. Кожна компанія-розробник змагається за увагу користувача та час, який він витрачає на користування їх продуктом. І, поряд із головними перевагами та корисністю від додатку, особлива увага приділяється архітектурі інформації, проектуванню взаємодії, дизайну інтерфейсу, грамотному розміщенню контенту тощо.

Така тенденція зумовлює прискорення розробки. Сучасні системи дизайну відкрили можливість швидко продукувати різноманітні інтерфейси, не вдаючись до довготривалого та кропіткого кодування. Для дизайну інтерфейсів використовують Photoshop, Sketch, Figma або інші редактори. Для виконання лабораторних робіт розглянемо процес дизайну інтерфейсів у Figma.

Figma – програмний продукт, який затьмарив лідера серед дизайн-середовищ Adobe Photoshop та його наступника Sketch. Головними перевагами Figma є незалежність від ОС, наявність браузерної версії, можливість одночасної командної роботи над проектом, проектування інтерактивності прототипів та історія версій.

Для роботи з Figma знадобиться обліковий запис, який може бути створений на базі вже існуючого аккаунту Google або соцмереж. Такий підхід забезпечує доступність проектів з будь-якої платформи.

Для розширення можливостей проектування інструментарій може бути доповнений плагінами. Завдяки **Community** талановитих розробників Figma має велику кількість різноманітних плагінів для анімації, дизайну іконок, редагування зображень, моделювання переходів, та реалізації безлічі інших цікавих та креативних ідей.

Для цього в меню Figma слід перейти до розділу Community та обрати вкладку **Plugins**. Якщо назва потрібного плагіну відома заздалегідь, зручніше скористатися пошуком.

Додатковою приємною опцією Figma є підтримка командної роботи над проектом. Правами на перегляд та редагування легко можна керувати завдяки опції Share, що розміщена у правому верхньому куті екрану.

Team Library – чудовий інструмент для обміну та використання спільних елементів (як, наприклад, іконки, зображення, тощо). Скористатися ним можна за допомогою меню, наведеного на рисунку А.1.

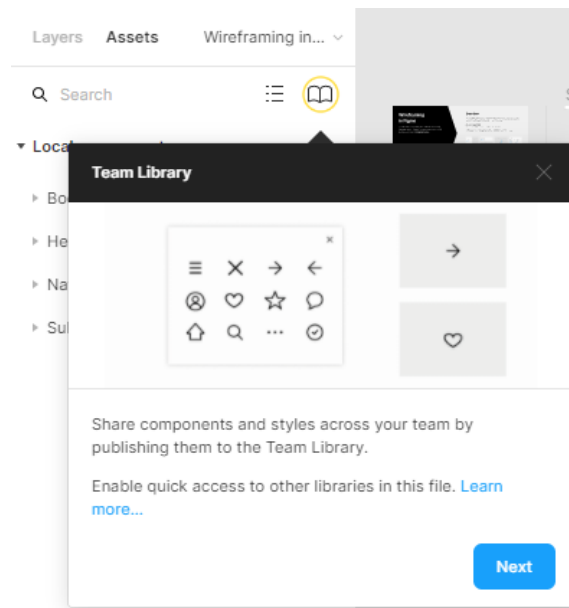


Рисунок А.1 – Вкладка бібліотеки елементів Team Library

У середовищі передбачена можливість експортувати ескізи з попередніх проектів, що суттєво зберігає час та зусилля, витрачені на розробку типових фреймів та компонент.

Важлива роль у процесі проектування інтерфейсів належить прототипуванню взаємодії. Окрім вбудованого сервісу перегляду прототипу, мобільний додаток **Figma Mirror** забезпечує миттєве відображення екранів на будь-якому гаджеті, що дозволяє переглядати результат проектування на відповідному екрані.

Правила оформлення звітів

Звіти мають бути оформлені українською мовою, Times New Roman, 14. Заголовок та мета відповідають лабораторній роботі. Обов'язково має бути зазначене завдання.

Ілюстрації мають бути підписані та пронумеровані: перша цифра – номер лабораторної, друга – порядковий номер рисунку в межах звіту. Наприкінці виконаної роботи – висновки за виконанням.

По завершенні циклу лабораторних робіт звіт повинен бути доповнений титульною сторінкою та скріплений. При проведенні навчання за допомогою дистанційної системи Moodle звіт з виконання окремої лабораторної роботи повинен бути завантажений як відповідь на відповідне завдання.

Лабораторна робота №1

Аналіз предметної області, написання сценаріїв використання, побудова Use Case діаграм та діаграм діяльності

Мета: навчитися основам побудови сценаріїв, діаграм варіантів використання та діаграм активності.

Теоретична частина



Для створення ефективної взаємодії та взаєморозуміння розробників ПЗ Граді Буч, Джеймс Рамбо та Івар Якобсон (розробник концепції об'єктно-орієнтованої інженерії ПЗ) систематизували наявні на той час підходи до моделювання складних систем. Так на світ з'явилася мова графічного опису для об'єктного моделювання, яку ми знаємо під назвою Unified

Modeling Language, UML. Ця мова не є мовою програмування, проте на основі візуалізованих за її допомогою моделей систем можлива генерація коду.

Серед різноманіття діаграм, що використовуються для опису UML-моделей, на початку навчальної діяльності будуть у нагоді дві діаграми поведінки, а саме: діаграма варіантів використання та діаграма активності.

Та початковою інформацією для розробки виступають сценарії.

Сценарії

Сценарій – це один із способів описати структуру поставленої задачі. Він може бути створений у вигляді розповіді про дії персонажу, це історія, епізод, що відбувається в даних часових рамках та у даному контексті.

Різні форми сценаріїв широко застосовуються при розробці програмного забезпечення. Сценарії завдань і взаємодій зазвичай багаті характеристиками і мають високу реалістичність. Сценарії при розробці користувальницького інтерфейсу описують взаємодію між користувачем (або типом користувачів) і системою. Звичайні сценарії мають деякі серйозні обмеження при спробі використовувати їх для проектування призначеного для користувача інтерфейсу. У них робиться наголос на реалістичність і деталі, при цьому на серйозні проблеми і загальну організацію звертається недостатньо уваги. Сценарії включають в себе правдоподібні описи комбінацій окремих дій і завдань, тому часто буває важко виділити і зрозуміти основну суть взаємодії.

Сценарії мають містити не лише послідовності дій, які приводять до бажаного результату. У світі не все відбувається за нашими бажаннями, тож бувають ситуації, коли «щось пішло не так». Це називається альтернативним потоком, та має бути передбачений під час розробки сценарію. Система має адекватно діяти навіть при порушенні ідеального розвитку подій.

Функціональні вимоги до системи моделюються і документуються у вигляді варіантів використання (Use Case). Варіанти опису варіантів

використання можуть бути представлені у відповідності до шаблону RUP або за описом Коберна. В рамках даної лабораторної роботи достатньо систематизувати вимоги до функціоналу у вигляді Таблиці 1.1.

Таблиця 1.1 – Сценарії взаємодії акторів із системою

Актор	Функція	Сценарій
Користувач		
...		

UML-діаграми

Для реалізації сценаріїв у вигляді графічних послідовностей використовують UML-діаграми.

Сутності

UML-діаграма складається із сукупності елементів, що називаються сутностями, та зв'язків між ними. Виконання завдання лабораторної роботи дозволить ознайомитися з лише де-якими з них. Отже, розглянемо найнеобхідніше.

Сутність – *Entity* – абстракція, що використовується для моделювання, та є основним об'єктно-орієнтовним елементом мови UML.

Дійова особа – *Actor* – сутність, що існує поза межами системи, що моделюється, та яка безпосередньо взаємодіє із системою.

Варіант використання – *Use Case* – складається з множини сценаріїв, які поєднані за певним критерієм та описують послідовність реакцій системи, що забезпечують суттєвий результат для Дійової особи (Актора).

Дія – *Action* – примітивне атомарне обчислення.

Сутність-примітка – *Note* – коментар, який використовується для надання додаткової інформації до будь-якого елемента моделі.

Для подальшого моделювання системи використовуються Діаграми, які відображають структуру створюваної моделі, що в подальшому суттєво полегшує її сприйняття та використання.

Діаграма – *Diagram* – це графічне представлення деякої частини графа моделі.

В даній роботі ми розглянемо лише дві канонічні (незмінні) діаграми, а саме:

- **діаграма варіантів використання** (вона же «діаграма прецедентів», вона же «use case діаграма») – надає найбільш загальне представлення про функціональне призначення системи.
- **діаграма діяльності** (активності) – спосіб візуального опису поведінки, подібного до блок-схеми алгоритму.

Відношення

Серед різноманіття відношень, які закладені у UML, нас цікавитимуть лише декілька основних типів, які знадобляться при побудові use case діаграми та діаграми активності.

Асоціація – *Association* – зв'язок, що відображає безпосередню взаємодію однієї сутності з іншою (або іншими). Прикладом асоціації є зв'язок між Актором та Варіантом використання.

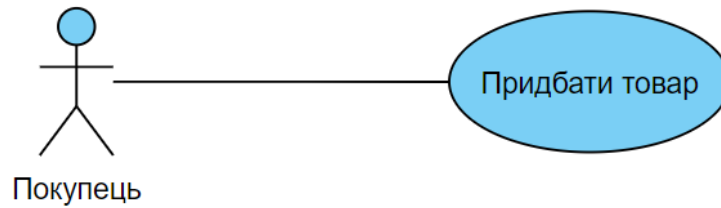


Рисунок 1.1 – Зв'язок асоціації між актором та варіантом використання

Узагальнення – *Generalization* – це відношення між двома сутностями, коли одна з них є окремим (спеціалізованим) випадком іншої. Василь є одним із покупців інтернет-магазину, а його замовлення на куртку – одним із багатьох замовлень.

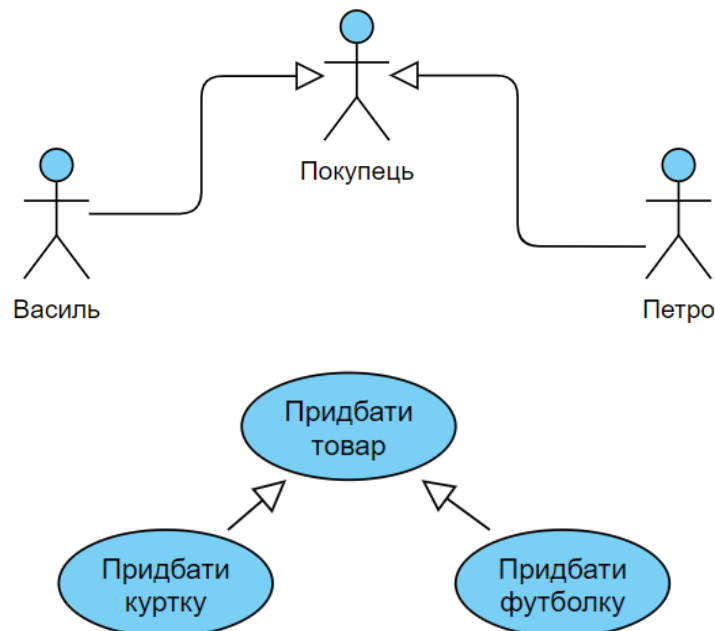


Рисунок 1.2 – Приклади узагальнення

Залежність – тип зв'язку між варіантами використання, який вказує, що один варіант використання залежить від іншого. Мова моделювання UML передбачає два типи залежності:

- **включення** (в деяких джерелах – композиція) – *«include»* – свідчить про те, що один варіант використання в певний момент часу повністю включає в себе виконання ще одного (або декількох) варіанту(-ів) використання;
- **розширення** – *«extend»* – вказує на можливість задіяння одного варіанта використання при виконанні іншого.

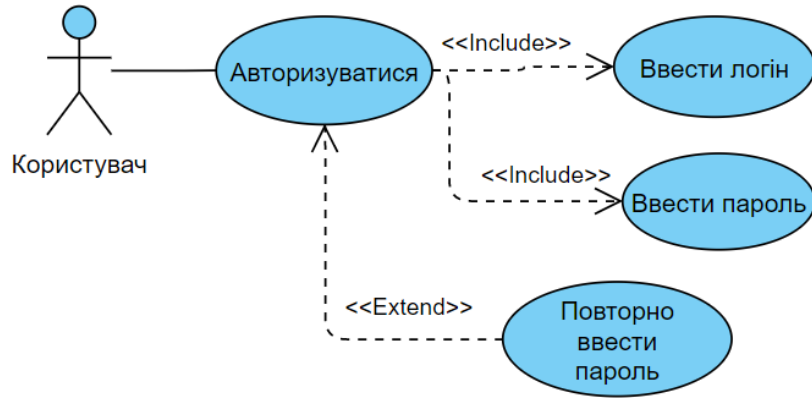


Рисунок 1.3 – Приклад застосування зв'язків включення та розширення

Діаграма варіантів використання

Діаграма прецедентів зазвичай використовує два основні типи сутностей: варіант використання та дійові особи, між якими встановлюються відношення асоціації, узагальнення та залежності (включення та розширення).

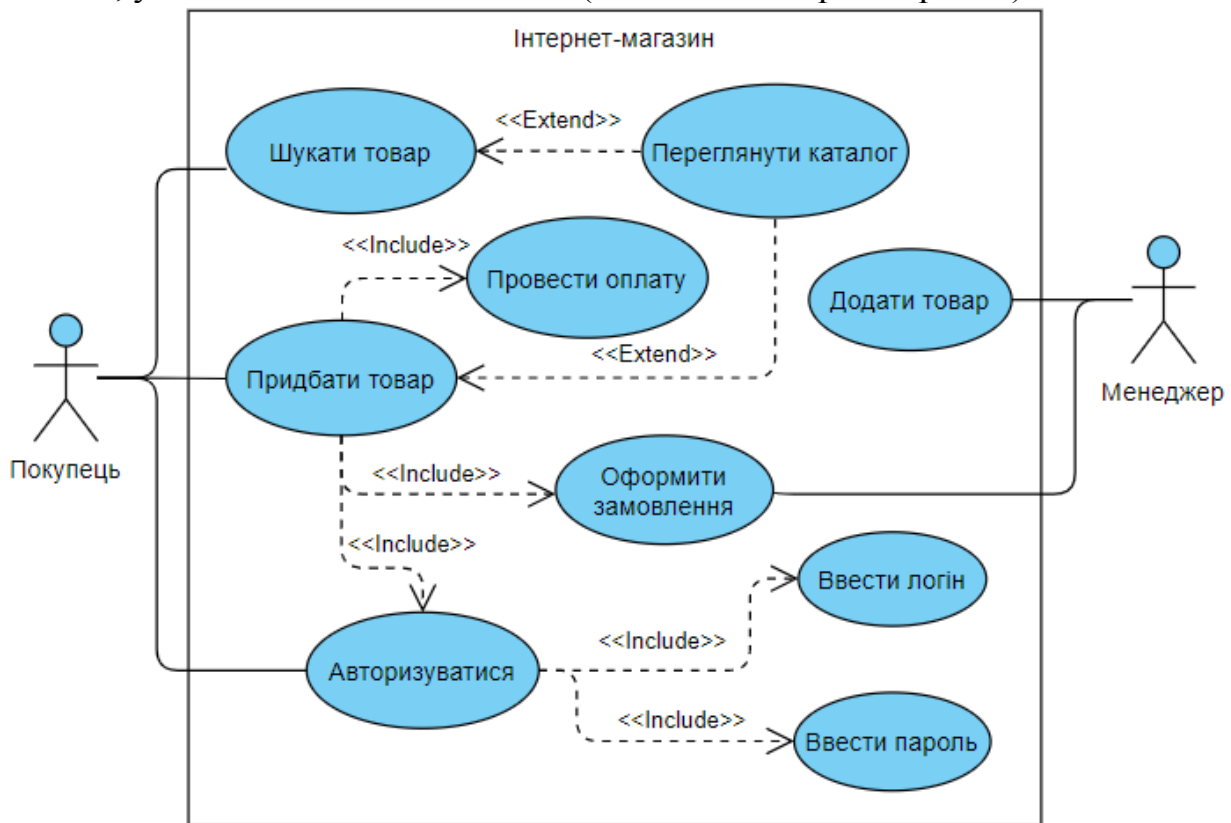


Рисунок 1.4 – Фрагмент Use Case діаграми

Діаграма активності

У діаграмі активності використовують один основний тип сутностей – дію (*action*), та один тип відношень – переходи або передачі керування (*transition*). Проте невелика кількість ключових елементів діаграми компенсується різноманіттям конструкцій: рішення, розгалуження, поєднання, початковий та кінцевий стани.

Окремим елементом діаграми активності можна виділити доріжки. Їх використання навіяно порівнянням з доріжками у басейні, які розділяють територію басейну для зручності пловців. Доріжки символізують сферу відповідальності певного актора або групи за визначену дію.

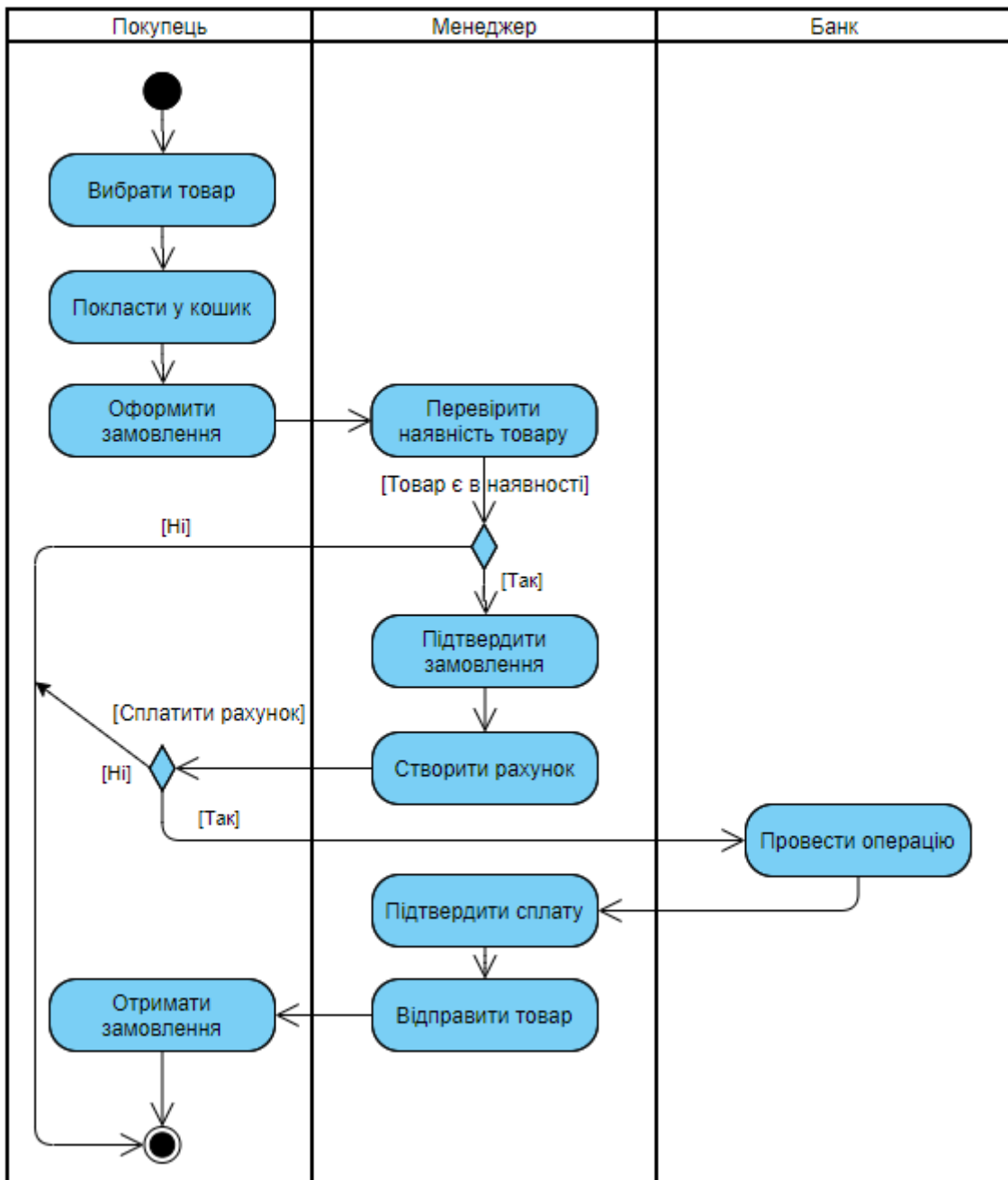


Рисунок 1.5 – Фрагмент діаграми активності покупця при придбанні товару в інтернет-магазині

Зміст роботи

1. Провести аналіз предметної області відповідно до обраного завдання.
2. Скласти п'ять сценаріїв використання програмного забезпечення користувачем.
3. Вивчити основи побудови use case діаграм і діаграм діяльності.
4. Побудувати use case діаграми.

5. Побудувати діаграми діяльності для кожного варіанту використання.
6. Оформити звіт про виконану роботу.
7. Захистити лабораторну роботу.

Варіанти завдань

1. **Інтернет магазин.** Повинні бути реалізовані сценарії: покупка товару, пошук товару, додавання нового товару в базу даних магазину, перегляд і обробка замовлень покупців, реєстрація нового покупця.
2. **Книжковий каталог.** Повинні бути реалізовані сценарії: додавання нової книги, пошук книги по декількох полях, бронювання книги, списання старих книг, реєстрація користувачів каталогу.
3. **Адресна книга.** Повинні бути реалізовані сценарії: додавання нового абонента, додавання категорій абонентів, пошук абонентів по декількох полях, додавання адміністратора каталогу (користувачів, які мають право редагувати дані адресної книги), редагування даних абонента.
4. **Розклад занять.** Повинні бути реалізовані сценарії: додавання нової групи, додавання занять (із зазначенням назви предмета, часу, аудиторії, групи, тижні, викладача, типу заняття), перегляд списку занять за обраною датою, додавання списку викладачів, пошук занять по декількох полях (предметом, викладачем, групою, часом, типом заняття).
5. **База студентів.** Повинні бути реалізовані сценарії: додавання нової групи, додавання нового студента, пошук студента по різних полях, додавання інформації про оцінки з різних предметів, відрахування студента.
6. **Прайс-лист фірми.** Повинні бути реалізовані сценарії: додавання нової категорії товарів, додавання нового товару, пошук товару по різних полях, додавання адміністратора прайс-листа (користувачів, які мають право редагувати прайс-лист), переміщення товару з однієї категорії в іншу.
7. **База складу фірми.** Повинні бути реалізовані сценарії: додавання нового товару на склад, списання товару, видача товару, пошук товару по різних полях, зміна місця розташування товару на складі.
8. **Аптечна база.** Повинні бути реалізовані сценарії: прийом замовлення від клієнта на виготовлення розчину, продаж ліків, списання прострочених ліків, додавання нових ліків в базу даних, пошук замовлень по різних полях.
9. **Кур'єрська служба.** Повинні бути реалізовані сценарії: прийом відправлення від клієнта, переадресування доставки, повернення відправлення, оформлення накладної, сплата послуг.
10. **Експрес-піцца.** Повинні бути реалізовані сценарії: розміщення замовлення, додавання нових позицій в меню, замовлення унікального набору інгредієнтів, пошук найближчої піцерії, розрахунок з використанням дисконтної карти.

11. Туристична агенція. Повинні бути реалізовані сценарії: пошук путівки за критеріями (ціна, дата, місто, тип готелю, сервіс тощо), бронювання путівки, зміна дати бронювання, пошук авіабілетів, замовлення додаткової екскурсії (туру).

Контрольні питання

1. Що таке UML?
2. Що таке сутність?
3. Що таке діаграма?
4. Які типи відношень передбачені у діаграмі варіантів використання?
5. Що таке дія?
6. Для чого створені діаграми варіантів використання?
7. Для чого призначені діаграми активності?
8. Які складові елементи діаграми активності?

Лабораторна робота №2

Створення фреймворків інтерфейсу мобільного додатку у Figma

Мета: ознайомитися з середовищем UX/UI дизайну Figma, створити фрейми для мобільних додатків.

Теоретична частина

Фрейм (wireframe, каркас) – це основне представлення інтерфейсу користувача. Для його розробки використовують лише чорний, білий та сірий кольори, завдяки чому фрейм взаємодії виглядає дуже просто. Вони не мають бути піксельно-ідеальними, тобто інтервал між елементами на сторінці не повинен виглядати саме так, яким ви хочете бачити його у кінцевому продукті. Суттєвим моментом для фреймів є можливість їх швидкого виготовлення.

Навіщо UX дизайнери використовують фрейми? Після проведення досліджень та планування настає фаза дизайну. Це означає, що після того, як були проведені бесіди з користувачами та аналіз конкурентів, розробники переходять до створення каркасів. Саме така послідовність дій робить фрейми більш інформативними. Те, що ви вирішили розробити, базується на тому, що ви почули, що хочуть ваші потенційні користувачі, що потрібно вашому бізнесу та які функції, на вашу думку, будуть кращими, ніж у конкурентів.

Мета полягає в тому, щоб швидко реалізувати запланований дизайн, для того щоб потім показати його користувачам і швидше провести тест на зручність користування (юзабіліті, usability).

Але навіщо це робити перед створенням високоточного макету? Тому що не слід замикатися на дизайні. Якщо зробити точний макет та протестувати його у потенційних користувачів, то може виявитися, що користувачам не сподобалося 70% роботи, тож цю частину доведеться перероблювати.

Фрейми швидко та легко піддаються зміні завдяки своїй простоті. В той же час, переробка макету потребує набагато більше часу та зусиль.

При розробці фрейму переслідують єдину мету – розмістити основну інформацію та сторінці та зробити її зручною для сприйняття. Блоки тексту, кнопки та зображення підпорядковуються архітектурі (ієрархії) інформації. Саме вона визначає положення всіх елементів на сторінці. Для того, щоб мати повне уявлення про каркаси для кожної сторінки сайту, рекомендують розміщувати каркаси на одній сторінці проекту. Ця рекомендація особливо цінна при роботі в команді, оскільки дає змогу бути в курсі змін, внесених колегами.

Подальший дизайн користувацької взаємодії реалізують за допомогою інтерактивних переходів. Це дає змогу запланувати всі можливі дії користувачів при використанні інтерфейсу. Слід також згадати про важливу роль створення попереднього прототипування, оскільки ще на етапі розробки фреймів можна побачити та оцінити обсяг запланованих робіт на реалізацію проекту.

Основні елементи каркасу

Фрейм, у його вузькому розумінні, – це екран майбутнього девайса. Беручи до уваги різноманітність можливих відображень, вибір відповідного екрану є першим кроком для розробки каркасу (рисунок 2.1).

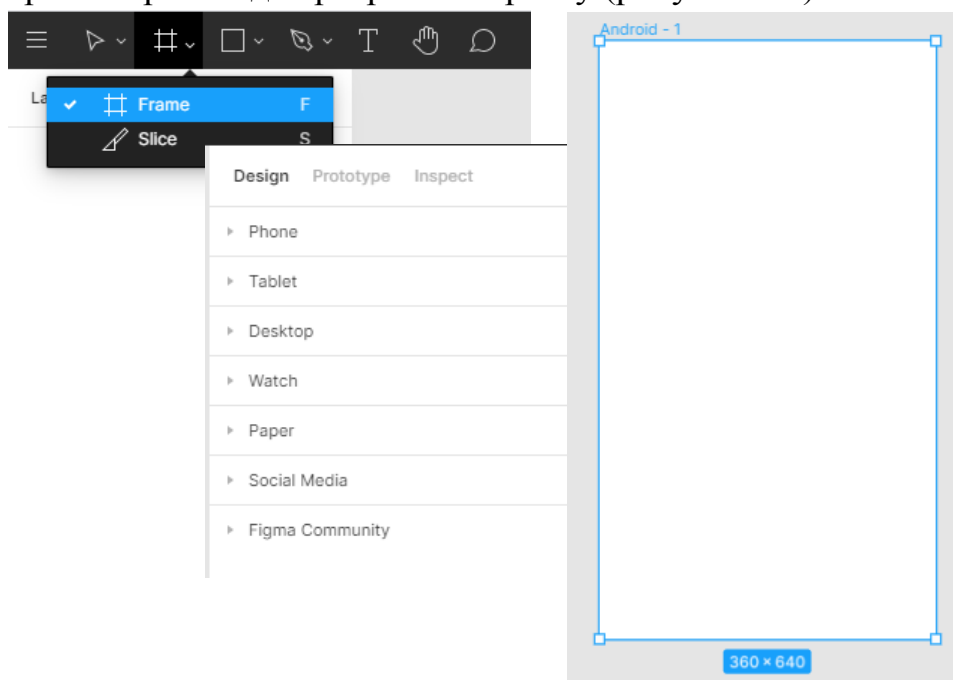


Рисунок 2.1 – Вкладка вибору фрейму

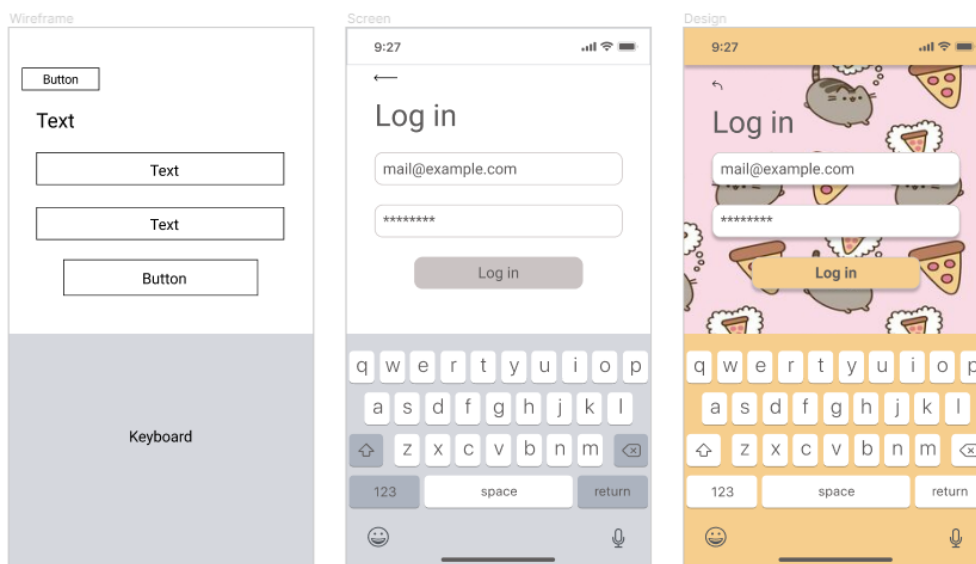


Рисунок 2.2 – Відмінності фрейму та дизайну одного екрану

Наступний за важливістю елемент – прямокутник. Саме він є найбільш вживаною фігурою для розробки фреймів будь-якого призначення. За допомогою прямокутника позначають поля для введення тексту, кнопки, зображення тощо.

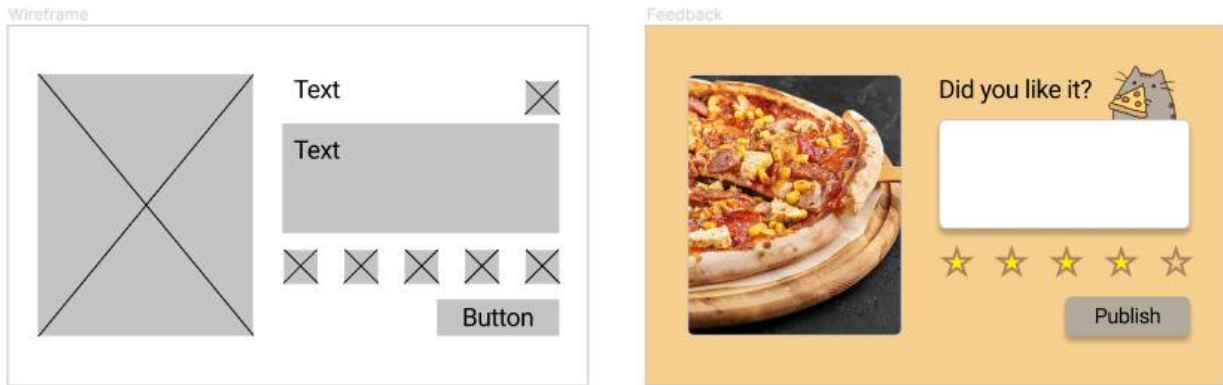


Рисунок 2.3 – Умовні позначення відповідних елементів фрейму

Зображення (image) позначається фігурою з діагонально перехрещеними лініями. Інтерактив інтерфейсу, зазвичай, реалізовано за допомогою кнопок (button), полей, меню тощо.

Окрім візуальної інформації, розміщеної у вигляді зображень, при проектуванні взаємодії велику увагу приділяють тексту. Він може виступати як заголовок, підпис, коментар, самостійний блок. Позначка «Text» також необхідна при вказуванні текстового поля.

Зміст роботи

1. Провести аналіз предметної області відповідно до обраного завдання.
2. Розробити п'ять фреймів для мобільного додатку.
3. Обґрунтувати розміщення елементів.
4. Оформити звіт про виконану роботу.
5. Захистити лабораторну роботу.

Варіанти завдань

1. Мобільний додаток для інтернет-магазину квітів.
2. Мобільний додаток банку.
3. Мобільний додаток для піцерії.
4. Мобільний додаток спортивних тренувань.
5. Мобільний додаток для магазину одягу.
6. Мобільний додаток “календар подій”.
7. Мобільний додаток для інтернет-магазину ігрових девайсів.
8. Мобільний додаток для кінотеатру.

Контрольні питання

1. Що таке фреймворк?
2. Для чого використовують фрейми?
3. Які основні елементи фрейму?
4. Як встановити розмір фрейму у Figma?

Лабораторна робота №3

Проектування дизайну мобільного додатку у середовищі Figma

Мета: отримати навички у розробці дизайну інтерфейсу мобільного додатку на основі фреймів.

Теоретична частина

Після того, як набір фреймів розроблено, обговорено та затверджено, приступають до дизайну UI. Саме на цьому етапі на зміну логіці та законам UX приходять креативність та знання правил та закономірностей UI. Всі елементи, які під час створення каркасів будуть представлені лише як схематичні позначки (прямокутники та текст), набувають зрозуміліший вигляд.

При роботі з Figma стануть у нагоді уміння використовувати навіть базові її можливості.

Основні елементи дизайну інтерфейсу

Сітки, колонки, рядки та змішана система.

Для полегшення роботи дизайнерів, а надалі і розробників, у середовищі передбачені елементи розмітки: сітки, колонки та їх поєднання.

За замовчуванням Figma створює сітку (*Grid*). Вона зручна при розробці мобільних додатків.

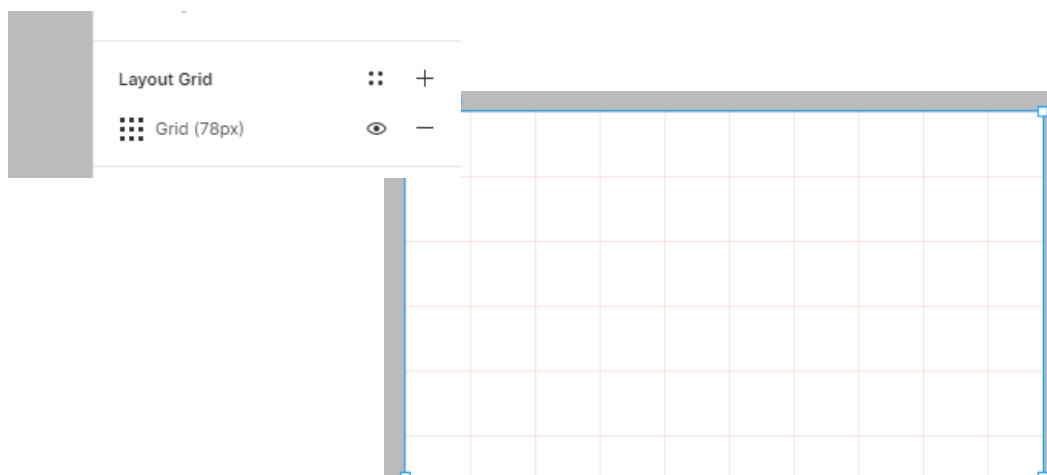


Рисунок 3.1 – Розмітка Сітка

У веб-дизайні частіше користуються колонками (*Columns*). Їх налаштування складається з набору параметрів:

- Count – кількість колонок;
- Width – ширина колонки;
- Gutter – відстань між колонками;
- Margin – відступ сітки від правого і лівого краю макету.

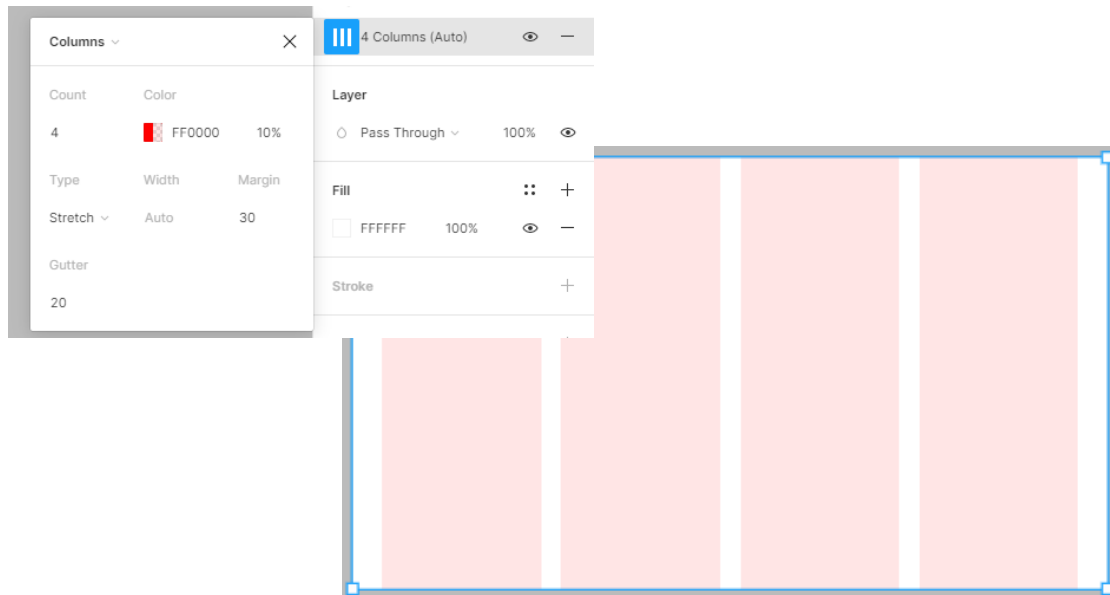


Рисунок 3.2 – Розмітка *Колонки* та параметри налаштування

Рядки (*Rows*) використовуються для розробки однотипних блоків – карток. Використання карток полегшує візуальне сканування, а контент надає достатню інформацію про об'єкт. Приклади використання: картки товарів або публікацій, треки у плейсті, список контактів або дзвінків, тощо.

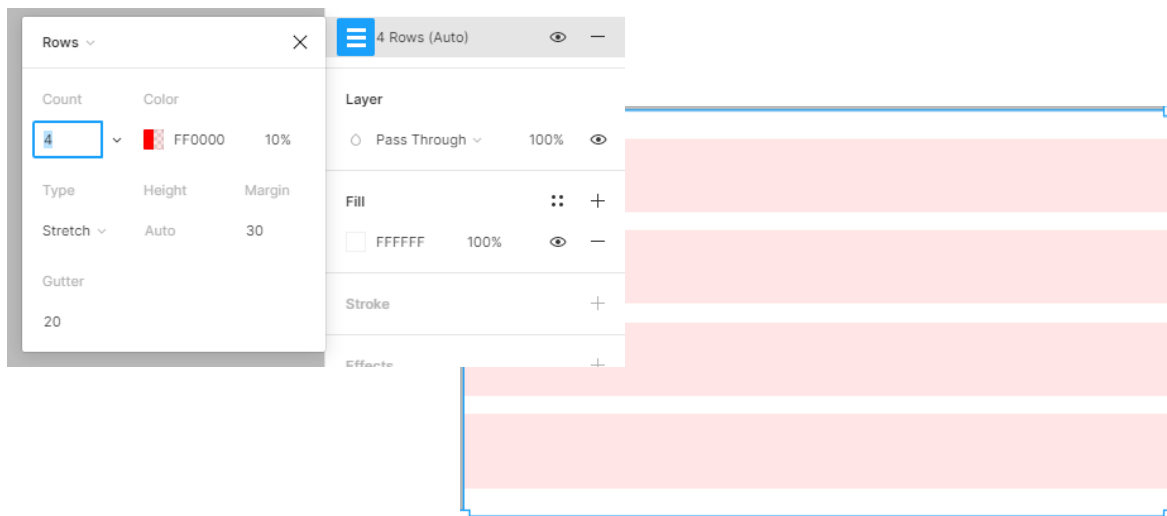


Рисунок 3.3 – Розмітка *Рядки* та параметри налаштування

Змішана або система направляючих. Використовується для складних систем, кастомних сайтів, портфоліо.

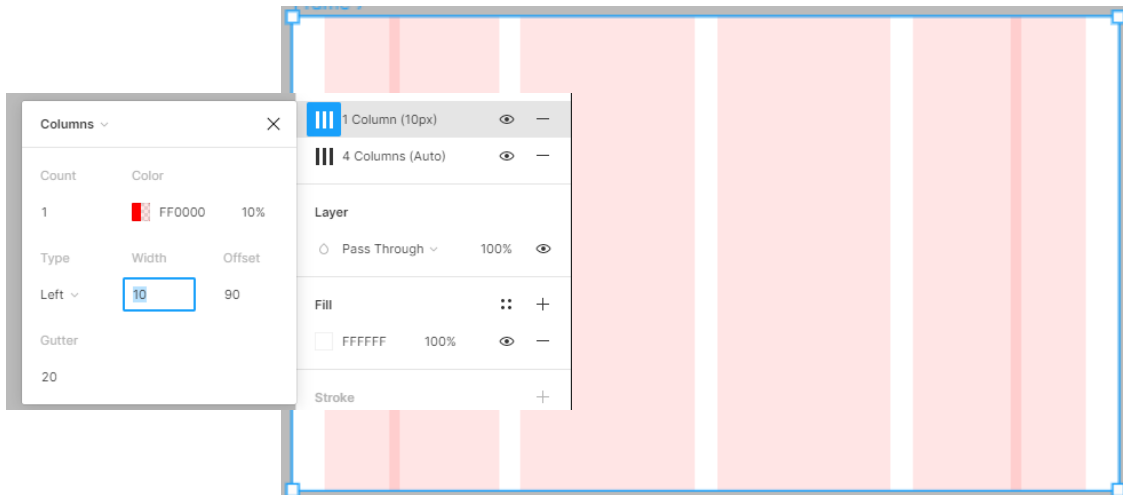
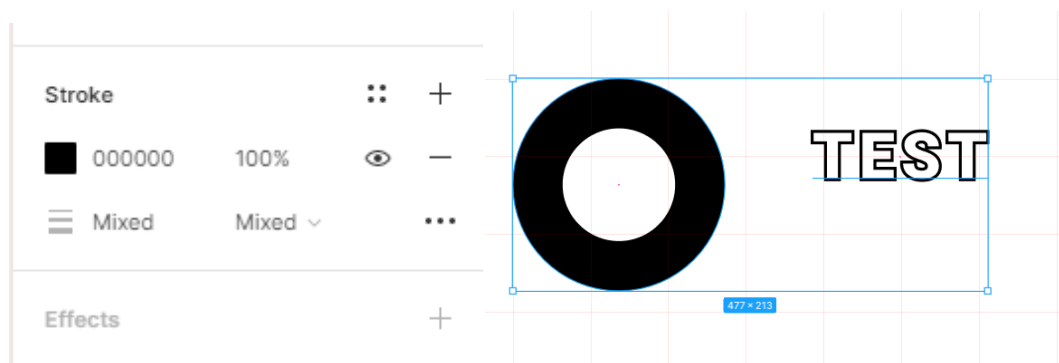


Рисунок 3.4 – Параметри налаштування змішаної розмітки

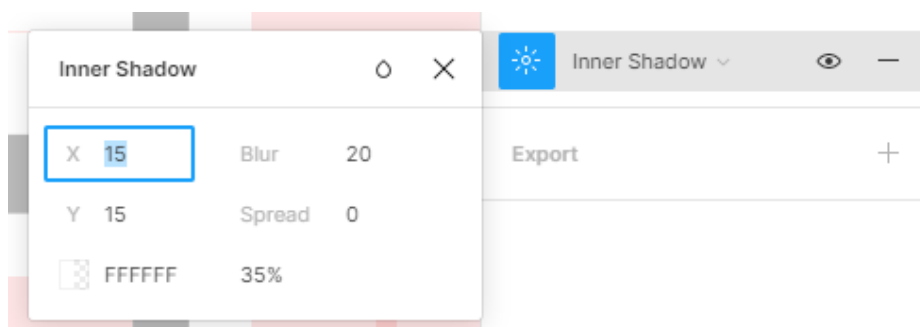
Контур (*Stroke*) – використовується для привернення уваги до елемента або в якості декоративного елемента. Задається будь-яких розмірів, кольорів, можна вибирати положення обведення (відображення в центр елемента, ззовні елемента та рівно по краю).

Рисунок 3.5 – Параметри елемента *Контур*

Ефекти

У середовищі Figma передбачені чотири ефекти: зовнішня та внутрішня тіні, розмиття фону та елемента.

Тінь (*Shadow*) ефективно використовується для надання елементам 3D ефекту або більш реалістичного вигляду.

Рисунок 3.6 – Ефект Внутрішня тінь (*Inner Shadow*)

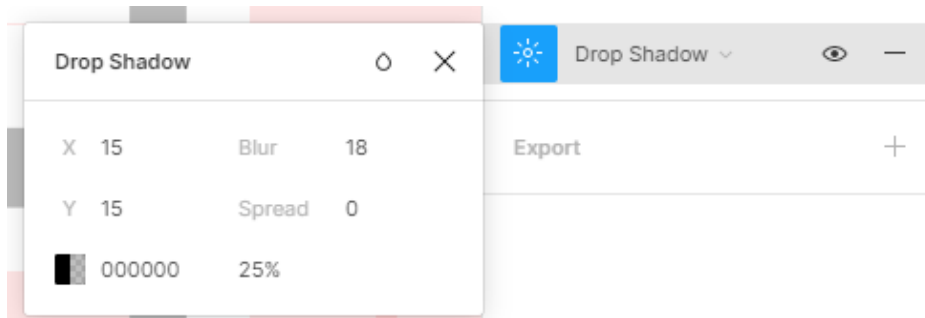


Рисунок 3.7 – Ефект Зовнішня тінь (*Drop Shadow*)



Рисунок 3.8 – Застосування тіней для створення ефекту Paper Cut Out

За допомогою додаткових параметрів можна змінювати колір, положення, розмиття та поширення тіней.

Ефект Розмиття (*Blur*) також є двох видів: розмиття елемента та розмиття всього, що знаходиться під елементом.

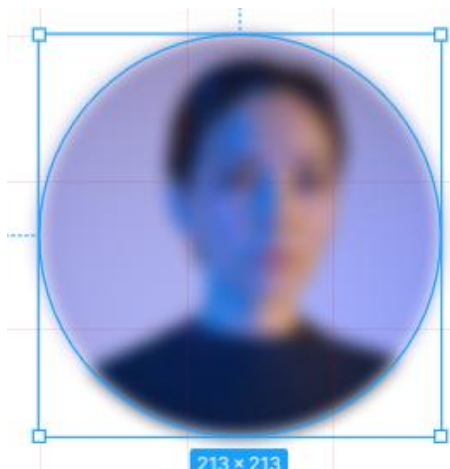
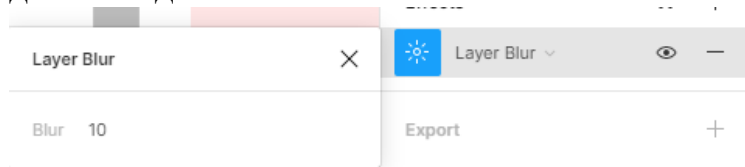


Рисунок 3.9 – Ефект Розмиття елемента (*Layer Blur*)

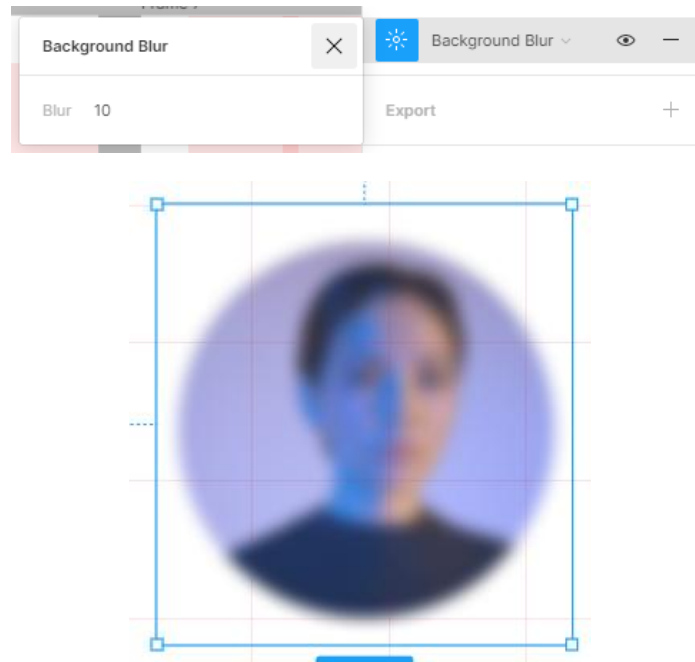


Рисунок 3.10 – Ефект Розмиття заднього фону (*Background Blur*)

Для зручної роботи у середовищі передбачена можливість створювання власних стилів, які надалі можна легко застосувати. Можна зберігати стилі сіток, кольорів, ефектів та контурів. Створення кастомного стилю робиться на основі вже здійсненого встановлення бажаних параметрів для типового об'єкта. Збереження налаштувань для типового об'єкта відбувається через меню (кнопка з чотирма крапками), командою «Create Style» та збереженням стилю під новою назвою.

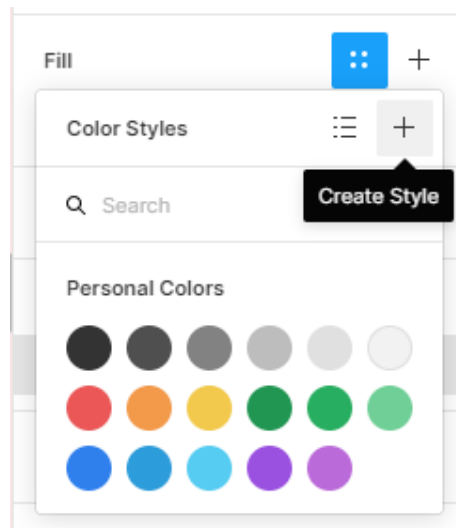


Рисунок 3.11 – Приклад створення кастомного стилю елемента

Зміст роботи

1. На базі розробок попередньої лабораторної роботи розробити дизайн інтерфейсу.
2. Оформити звіт про виконану роботу (з вказанням посилання на проект).
3. Захистити лабораторну роботу.

Варіанти завдань

Завдання відповідають раніше обраним у попередній лабораторній роботі.

Контрольні питання

1. Що таке сітка? Навіщо вона використовується?
2. Які типи сітки?
3. Застосування колонок.
4. Що таке Gutter?
5. Застосування рядків.
6. Різноманітні ефекти, доступні у Figma.

Лабораторна робота №4

Прототипування взаємодії користувача з інтерфейсом

Мета: ознайомитися з можливостями Figma у проектуванні подорожі користувача при розробці інтерфейсів.

Теоретична частина

При проектуванні інтерфейсів особливу увагу приділяють взаємодії користувача з елементами та його переходам між екранами додатку. Взаємодія між користувачем та продуктом (Interaction Design, IxD) має бути або природньою (якщо на екрані є яскрава кнопка, вона дає зрозуміти, що на неї можна натиснути), або просто звичною (наприклад, найпопулярніші месенджери мають дуже схожі інтерфейси, тож користувачеві не потрібно прикладати додаткових зусиль, щоб зрозуміти як працювати з додатком).

Для того, щоб вказати користувачеві на функції, вкладені в інтерфейс, дизайнери вдаються до анімації. Завдяки цьому ми розуміємо, що відбуватиметься на наступному кроці, або ж отримуємо естетичне задоволення від витонченої зміни екранів, більш схожої на продукт кіноіндустрії. Можливості анімації обмежуються лише уявою дизайнера.

Для анімації у Figma передбачена окрема вкладка Prototype, яку можна знайти у верхньому правому куті (рисунок 4.1). Саме за її допомогою активується режим дизайну взаємодії і проводяться всі налаштування переходів між фреймами, анімації кнопок та pop-up вікон. Анімація дозволяє підвищити зрозумілість інтерфейсу завдяки акцентуванню уваги користувача на інтерактивних елементах. Частіше за все в інтерфейсах зустрічається низка анімацій, які за допомогою візуальних властивостей елемента (пригадуємо про афорданс), демонструють зміну станів елемента при взаємодії з ним користувача.

При натисканні: означає, що взаємодія відбуватиметься, коли користувач натискає вибраний об'єкт. Якщо взаємодія відбувається на кадрі, то користувач зможе клацнути в будь-якому місці цього кадру, щоб викликати взаємодію.

Під час наведення: взаємодія розпочнеться, коли користувач наведе курсор на вибраний об'єкт або кадр, що певним чином змінить їх вигляд. Коли користувач закінчить наведення, елементи повернуться до початкового стану.

Введення миші / Вихід миші: взаємодія розпочнеться, коли миша увійде або вийде з вибраного елемента (або кадру). Дуже схоже на функцію **Під час наведення** курсора, але воно не переривається автоматично, коли наведення закінчиться. Яскравим прикладом застосування є розгортання меню та підменю.

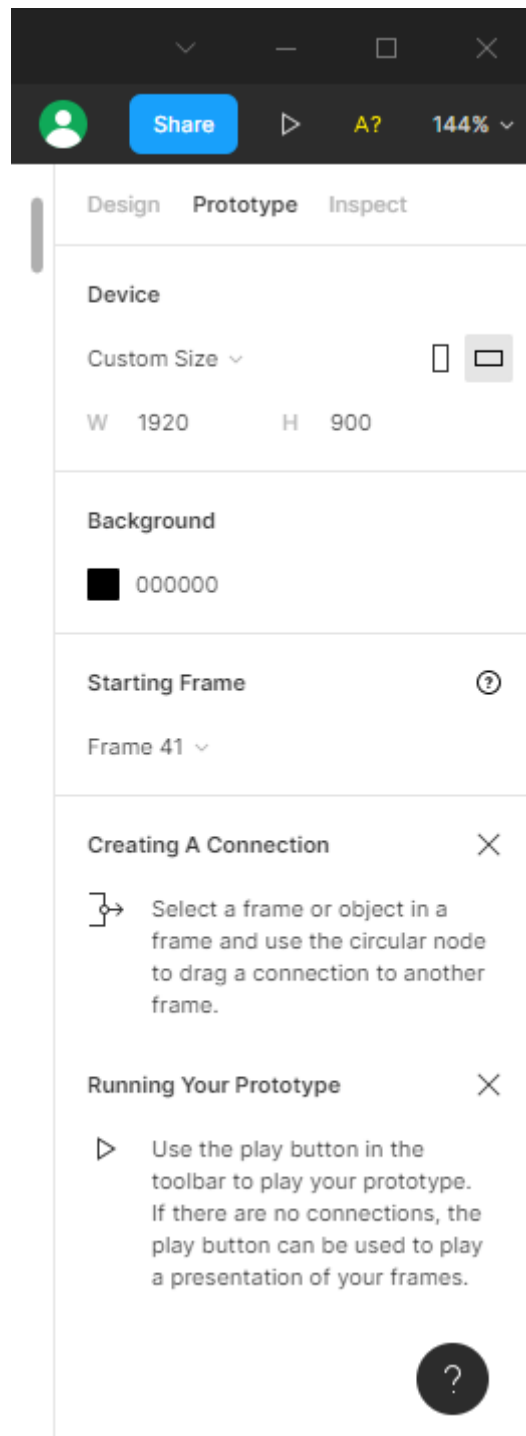


Рисунок 4.1 – Вкладка Prototype

Після затримки: ця взаємодія відбувається автоматично після закінчення зазначеного часу. Якщо затримка вказана 500 мс, взаємодія розпочнеться через 500 мс, починаючи з того моменту, коли користувач потрапив до цього кадру. Цей тип взаємодії обмежений лише кадрами, тож такий тип взаємодії не застосовується під час створення прототипів кнопок.

Перелік можливих варіантів анімації представлений на рисунку 4.2.

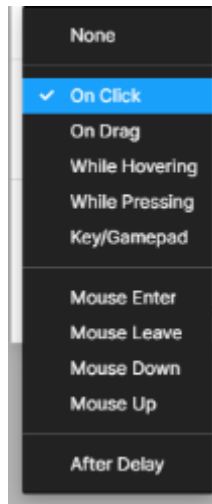


Рисунок 4.2 – Перелік варіантів анімації у Figma

Для задавання переходів від одного фрейму до іншого, запропоновані три варіанти (рисунок 4.3).

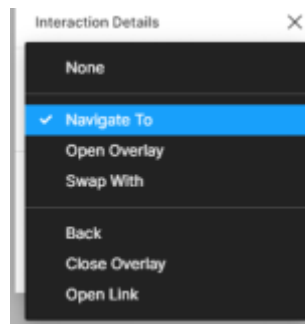


Рисунок 4.3 – Перелік переходів

Navigate To – просто перенаправляє анімацію на обраний фрейм. Іншими словами, змінює один фрейм на інший незалежно від розмірів.

Open Overlay – використовується для анімацій впливаючих вікон, програвання анімацій відкривання меню на мобільних девайсах і т.д. Open Overlay має власний перелік налаштувань, що показаний на рисунку 4.4.

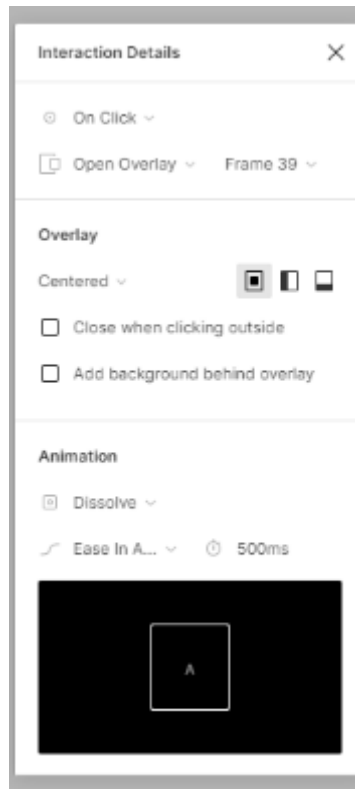


Рисунок 4.4 – Меню налаштування Open Overlay

Меню дозволяє дизайнеру вказати яким саме чином треба відкрити фрейм. Також меню дає змогу створити власне кастомне положення. Два чекбокси визначають поведінку фрейму при взаємодії користувача з визначеним елементом, або поза ним, та можливість затемнення фону за активним фреймом (оверлей).

Swap With – міняє місцями фрейми (оверлей).

Тепер поговоримо як саме задається анімація візуально (рисунок 4.5).

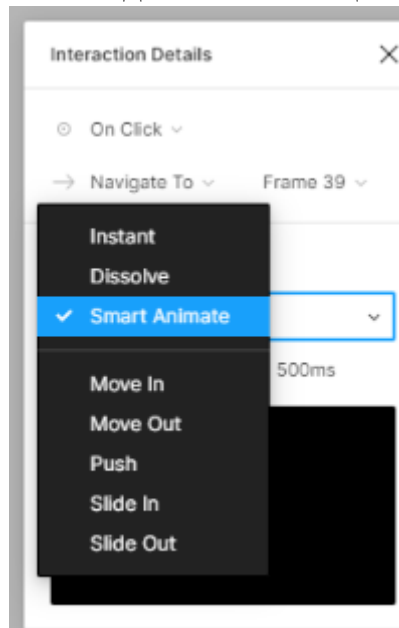


Рисунок 4.5 – Меню вибору типів анімації

Тип анімації: Figma пропонує кілька попередньо встановлених анімацій на вибір:

- *Instant* – означає, що ваш потік миттєво зміниться, анімації не буде;
- *Dissolve* – затухання між двома кадрами;
- *Move In/Out, Push* та *Slide In/Out* – загальні варіанти прототипування інтерфейсу користувача, напевно, ви знаєте їх лише із використання мобільних додатків.

Хронометраж: Керувати уповільненням для кожної анімації на Figma можна через вибір значень *Ease In*, *Ease Out* або *Ease In and Out*. Це стосується уповільнення руху в анімації. *Ease In* означає, що анімація почнеться повільно і буде прискорюватися в процесі руху, *Ease Out* – навпаки, анімація почнеться дуже швидко і буде закінчуватися повільно. *Ease In and Out* означає, що анімація почнеться повільно, прискорюватиметься, а потім в кінці сповільниться.

Smart Animate – це спосіб зв'язати два подібних кадри та отримати плавну анімацію між ними. Figma інтерполює кадри та передає бажаний рух. Перший фрейм виступає в ролі ключового кадру А, в той час як другий фрейм – ключовий кадр В. Логіка середовища Figma аналізує зміну положення активного об'єкту (який є вихідною точкою для стрілки, рисунок 4.6) та генерує всі проміжні кадри, відображаючи переміщення активного об'єкту в положення В.

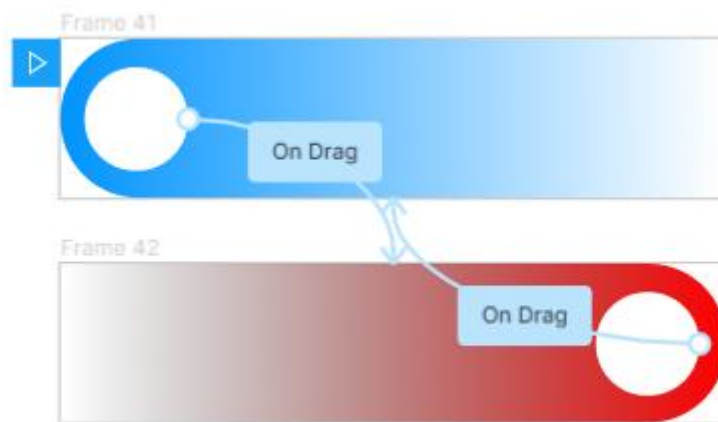


Рисунок 4.6 – Приклад переходів

Для того, щоб переглянути результати анімації, у Figma передбачений спеціальний елемент – плеєр, синій квадрат з позначкою «Play». Він встановлюється у верхньому кутку лівої грані початкового фрейму, та може легко переміщуватися від однієї групи фреймів до іншої. Запуск перегляду результатів анімації здійснюється натисканням іконки «Play» у верхньому правому кутку вікна (рисунок 4.7).

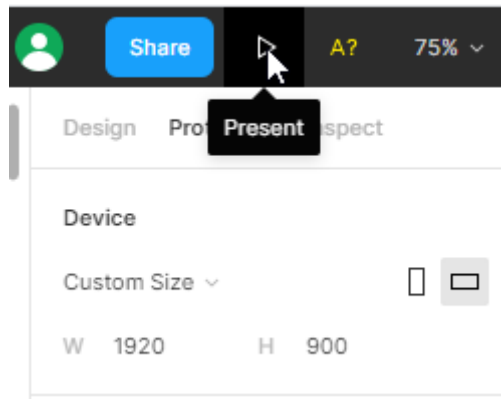


Рисунок 4.7 – Запуск перегляду анімації

Демонстрація відбувається в окремому вікні і дає можливість дизайнеру оцінити отриманий результат. Перезапуск перегляду здійснюється за допомогою кнопки «Restart» у правому нижньому кутку екрану, або клавішею R.

Зміст роботи

1. На базі розробок попередніх лабораторних робіт спроектувати взаємодію користувача з інтерфейсом.
2. Оформити звіт про виконану роботу (з вказанням посилання на проект).
3. Захистити лабораторну роботу.

Варіанти завдань

Завдання відповідають раніше обраним у лабораторній роботі №2.

Контрольні питання

1. Що таке IxD?
2. Яким чином налаштовуються переходи?
3. Які типи переходів доступні?
4. Які типи маніпуляцій передбачені у середовищі?
5. Які можливості анімації Smart Animate?
6. Яким чином налаштовують швидкість переходів?

Лабораторна робота №5

Адаптивні блоки на базі функції Auto Layout

Мета: ознайомитися з можливостями блокової адаптації контенту з використанням Auto Layout функції.

Теоретична частина

Auto Layout – функція Figma, що дозволяє динамічно розрізняти шари у фреймах. Наприклад, при налаштуванні відступів, поведінки та адаптивності у компонента. Auto Layout спрощує роботу дизайнера та суттєво скорочує час розробки.

Типовим зразком застосування Auto Layout є дизайн типового та розповсюдженого елемента будь-якого інтерфейсу – кнопки.

Під час роботи над макетом часто доводиться змінювати вигляд кнопки, і зробити це без використання компонент та функції Auto Layout дуже проблематично. Наприклад, макет складається із 20-30 сторінок, і у кожен з них доводиться вносити правки у вигляді кнопки: змінювати розмір, внутрішні відступи, ін., – що приводить до повторюваних дій та часових (а отже і фінансових) витрат.

Створення Auto Layout передбачає роботу з одним або декількома елементами. Для проектування кнопки складовими будуть прямокутник синього кольору із заокругленими кутами та текстовий блок (рисунок 5.1).

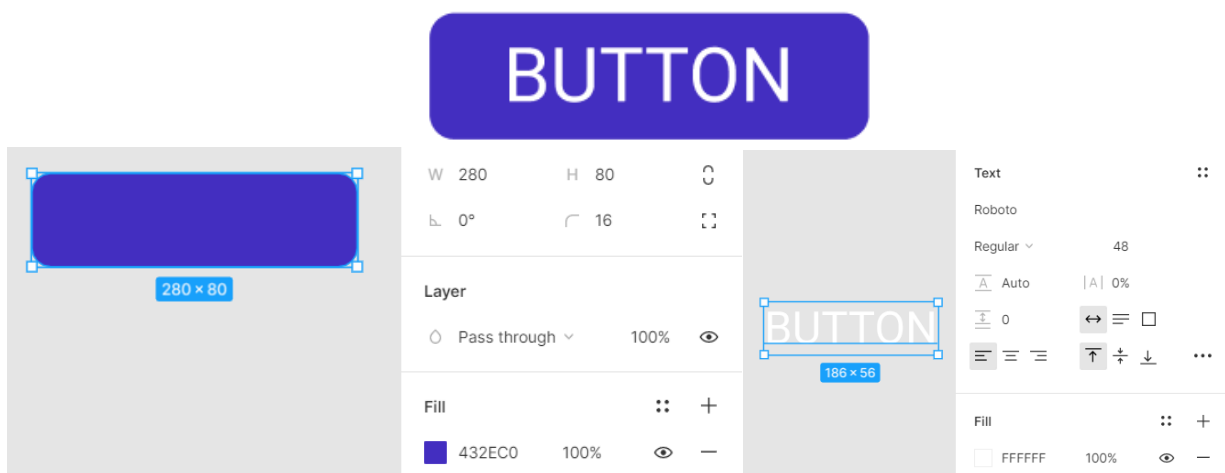


Рисунок 5.1 – Базова кнопка та складові

Для того, щоб об'єднати елементи в Auto Layout, їх потрібно виділити та об'єднати за допомогою Shift+A або кнопкою Auto Layout на панелі інструментів.

В панелі можна вказати горизонтальне або вертикальне розміщення елементів, відстань між ними, внутрішній відступ (для створення карток) та розташування елемента всередині лейаута.

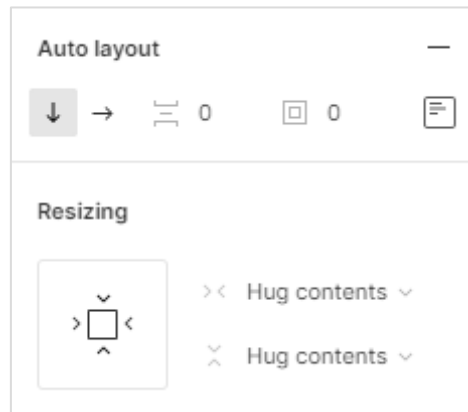


Рисунок 5.2 – Панель налаштування властивостей Auto Layout

На прикладі синьої кнопки властивості Auto Layout впливають на кожну зі складових:

Hug content – синій прямокутник приймає мінімально можливий розмір, що продиктований розміром текстового блоку та відступами.

Fill container – прямокутник заповнить весь простір контейнера, залишив вільним простір вказаних відступів.

Fixed – розмір прямокутника-тіла кнопки не змінюється.

Розташування дочірніх елементів в батьківському контейнері також можна вказати набором параметрів. Так, множину елементів можна вирівнювати за заздалегідь вказаною стороною (варіанти **Packed**) або розмістити рівномірно у просторі контейнеру (**Space between**).

Розташування текстового блоку задається набором властивостей, як вказано на рисунку 5.3.

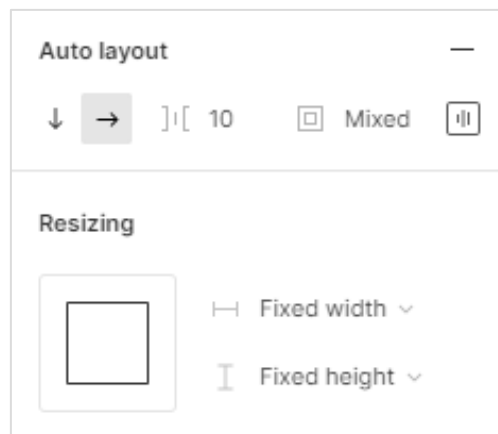


Рисунок 5.3 – Налаштування параметрів текстового блоку

Отже, розглянемо створення лейауту на прикладі кнопки, приведеної на рисунку 5.1. Перше – умова, за якої текстовий блок буде автоматично вирівнюватися за центром незалежно від розміру кнопки.

Реалізація функції Auto Layout використовує таку особливість як вкладеність. Зазвичай її використовують для створення карток, списків, графіків, хедерів, футерів, тощо. Розглянемо вкладеність на прикладі списку. Рядок списку складається із зображення та текстового блоку. Створимо

автолейаут, а потім продублюємо декілька разів. В результаті отримуємо спільний лейаут, що складається з п'яти рядкових лейаутів (рисунок 5.4).

- + Lorem ipsum dolor sit amet, consectetur adipiscing elit.
- + Lorem ipsum dolor sit amet, consectetur adipiscing elit.
- + Lorem ipsum dolor sit amet, consectetur adipiscing elit.
- + Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Рисунок 5.4 – Первинний список

Припустимо, пункти 1 та 4 містять вкладені списки. Виділимо їх з-поміж пунктів первинного списку за допомогою іншої іконки та відступу зліва (20 пікселів).

Для цього треба продублювати один з внутрішніх лейаутів та встановити наступні налаштування:

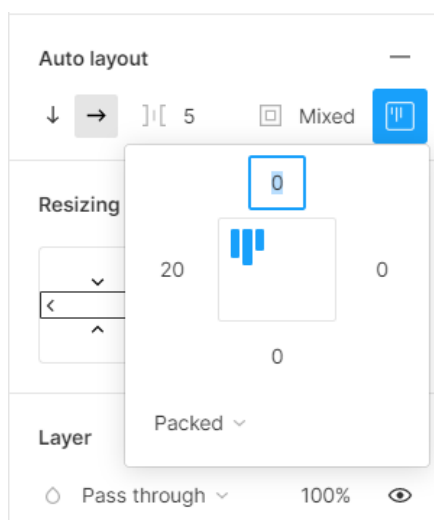


Рисунок 5.5 – Встановлення відступів вкладеного списку

Останній крок – замінити іконку пункту вкладеного списку. На базі створеного списку можна сформувати картку. Також доповнимо внутрішній відступ для контейнера та визначимося із фоном.

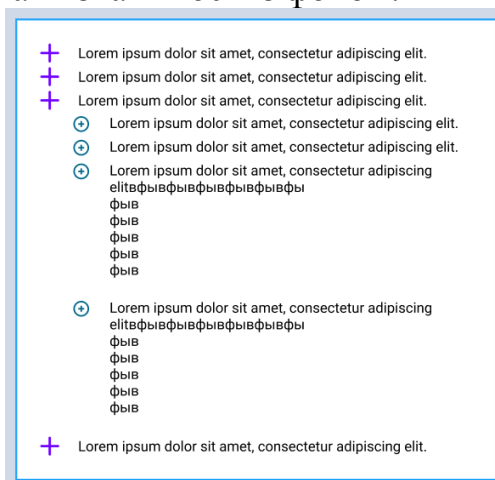


Рисунок 5.6 – Готовий список

Зміст роботи

1. Виконати Завдання 1 та Завдання 2. Картка та графік мають відповідати тематиці варіанту попередньої лабораторної роботи.
2. Оформити звіт про виконану роботу (з вказанням посилання на проект).
3. Захистити лабораторну роботу.

Варіанти завдань

1. **Завдання 1.** На базі розробок попередніх лабораторних робіт створити карточку, яка буде змінюватися при горизонтальному розтягуванні. Картка повинна містити: зображення (фото), текст та іконка в одному рядку. Наприклад, текст з відступом з лівого боку 10 пікселів, іконка з відступом справа 10 пікселів. Далі один великий текстовий блок, що буде реагувати на зміну ширини картки. Внизу повинна бути кнопка. Приклад картки наведено нижче, на рисунку 5.7.

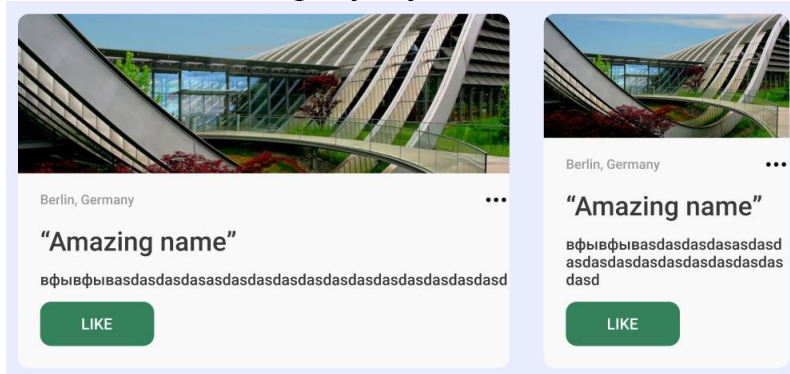


Рисунок 5.7 – Приклад картки

2. **Завдання 2.** Створити масштабовану картку графіка. Приклад складових наведений на рисунку 5.8.



Рисунок 5.8 – Приклад виконання завдання

Контрольні питання

1. Що дозволяє автоматизувати функція Auto Layout?
2. Яка головна особливість фреймів, що забезпечує створення Auto Layout?

Лабораторна робота №6

Структура презентації дизайн-рішень

Мета: набути навички демонстрації проекту та створення презентації розробленого дизайну

Теоретична частина

Попри уміння проектувати зручний інтерфейс, необхідно уміти прорекламувати результат власних зусиль. Інакше кажучи, будь-який продукт потребує презентації потенційному замовникові або клієнтові.

Структура презентації

Навчитися створювати дійсно ефективні презентації можна лише завдяки навчанню у кращих. Саме тому вивчайте презентації успішних дизайнерів: звертайте увагу на структуру – матеріал будується за схожим принципом – зав'язка, розвиток дії і розв'язка. Нагадує скоріше драму або літературний твір, але ми будемо називати це сторітеллінгом. Ми розповідаємо замовнику історію: починаємо з проблеми і закінчуємо тим, як її вирішили.

Хороший варіант, який підходить для більшості презентацій, складається з наступних кроків:

- 1) опис проекту та головного завдання;
- 2) прототип та начерки;
- 3) головна сторінка;
- 4) другорядні сторінки;
- 5) мобільна версія (за наявності);
- 6) результати реалізації (мокап, фото, анімація).

Досвідчені рекомендують створювати презентації не коротше 20 000 пікселів. Дотримання цього правила дозволить у повній мірі розкрити суть проекту.

Так як вся взаємодія орієнтована на викликання у споглядача позитивних емоцій, ця функція покладена на перший екран.

Перший екран – визначальна емоція

Перший екран задає весь тон презентації, говорить про стилізацію, емоції всього проекту. Це перша точка емоційного контакту кураторів і глядачів з проектом, тому зробіть цей контакт вражаючим, щоб збільшити лояльність до проекту. Використовуйте графіку, фотографії, анімації, найбільш вдалі екрани інтерфейсу. Мета – захопити глядача і мотивувати дивитися далі.

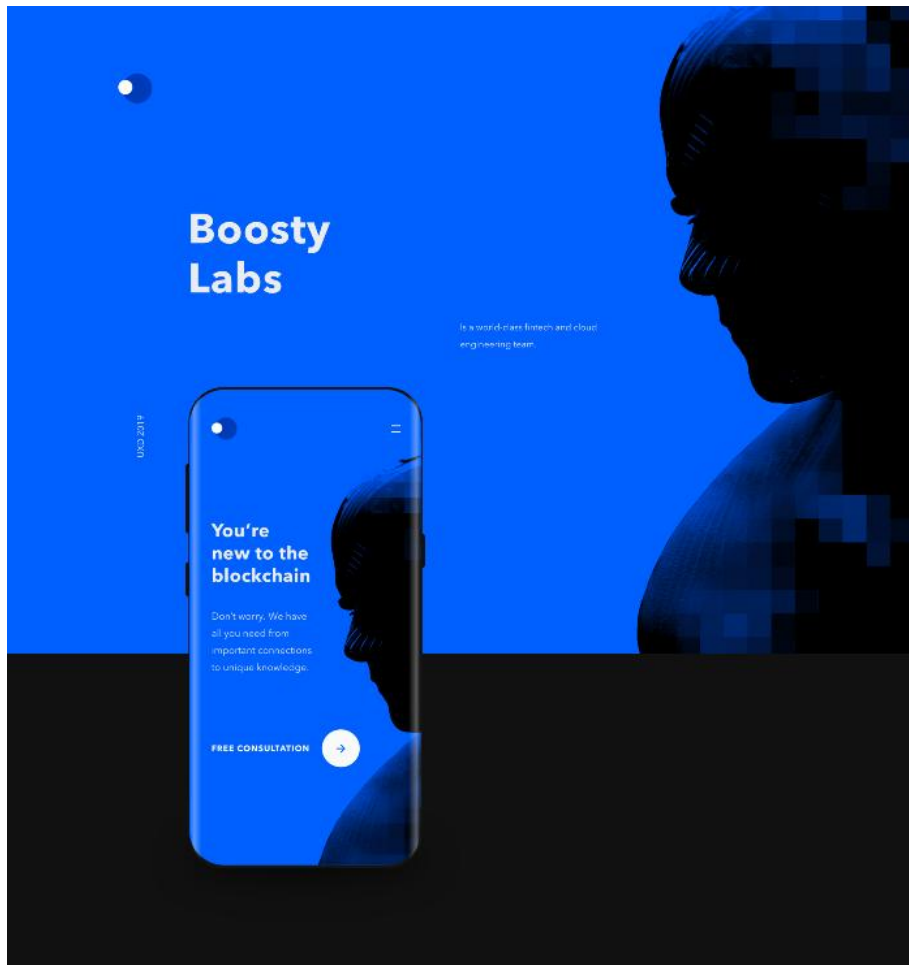


Рисунок 6.1 – Приклад виконання першого екрану

UX-обґрунтованість візуального рішення

Для галереї Взаємодія (Interaction) вкрай важливо показати UX кроки. Яке завдання бізнесу ви вирішували? Яка потреба користувачів? Які етапи побудови логіки пройшли – інтерв'ю, персонажі, sitemap, customer journey map (cjm), userflow, прототипи. Опишіть і покажіть будь-які інші кроки, які допомогли зробити вам зручний продукт і підтвердіть результат в цифрах.

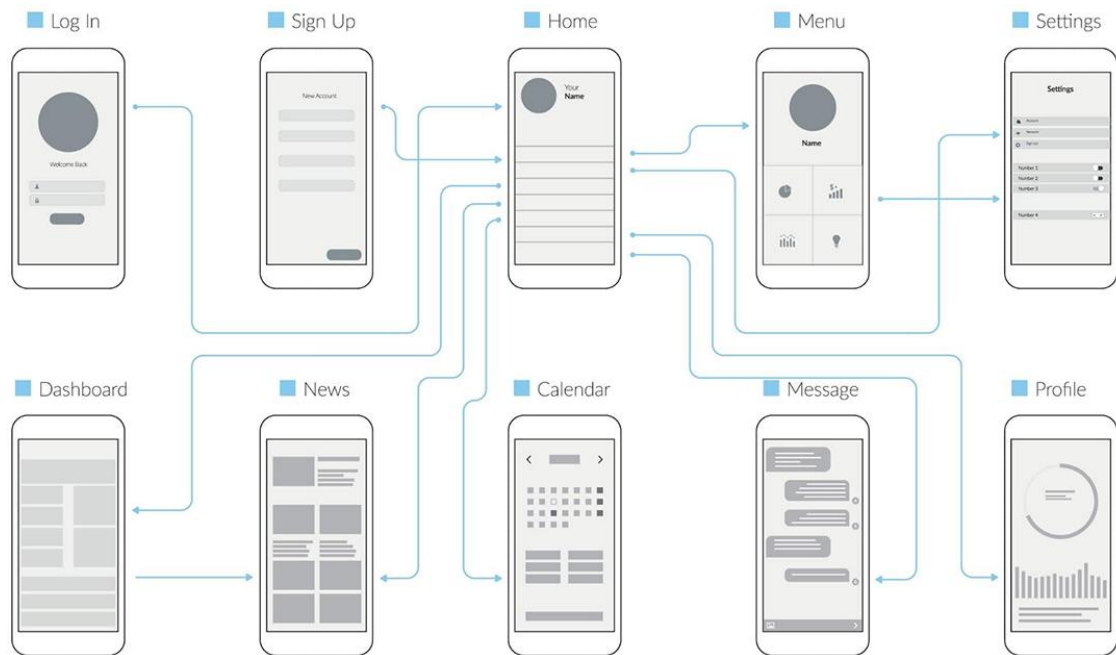


Рисунок 6.2 – Приклад User Flow діаграми для мобільного додатку

Сітка і pixel perfect

Сітка лежить в основі будь-якого проекту і може розповісти про ваш художній задум (прогресивні сітки або золотий перетин), а також про вірність IOS і Android керівництвом. Покажіть що ваш проект не тільки результат дослідження і творчості, а й продукт, придатний для технічної реалізації.



Рисунок 6.3 – Приклад використання сітки у візуальному дизайні

Анімація

Презентація, в якій є трохи анімації, залучає і захоплює глядачів сильніше. Не обов'язково створювати складні анімації, досить анімувати легкі ділянки за допомогою Figma.

Останній екран

Значення останнього екрану теж виведено досвідченим шляхом. Проекти, які залишають після себе приємні емоції, мають більше шансів на успіх. Можна вразити глядача останнім *tasty pieces*, можна побажати гарного настрою, можна залишити жарт або маленьку анімацію сердечка. Будь-яка ідея повинна бути спрямована на те, щоб глядачеві стало добре і він залишив позитивний відгук.

Довжина презентації

У проектах бездоганної якості зазвичай 6 екранів. Якщо ви відчуваєте, що ваша робота не така, значить потрібно показати мінімум один призначений для користувача шлях. Наприклад, покажіть в інтернет-магазині головну сторінку, каталог, картку товару, кошик, процес оплати товару та супровідні модальні вікна. Виходить процес від попадання на сайт до покупки – це один призначений для користувача шлях. Не забувайте знаходити ділянки для *tasty pieces*. Ними можуть стати банери, блок підписки, знижки та інші маркетингові доповнення.

Зміст роботи

1. На базі розробок попередніх лабораторних робіт розробити презентацію проекту.
2. Оформити звіт про виконану роботу (з вказанням посилання на проект).
3. Захистити лабораторну роботу.

Варіанти завдань

Завдання відповідають раніше обраним у попередній лабораторній роботі.

Контрольні питання

1. Як створити кастомну сітку? З яких компонентів вона складається?
2. Чи можна робити якірні посилання у макеті? Якщо можна, то яким чином?
3. Із чого складається структура презентації веб-дизайн проекту?
4. Що таке сторітеллінг, як він використовується в презентації веб-дизайн проектів?

Рекомендовані джерела

1. Grady Booch, James Rumbaugh, Ivar Jacobson. Unified Modeling Language Reference Manual. Addison-Wesley.
2. Коберн, А. Современные методы описания функциональных требований к системам / А. Коберн. - М.: Лори, 2002. https://biconsult.ru/img/bi_portal/Alister_Kobern_sovremennie_trebovaniya.pdf
3. Figma. <https://www.figma.com/>
4. UX Planet. <https://uxplanet.org/>
5. Interaction Design Foundation. <https://www.interaction-design.org/>
6. Reimann, Robert & Cooper, Alan & Cronin, David & Noessel, Chris. (2014). About Face: The Essentials of Interaction Design, 4th Edition. – Available at: https://fall14se.files.wordpress.com/2017/04/alan-cooper-robert-reimann-david-cronin-christopher-noessel-about-face_the-essentials-of-interaction-design-wiley-2014.pdf
7. Якоб Нільсен, Ралука Будіу Mobile Usability. Як створювати ідеально зручні програми для мобільних пристроїв, 2013. — 213 с.