

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ  
Навчально-науковий інститут інноваційних освітніх технологій  
Кафедра комп'ютерних мультимедійних технологій

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач кафедри

\_\_\_\_\_ С.М. Лобода

“ \_\_\_\_ ” \_\_\_\_\_ 20\_\_ р.

# ДИПЛОМНА РОБОТА

(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ МАГІСТР  
ЗА ОСВІТНЬО-ПРОФЕСІЙНОЮ ПРОГРАМОЮ “ТЕХНОЛОГІЇ ЕЛЕКТРОННИХ  
МУЛЬТИМЕДІЙНИХ ВИДАНЬ”

**Тема: “Інформаційно-технологічні особливості проєктування інтерфейсу  
державної установи”**

Виконавець: студент групи ВП-201Мз Супрун Ілля Анатолійович

Керівник: доктор педагогічних наук, професор Лобода Світлана Миколаївна

Нормоконтролер: \_\_\_\_\_ ст. викладач Таран Віктор Миколайович

КИЇВ 2020

# НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Інститут інноваційних технологій

Кафедра комп'ютерних мультимедійних технологій

Спеціальність: 186 “Видавництво та поліграфія”

Освітньо-професійна програма “Технології електронних мультимедійних видань”

ЗАТВЕРДЖУЮ

Завідувач кафедри

\_\_\_\_\_ Лобода С.М.

“ \_\_\_\_ ” \_\_\_\_\_ 2020 р.

## ЗАВДАННЯ

### на виконання дипломної роботи

Супруна Іллі Анатолійовича

1. Тема роботи “ Інформаційно-технологічні особливості проектування інтерфейсу державної установи ” затверджена наказом ректора від “...”р.
2. Термін виконання роботи:
3. Вихідні дані роботи: Технічне завдання на проектування інтерфейсу веб-ресурсу Київського міського центру громадського здоров'я.
4. Зміст пояснювальної записки:
  - Розділ 1.
  - Розділ 2.
  - Розділ 3.
5. Перелік обов'язкового ілюстративного матеріалу: таблиці, фотографії, ілюстрації, презентація PowerPoint.

## 6. Календарний план-графік

№ з/п	Завдання	Термін виконання	Підпис керівника
1	Отримання завдання до виконання дипломної роботи	05.10.2020 – 10.10.2020	Виконано
2	Ознайомлення з постановкою задачі та аналіз предметної області.	11.10.2020 – 19.10.2020	Виконано
3	Підбір та аналіз наявних літературних джерел за темою	20.10.2020– 27.10.2020	Виконано
4	Аналіз існуючих інтерфейсів державних установ	28.10.2020 – 12.11.2020	Виконано
5	Проектування wireframes та реалізація інтерфейсу	13.11.2020 – 29.11.2020	Виконано
6	Оформлення, редагування та друк пояснювальної записки	30.11.2020 – 05.12.2020	Виконано
7	Проходження нормоконтролю	06.11.2020	Виконано
8	Підготовка презентації та доповіді.	07.11.2020 – 13.11.2020	Виконано
9	Отримання відгуку керівника, рецензії.	14.11.2020 – 22.11.2020	Виконано
10	Захист дипломної роботи.	23.12.20	Виконано

7. Дата видачі завдання: “\_\_” 2020 р.

Керівник дипломної роботи : \_\_\_\_\_  
(підпис керівника) (П.І.Б.)

Завдання прийняв до виконання: \_\_\_\_\_  
(підпис випускника) (П.І.Б.)

## РЕФЕРАТ

Пояснювальна записка до дипломної роботи “ Інформаційно-технологічні особливості проєктування інтерфейсу державної установи ”: \_\_ сторінки, рисунок, таблиць, використаних джерела.

ПРОЄКТУВАННЯ, ВЕБ-РЕСУРС, ІНТЕРФЕЙС, USER-FLOW, DESIGN GUIDELINES, COMPONENTS, UI KIT, FIGMA, MINDMASTER, INVISION

**Мета дипломної роботи** – спроектувати інтерфейс веб-ресурсу "Київський міський центр громадського здоров'я".

**Об'єкт дослідження** – технологічні особливості процесу проєктування інтерфейсу державної установи.

**Предмет дослідження** – складові інтерфейсу веб-ресурсу державної установи "Київський міський центр громадського здоров'я"

**Методи дослідження:** – теоретичні – аналіз, порівняння, систематизація та узагальнення наукової літератури, електронних ресурсів для з'ясування технології проєктування інтерфейсу веб-ресурсу; методи системного аналізу для визначення видо-типологічної класифікації та принципів проєктування інтерфейсу веб-ресурсів; порівняльний аналіз для вибору програмного забезпечення; емпіричні – тестування для аналізу зручності користування веб-ресурсом.

**Програмні засоби** – кросс-платформенний онлайн сервіс для проєктування інтерфейсів Figma, графічний редактор Adobe Photoshop CC , кросс-платформенний онлайн сервіс побудови user-flow MindMaster, кросс-платформенний онлайн сервіс прототипування InVision.

**Рекомендації щодо використання результатів:** результати магістерської роботи рекомендується використовувати як готовий проєкт для компанії Bits Digital разом з Київським міським центром громадського здоров'я у якості матеріалів для розробки веб-ресурсу державної установи.

# ЗМІСТ

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

### ВСТУП

#### Розділ 1. Теоретичні засади створення інтерфейсу

1.1. Класифікація інтерфейсів

1.2. Принципи проєктування інтерфейсу

1.3. Специфіка проєктування інтерфейсу веб-ресурсу державної установи

1.4. Порівняльний аналіз інтерфейсів державних веб-ресурсів

Висновки до розділу

#### Розділ 2. Технології проєктування інтерфейсу

2.1. Програмні засоби для проєктування інтерфейсу

2.2. Принципи побудови архітектури веб-ресурсу та user-flow

2.3. Wireframe як основа якісного UX

2.4. Етапи технології проєктування інтерфейсу

Висновки до розділу

#### Розділ 3. Практична реалізація проєктування інтерфейсу "Київський міський центр громадського здоров'я"

3.1. Дослідження та аналітика потреб користувачів веб-ресурсу

3.2. Проєктування wireframes на основі аналітичних даних

3.3. Розробка дизайну інтерфейсу на основі wireframes

Висновки до розділу

### ВИСНОВКИ

### СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

## **ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ**

## ВСТУП

**Актуальність теми.** В умовах сьогодення актуальною проблемою є швидкість доступу до інформації, а також перевірка даної інформації на достовірність. В часи диджиталізації основним джерелом інформації є інтернет, проте вітчизняні державні веб-ресурси розміщені в мережі знаходяться в занедбаному стані, через технології та підхід до їх створення. Проте вони є важливим ланцюжком в комп'ютеризації суспільства, адже в першу чергу вони виконують інформативні функції та комунікативні, а також дозволяють запровадити проекти що дозволяють висвітлювати найважливішу інформацію, всім користувачам не залежно від їх професійної діяльності.

Все більшої актуальності набувають методологічні, загальнонаукові і технологічні питання пов'язані з проектуванням інтерфейсів, та процесами ефективного використання всіх можливостей інтерфейсу для організації найефективнішого способу поширення та прийняття інформації.

Підходи до проектування інтерфейсів, а саме етапи, відповідність специфікаціям, організація проекту описані в книгах А.Купера "Інтерфейс", Д.Норман "Дизайн простих речей" та інших.

Проблеми проектування досвіду використання розкривається книгах у Р.Унгер "UX-дизайн. Практическое руководство по проектированию опыта взаимодействия", Г.Кришна "Хороший интерфейс — невидимый интерфейс" та інших, яка є найважливішим фактором для користувачів, оскільки сприяє ефективному пошуку та сприянню інформації.

**Об'єкт дослідження** – технологічні особливості процесу проектування інтерфейсу державної установи.

**Предмет дослідження** – складові інтерфейсу веб-ресурсу державної установи "Київський міський центр громадського здоров'я"

**Мета дипломного проекту** – полягає у визначенні та обґрунтуванні методів та засобів проектування інтерфейсу державної установи "Київський міський центр громадського здоров'я", та його практична реалізація.

### **Завдання дипломної роботи:**

- 1) Дослідити класифікацію та принципи проектування інтерфейсу;
- 2) Здійснити аналіз веб-ресурсів державних установ, розглянути їх структуру і компоненти;
- 3) Визначити специфіки проектування інтерфейсу державних установ;
- 4) Проаналізувати сучасне програмне забезпечення проектування інтерфейсів;
- 5) Розробити концепцію та практично реалізувати інтерфейс державної установи "Київський міський центр громадського здоров'я".

**Методи дослідження:** – *теоретичні загальнонаукові:* аналіз науково-технічної літератури з проблеми дослідження, ресурсів Інтернет з метою виділення теоретичних засад дослідження; метод структурного аналізу - для визначення структури веб-ресурсів і їх структурних компонентів; порівняльний аналіз- для порівняння веб-ресурсів структурних підрозділів вітчизняних державних установ; – *специфічні наукові:* системний аналіз і проектування - для розробки інтерфейсу державної установи "Київський міський центр громадського здоров'я" шляхом розчленування проблеми на частини (етапи); метод макетування для розробки інтерфейсу; метод порівняння отриманих теоретичних знань програмних засобів для створення інтерфейсу державної установи; *емпіричні* – тестування для аналізу зручності користування веб-ресурсом.

**Наукова новизна:** полягає в тому, що вперше було розроблено інтерфейс державної установи "Київський міський центр громадського здоров'я" з підтримкою всієї структурних і функціональних компонентів необхідних державній установі охорони здоров'я.

**Практична значимість:** полягає у комплексному дослідженні розробки інтерфейсу державної установи, як засобу для розширення можливостей доступу до інформації, та її обміну, також вдосконалення технологій процесів проектування інтерфейсу державної установи, та може бути використано при проведенні практичних занять зі студентами, для поглибленого вивчення курсу розробки та проектування інтерфейсу.



**Результати магістерської роботи впроваджено** у роботі підприємства Bits Digital, про що свідчить довідка про впровадження №

## РОЗДІЛ 1. ТЕОРЕТИЧНІ ЗАСАДИ СТВОРЕННЯ ІНТЕРФЕЙСУ

### 1.1 Класифікація інтерфейсів

Однією з форм користувацького інтерфейсу, яка дозволяє користувачам взаємодіяти з електронними пристроями за допомогою графічних іконок і звукових індикаторів, замість текстових призначених для користувача інтерфейсів, набраних командних міток або навігації по тексту є графічний користувацький інтерфейс. Для вивчення спроможностей інтерфейсів командного рядка, що вимагає введення команд на клавіатурі комп'ютера були введені графічні інтерфейси.

У графічному інтерфейсі всі дії, як правило, виконуються за допомогою прямого маніпулювання графічними елементами. Графічні інтерфейси використовуються в багатьох портативних мобільних пристроях, таких як MP3-плеєри, портативні медіаплеєри, ігрові пристрої, смартфони, а також невеликі пристрої офісного, промислового та побутового управління.

Інтерфейс - це, по-перше, ознак та правил і які можна узагальнити об'єднати за спільною ознакою. Тобто, поняття "вид інтерфейсу" це як об'єднання за схожістю способів взаємодії людини і комп'ютера. Тобто можливо запропонувати наступну схематичну класифікацію різних інтерфейсів спілкування людини з комп'ютером.

Сучасними видами інтерфейсів є:

1) Командний інтерфейс. У цьому інтерфейсі користувач, щоб отримати результат повинна подавати "команди" комп'ютеру, після чого комп'ютер їх виконує та видає результат людині. Командний інтерфейс реалізується у вигляді технології командного рядка та пакетної технології.

2) WIMP - інтерфейс (Window - вікно, Image - образ, Menu - меню, Pointer - покажчик). Характерною ознакою цього виду інтерфейсу є те, що спілкування з користувачем ведеться за допомогою графічних образів - меню, вікон, інших елементів, а не за допомогою команд. Хоча і в ньому подаються команди машині, але це робиться "замасковано", через графічні образи. Цей вид інтерфейсу реалізується

на двох рівнях технологій: простий графічний інтерфейс і "чистий" WIMP - інтерфейс.

3) SILK - інтерфейс (Speech - мова, Image - образ, Language - мова, Knowledge - знання). Цей вид інтерфейсу є найбільш наближеним до звичайної, людської форми спілкування. У ньому відбувається звичайна "розмова" людини з комп'ютером. При цьому комп'ютер аналізує людську мову і знаходячи в ній ключові фрази отримує для себе команди. Після чого, результат виконання команд він перетворює в зрозумілу людині форму. Цей вид інтерфейсу є найбільш вимогливим до апаратних ресурсів комп'ютера, і тому він застосовується в основному для військових цілей.

Правила взаємодії операційної системи з користувачами, а також сусідніх рівнів в мережі ЕОМ і є інтерфейс. Від нього залежить технологія спілкування людини з комп'ютером.

При цьому унітарним способом введення інформації від людини до комп'ютера служить клавіатура, а комп'ютер виводить інформацію людині за допомогою алфавітно-цифрового дисплея (монітора). Комбінація монітора з клавіатурою називається терміналом, або консоллю.

Набір команд які будуть виконуватись набираються в командний рядок, який представляє собою текстове поле, в якому будуть виконані набрані команди. Букви відображаються на дисплеї, а неправильно набрані команди або символи можна виправити. Команда завершується натисканням клавіші Enter Після цього відбувається перехід на початок наступного рядка. Після цих дій комп'ютер видає результати на монітор. Після чого процес спілкування повторюється.

Така технологія командного рядка працювала на монохромних алфавітно-цифрових дисплеях. Але технічні характеристики дисплея були не значні, тому що вводити дозволялося тільки букви, цифри і розділові знаки. У якості монітора можна було використовувати телевізійний приймач і навіть трубку осцилографа.

Текстові файли стали переважним видом при роботі з командним інтерфейсом, оскільки за допомогою клавіатури можна було створити тільки їх.

Спочатку графічний інтерфейс був дуже схожий на технологію командного рядка. Відмінності полягали в наступному.

При відображенні символів допускалося виділення їх частини кольором, підкресленням і мерехтінням та інверсним зображенням. Внаслідок чого підвищилася виразність зображення.

Залежно від конкретного використання графічного інтерфейсу курсор може представлятися не тільки мерехтливим прямокутником, але і ділянкою, що охоплює кілька символів і частину екрану. Ця виділена ділянка відрізняється від інших, невиділених частин (як правило, кольором).

Задля швидкого виділення потрібної частини екрану і переміщення курсору, в редакції графічного інтерфейсу стали використовуватися маніпулятори. Тобто щоб управляти курсором на клавіатурі стали крписуватися не тільки клавішою Enter а й маніпуляторами.

Інтерфейс WIMP став другим етапом у розвитку графічного інтерфейсу. Він характеризується наступними особливостями.

У вікнах – відповідних, окреслених рамкою частинах екрану, відбувається робота з програмами, файлами та документами.

Усі об'єкти, програми, файли, документи, та інші пристрої подаються у вигляді значків – іконок, які при відкритті перетворюються на вікна.

За допомогою меню здійснюються відповідні дії з об'єктами. На першому етапі становлення графічного інтерфейсу меню не мало головного значення, а лише служило доповненням до командного рядка. У чистому WIMP - інтерфейсі воно стає основним елементом управління.

Таким чином маніпулятор перестає бути доповненням до клавіатури і стає одним з елементів керування. За його допомогою з'являється можливість вказувати на будь-яку область іконки, вікна чи екрану. Після цього з'являється можливість виділити її, і далі здійснюють управління ними через меню або з використанням інших технологій.

Слід зазначити, що для реалізації WIMP потрібно кольоровий растровий дисплей з високим розрішенням та маніпулятор. А також програми, що орієнтуються на цей вид інтерфейсу і пред'являють підвищені вимоги до продуктивності комп'ютера, обсягу його пам'яті і т.п. Такий вид інтерфейсу найбільш зрозумілий та простий у засвоєнні. Ось чому зараз WIMP - інтерфейс став фактично стандартом. Найкращим прикладом програм з графічним інтерфейсом вважається операційна система Microsoft Windows.

Після широкого поширення технологій розпізнавання мовлення і появи недорогих звукових карт у середині 90-х років, з'явилася так звана "мовна технологія" SILK - інтерфейсу. Вона працює за допомогою команд, які подаються голосом шляхом проголошення спеціальних зарезервованих слів.

Основними такими командами є:

- "Привіт,..." - активація голосового інтерфейсу.
- "Бувай, ...." - закриття інтерфейсу голосового керування.
- "Відкрити" - запуск тієї чи іншої програми. Назва програми називається в наступною.
- "..., запиши" - перехід в режим набору голосом текстової інформації.
- І деякі інші.

При цьому слова проговорюються чітко, в одному темпі. Між словами обов'язково потрібно робити паузи. Через недостатній алгоритм розпізнавання мовлення такі системи потребують попереднього індивідуального налаштування на

конкретного користувача. "Мовна" технологія вважається найпростішою реалізацією SILK - інтерфейсу.

## 1.2 Принципи проєктування інтерфейсів

Процес сучасного проєктування програмного забезпечення окрім вирішення різноманітних проблем, включаючає зменшення витрат та скорочення часу на проєктування, покращення якості пропонованих рішень, простоту й використання програмного забезпечення, вивчення і впровадження нових технологій та інструментів. А також можливість робити це краще за конкурентів.

Задоволеність користувачів і простота використання програмного продукту значною мірою визначаються інтерфейсом користувача. Загалом, задоволеність споживачів - це функція декількох факторів (вони є причинами, характером процесу або рушійною силою, що визначає функціонування особистості);

- функціоналу інтерфейсу користувача
- часу реакції
- надійності
- пристосованості до інтеграції,
- інформаційної підтримки,
- готовності до супроводження
- інших факторів.

Виділені фактори відіграють найбільш важливу роль. Функція ІК повинна повністю відображати функцію програми. Час її реакції повинен бути мінімальним. Тому користувач не повинен чекати довше, ніж це потрібно для виконання зазначеного завдання. Надійність - це властивість, яка економить час у межах обмеженого діапазону всіх параметрів, що характеризують здатність виконувати необхідні функції за заданими параметрами. Пристосованість до інтеграції відіграє важливу роль у задоволенні користувачів, оскільки вони знайомляться з програмою та інтерфейсом під час встановлення програми. Інформаційна підтримка користувачів - це навчально-довідковий компонент програми, який визначає, наскільки швидко та легко користувачі знайомляться з новими програмними продуктами. Технічне

обслуговування ІК - це процес вдосконалення, оптимізації та усунення недоліків інтерфейсу після розробки програмного забезпечення. Важливо використовувати інтерфейс у ЗМІ для збільшення якості використання за участю користувача. Інші фактори включають узгодженість, інтеграцію та витрати на випромінювання, що впливає на задоволеність користувача інтерфейсом та програмним забезпеченням в цілому.

На кожному етапі життєвого циклу ІК-програмного забезпечення слід враховувати всі причини задоволеності користувачів та його важливість.

Проблеми з ІК-дизайном включають той факт, що користувачі на етапі проектування не можуть чітко висловити свої вимоги та вимоги до програмних продуктів та інтерфейсів, а на етапі програмування вони добре обізнані про бажані та невідповідні якості певних властивостей.

У багатьох випадках характеристики інтерфейсу з точки зору продуктивності, інтеграції та послідовності на етапі проектування програмного забезпечення не є чітко визначеними, але визначаються різними очікуваннями, що призводить до нерозуміння сподівань замовників та розробників програмного забезпечення. Отже, ці потреби повинні бути чітко визначеними та вимірюваними (включаючи тестові функції). Дизайнери ІР можуть не бачити і не розуміти відчутних і зрозумілих потреб споживачів.

У минулому інтерфейс користувача розвинувся лише завдяки винаходу технологій та систем, заснованих на розробленому програмному забезпеченні. Цей підхід називається розробкою системи або розробкою технології. Вподобання користувачів ніколи не враховувались, а функції та рекламні інтерфейси програмного забезпечення були доступні розробникам.

Однією з основних причин створення невдалого програмного забезпечення є недостатня участь користувачів у проекті. Не менш важливим є результат належної участі споживачів у проекті, особливо відсутність знань реальних користувачів продукту та середовища застосування розробленого програмного забезпечення. Детальна інформація про користувача та його середовище допомагає налаштувати систему, для якої слід створити ІК-структуру, та забезпечити її функціональність. На

додаток до вимог інтерфейсу та його функцій, інформація про користувача та його середовище допомагає команді розробників виявити характеристики продукту, які потрібні споживачам. Це важливо при виборі відповідного методу розробки ІК та при розробці звичайної моделі.

З початку 1980-х років розробка програмного забезпечення змінила увагу на проблеми користувачів, і вони почали брати участь у розробці. Однак це була їхня робота - спостерігати. Вони з'ясували, що потрібно для програми і які завдання вони вирішать за її допомогою. На даний час у багатьох розробників існує стратегія "розвитку друзів та знайомих" та "розвитку споживачів". Чудова річ цього методу полягає в тому, що користувачі вважаються активними учасниками процесу розробки. Участь користувача покращує користувацький інтерфейс та доступ до програмного забезпечення та гарантує, що отримане програмне забезпечення відповідає вимогам та критеріям. Розроблення, орієнтоване на користувачів, що навчаються, спрямоване на те, щоб у процесі вирішення своїх задач людина навчалась новим навичкам роботи з ПЗ, тобто на її інтелектуальний розвиток, тренування її уяви і одержання знань в різних галузях.

Збільшення споживачів базується на наступних рекомендаціях.

- 1) Розуміння потреб користувачів є основою всього проекту.
- 2) Переконайтеся, що ви робите все, до чого користувачі можуть мати доступ.
- 3) Творчі проекти завжди є результатом зусиль групи експертів з різних галузей.
- 4) IP-рішення повинні базуватися на відгуках користувачів.
- 5) Відгуки користувачів повинні збиратися з певною точністю, швидкістю та частотою.
- 6) Компетентні та справжні користувачі отримують відгуки.
- 7) Орієнтований на споживача розвиток слід стандартизувати та впроваджувати.
- 8) Розвиток, орієнтований на споживача, повинен постійно оновлюватися.

Основні правила для дизайну, орієнтованого на користувача:

- 1) Поінформованість та відповідальність користувачів, залучення користувачів до різних фаз життєвого циклу програмного забезпечення



2) Визначення вимірюваних цілей, критеріїв ефективності з точки зору споживачів та користувачів

3) Проект повинен передбачати нові функції користувача, пов'язані з товаром, такі як упаковка, маркетинг, навчання, опублікована інформація, роздільна здатність, установка, зображення у вигляді графіка на екрані (фотографії), що відображаються в додатку. Цифрове зображення, захоплене командною програмою користувач-користувач), розклад, підтримка, інша оперативна підтримка, оновлення та встановлення

4) Оцінка програмного забезпечення та фактична участь користувачів для визначення того, чи були досягнуті цілі, а також у чому проблема

5) Спираючись на оцінку переглянути, якщо мета не була досягнута або якщо є проблема – виправити її

Якщо регулярно повертатися до попередніх кроків, ви не отримаєте успішного інтерфейсу, тому вам доведеться повторювати кожен процес ІК. Критерієм завершення ітеративного розвитку є той факт, що всі застосовані вимоги виконуються, а сам продукт відповідає наміченій цілі. Ітераційний процес може зайняти багато часу через повторювані стадії розвитку. Однак ранні етапи фази допомагають створити можливості для розробки та виробництва дослідних зразків, заощаджуючи час на впровадження та тестування в наступних ітераціях.

Отже, найефективнішим підходом до розробки ІК, орієнтованого на користувача, є розвиток, ітеративний підхід „ззовні всередині”. Орієнтовані на користувача кроки ІК-проекування наведені в таблиці 1.1.

Таблиця 1.1.3 - Етапи проектування ІК, орієнтованого на користувача

Етап	Опис етапу
(1)	(2)
План	План створення продукту орієнтований на побудову ІК та забезпечення практичності; план враховує календарні терміни для кожного з етапів процесу

	розроблення ІК, визначає основні ризики, об'єднує всі можливі методи, встановлює цілі та критерії оцінювання ІК
Вимоги	На етапі встановлення вимог вирішуються задачі: опис профіля користувачів, постановка задач користувачів, оцінка поточного рівня практичності, аналіз можливостей ІК, аналіз тенденцій розвитку ІК
Концептуальне проектування	Концептуальний проект є сукупністю високорівневих описів, абстракцій та оглядової інформації, яка дає розробникам та кінцевим користувачам загальне уявлення про програмний продукт, його структуру та ІК
Проектування	Проект ІК є сукупністю характеристик програми, які сприймаються користувачем (вхідні сигнали, взаємодія користувача, відгук системи на вхідні сигнали та взаємодію користувача)
Прототипування	Створення прототипів та моделювання – ефективні засоби ранньої оцінки проекту; прототип – це матеріалізація побудованого проекту з використанням його передбачуваної платформи реалізації, включаючи обладнання, ОС, мови і засоби реалізації
Специфікація	Матеріалізація проекту програмного продукту в документальній формі, яка описує дії користувачів, а також вигляд та поведінку ПЗ в специфічних ситуаціях
Конструювання	Написання коду та автономне тестування
Оцінювання	Всі методи оцінювання пов'язані із залученням потенційних користувачів програмного продукту

(1)	(2)
Ітеративний підхід	Загальні критерії досягнення цілей створення ІК повинні бути чітко визначені, зрозумілі й прийняті замовниками та розробниками; досягнення поставлених цілей може вимагати багатократних ітерацій
Впровадження та підтримка	На етапі впровадження здійснюється: оцінка ПЗ за участю користувачів, які не залучались до розробки, пілотне тестування, виконання задач, які не оцінювались або не були передбачені під час проектування та розроблення

Щоб створити стиль інтерфейсу та успішне програмне забезпечення потрібний хороший персонал, що володіє різноманітними спеціалізованими навичками у таких сферах: дизайн ІК; програмне забезпечення та технологія розробки інтерфейсу; тестування та оцінка якості програмного забезпечення та ІК; Стандартизація програмного забезпечення та інтерфейсу; інструменти для його впровадження та використання у додатках; дизайн та реалізація візуально-графічних структур; розробка та впровадження засобів навчання, посібників, систем підтримки та електронної оперативної підтримки психології та ергономіки, поведінки людини, сприйняття, навчання і пізнання; бізнес-план; керівник проекту. Отже, потрібна група розробників для успішного проектування ІК. Кожна команда розробників є експертом у певній галузі і має такі характеристики: здатність працювати в команді, здатність працювати з розробниками та бізнес-структурами; здатність розуміти користувачів та чітко формулювати невизначені вимоги щодо пошуку інформації; можливість визначення та узгодження з користувачами кількісної оцінки практичності, інтеграції та послідовності пошуку інформації; можливість використання засобів і методів розробки програмного забезпечення; методи використання інструментів для тестування програмного забезпечення інтерфейсу користувача та для розрахунку

надійності і якості програмного забезпечення інтерфейсу користувача; технологія швидкої й ефективної оцінки та відновлення конструкції і застосування.

Організаційні, групові та індивідуальні проблеми, що виникають при розробці ІК, набагато складніші, ніж технічні проблеми. 80% проблем стосується людей і лише 20% технічних.

Важливість різних професійних навичок та залучення до проекту відповідної кількості професіоналів, які мають кожен тип знань, вимагає ретельного планування, щоб проект був успішним. Правильне визначення ролі розробника в команді розробників усуває невизначеності. Один експерт може зіграти одну або кілька ролей.

Чому слід дотримуватися принципів створення користувацького інтерфейсу?

Програма повинна бути розроблена відповідно до вимог та побажань користувача - система має бути адаптована до замовника. Ось чому принципи дизайну так важливі.

Користувачі комп'ютерів можуть отримати успішний досвід, який додасть їм переконаність у своїх силах та підвищить високу впевненість у собі в роботі з комп'ютером. Їхні дії з комп'ютером можна охарактеризувати як "успіх приносить успіх". Будь-який позитивний досвід спілкування з програмою дозволяє клієнтові розширити свої знання програмного забезпечення та підвищити рівень своєї компетентності. Добре розроблений інтерфейс, такий як хороші вправи та підручники, забезпечує плідну взаємодію між користувачем та комп'ютером. Успішні інтерфейси можуть навіть допомогти людині позбутися звичних програм, які він використовує, і відкрити нові, щоб поглибити своє розуміння інтерфейсів та комп'ютерів.

Принципи проектування інтерфейсу - це концепції та ідеї високого рівня, які можуть бути використані при розробці програмного забезпечення. Вам потрібно визначити, який принцип є найбільш важливим та прийнятним для вашої системи.

При плануванні потрібно визначити найважливіший принцип, який буде вирішальним у пошуку компромісів. Дотримання всіх принципів проектування може призвести до того, що в деяких ситуаціях це не виправдовує і негативно впливає на кінцевий результат. Принципи - це керівництво до дії.

Три принципи розробки користувацького інтерфейсу сформульовані наступним чином:

- 1) управління інтерфейсом;
- 2) зменшення навантаження пам'яті користувача;
- 3) послідовність інтерфейсу користувача.

Де знайти принципи розробки користувацького інтерфейсу.

Хансен представив перший перелік принципів дизайну. Принципи такі:

- 1) знає клієнта;
- 2) зменшити пам'ять;
- 3) оптимізація бізнесу;
- 4) виправте помилки.

Багато основних виробників операційних систем випускають нові продукти, публікують відповідні інструкції та вказівки. Посібники виготовлені компанією Apple Computer, Inc. (1992), IBM Corporation (1992), Microsoft Corporation (1995) та UNIX OSF / Motif (1993). Ці публікації розкривають принципи підходу до проектування інтерфейсу.

Читачі можуть використовувати ці публікації як навчальний посібник. Термінологія інтерфейсу може відрізнитися залежно від каталогу, але привабливість матеріалу, рівень складності та принципи, описані в інтерфейсі користувача, однакові для всіх книг.

Важливість поваги принципів

"Невідповідність інтерфейсу може коштувати великій компанії мільйони доларів через втрату робочої сили та збільшення витрат на технічну підтримку".  
"Джессі Бріст".

Ці принципи стосуються всього програмного та апаратного забезпечення, усіх типів та стилів інтерфейсів. Вони розроблялися протягом тривалого часу: проводились дослідження в галузі програмного інтерфейсу, проводились заходи, проводились інтерв'ю з користувачами багатьох комп'ютерних платформ, включаючи Macintosh та персональний комп'ютер.

Ці принципи витримали випробування часом і впровадженням нових комп'ютерних технологій. Як об Нільсен зазначив: "Принципи залишаються основними, навіть якщо програма має футуристичний тривимірний дизайн, що включає DataGlove, що розпізнає рух та відеозображення в реальному часі." Вони будуть актуальними, оскільки вони висловлюють основну ідею діалогу з автомобілем за допомогою команд".

Визначення цих принципів залежить від обладнання, операційної системи, компонентів інтерфейсу користувача та його завдань. Найчастіше ділове рішення переважає використання принципів дизайнерами. Дизайн та дизайнерські моделі також різні і впливають на застосування принципів. На деяких важливих етапах розробки проекту може постати питання: "Що буде далі?". Відповідь повинна бути такою: "Що хоче користувач!".

Рішення щодо оптимальних принципів проектування інтерфейсу повинні приймати всі члени команди проекту. Ці рішення допоможуть покращити закупівлю та розробку програмних продуктів.

Правила оформлення інтерфейсу користувача:

"Будьте простими, але не більш примітивними".

Альберт Ейнштейн

Правило 1: Контроль за замовником

Досвідчені дизайнери дозволяють користувачам вирішувати певні завдання на власний розсуд. Архітектори повинні завершити тротуари між ними після завершення будівництва складного комплексу будівель. Вони ще не знають, де люди потраплять на сайт. Тому доріжки ніколи не руйнуються одночасно з будівництвом будинків. На майданчику між будинками стоять таблички із написом: "Будь ласка, йдіть по траві". Через деякий час будівельники повертаються і лише зараз, на "бажання" населення, вимощені доріжки покривають асфальтом.

Принципи, що дозволяють користувачам керувати системою:

1) розумно використовувати режими;

2) дозволити користувачеві вибрати: працювати з мишкою чи клавіатурою або комбінацією обох;

3) дозволити клієнту зосередитися;

4) демонструвати повідомлення, які допоможуть йому у його діяльності;

5) створити умови для негайних заходів та заходів зворотного зв'язку;

6) забезпечити відповідні маршрути та виходи;

7) адаптація системи до користувачів різного рівня підготовки;

8) зробити інтерфейс користувача більш зрозумілим;

9) дозволити користувачеві налаштувати інтерфейс на свій смак;

10) дозволяє користувачеві безпосередньо маніпулювати об'єктами інтерфейсу;

Такий шлях не завжди поганий. Людина повинна мати можливість вибрати способи, яких вона потребує. Інтерфейс повинен бути настільки природним, щоб користувач міг з ним комфортно працювати. Користувач не замислюється про перехід на режим вставлення або копіювання під час роботи в текстовому редакторі - це абсолютно раціонально і природно.

Можливість роботи з клавіатурою замість миші. Це означає, що користувачеві буде легше брати участь, просто не зможе ним скористатися або не доведеться. Конфіденційність створюється під час використання миші для прискорення роботи. Однак під час роботи на клавіатурі ви не можете їх зв'язати - для таких випадків існують "випадаючі" меню.

Дозволити користувачеві переключити увагу.

Це не змушує користувачів завершувати початкову послідовність. Виберіть їх - видаліть або збережіть дані і поверніться туди, де вони їх зламали. Дозволити користувачам контролювати робочий процес у програмі.

Показувати пояснювальні повідомлення та тексти

Використовуйте зручні для користувача терміни в усьому інтерфейсі. Їм не потрібно знати про біти та байти!

Вам потрібно вибрати правильний тон для повідомлень та запрошень. Не менш важливим є страхування від проблем та помилок. Невдала термінологія та неправильний тон призведуть до того, що користувачі звинувачують себе в помилках.

Забезпечуйте негайний зворотний зв'язок та відгуки

Кожен програмний продукт повинен включати функції UNDO та REDO. Користувачеві слід повідомити, що цю дію не можна відмінити, і, якщо можливо, дозволяється альтернативна дія. Інформуйте людину про те, що зараз відбувається.

Переконайтесь, що дороги та виїзди чисті

Користувачам подобається працювати з будь-яким програмним інтерфейсом. Також інтерфейси, що використовуються в промисловості, не повинні лякати користувача, не боятися натискань кнопок або переходу на інший екран. Вториння в Інтернет показало, що навігація є основною інтерактивною технікою в Інтернеті. Якщо користувач розуміє, як потрапити на потрібну сторінку в WWW, є 80% шансів, що він зрозуміє інтерфейс. Люди дуже швидко вчаться працювати з браузером.

Адаптуйте до користувачів різного рівня кваліфікації

Не жертвуйте досвідченими користувачами на благо звичайних людей. Вам потрібно надати їм швидкий доступ до функцій програми. Якщо ви використовуєте один макрос, не втомлюйте їх кількома кроками.

Зробіть інтерфейс користувача "прозорим"

Інтерфейс користувача - це "міфічна" частина програмного продукту. Завдяки хорошему проекту користувачі взагалі не відчують його "присутності". Якщо його встановити невдало, користувачам навіть доведеться багато попрацювати, щоб ефективно використовувати програмний продукт. Завдання інтерфейсу полягає в тому, щоб допомогти людям залишатися впевненими, ніби вони перебувають "усередині" комп'ютера, вільно маніпулюючи предметами та працюючи з ними. Це називається "прозорим" інтерфейсом!

"Прозорість" інтерфейсу забезпечується тим, що людина зможе використовувати об'єкти, крім системних команд.

Дозволяйте користувачеві налаштовувати інтерфейс

Секрет "прозорого" інтерфейсу безпосередньо пов'язаний з ментальною моделлю. Користувач повинен бути безпосередньо зосереджений на завданнях, з якими він стикається, і не розуміти функцій програми.



Представляючи функції OS / 2 та діалогові вікна зміни функцій у Windows 95, можна змінювати кілька системних налаштувань та об'єктів. Розробники Windows 95 навіть створили додаткову програму - Tweak UI. Програмні продукти повинні використовувати функції операційної системи відповідно до інших програм. Однак усі інші атрибути API, включаючи меню та кнопки, повинні мати функцію налаштування.

Дозволяйте користувачеві безпосередньо керувати об'єктами інтерфейсу

Користувач починає сумніватися у власній силі, якщо безпосереднє маніпулювання предметами не відповідає його ментальній моделі та системі уявлень про взаємодію з реальним світом. Просте правило: збільшуйте метафору, але не переривайте її. Іноді система прямого маніпулювання виходить з ладу, якщо користувач не знає, що взяти і де його розмістити. Об'єкти повинні "кричати" на користувачів: "Хапай мене, звільняй, поведися зі мною як з об'єктом, який я представляю!". Інакше людина не зрозуміє, як працювати з цим об'єктом. Єдина проблема безпосереднього маніпулювання полягає в тому, що візуально не ясно, чи можна об'єкт переміщати та відображати. Користувачі повинні почуватись комфортно, виконуючи цю операцію, та усвідомлювати запланований результат. Крім того, інтерфейс повинен бути простим для вивчення.

Дозвольте клієнту подумати, що він контролює ситуацію

Добре розроблений інтерфейс повинен бути зручним та цікавим для запуску комп'ютера. Люди не люблять сидіти за комп'ютером, коли комп'ютер зайнятий, і нічого не роблять. Якщо ви не можете надати клієнту контроль, ви повинні створити його ілюзію!

Правило 2: Зменшити споживання вантажу

На основі знань про те, як людина зберігає та запам'ятовує інформацію, енергія комп'ютерного інтерфейсу повинна захищати від перевантаження пам'яті.

Принципи зменшення навантаження пам'яті користувача:

- 1) не завантажувати короткочасну пам'ять;
- 2) покладатися на підтвердження, а не на повторення;
- 3) представити візуальну заставку;

4) забезпечити налаштування за замовчуванням, команди Скасувати та Скасувати;

5) забезпечити "швидкі" дороги;

6) активізувати синтаксис дій з об'єктами;

7) використовувати реальні метафори;

8) використовувати розкриття та тлумачення понять та дій;

9) підвищити чіткість зору.

Не перевантажуйте короткочасну пам'ять

Це не змушує користувачів пам'ятати і повторювати те, що комп'ютер може (і повинен) робити. Наприклад, коли вам потрібно заповнити анкету, вам треба ввести інформацію - ім'я, адресу, номер телефону, записаний системою для подальшого використання, коли користувач знову входить в систему або відкриває запис. Система повинна "пам'ятати" введені дані та постійно забезпечувати безперервний доступ до них.

Покладайтеся на підтвердження, а не на повторення

Вкажіть списки та меню, що містять елементи або документи, які користувачі можуть вибрати без ручного входу. Наприклад, чому люди запам'ятовують двобуквені скорочення для кожної країни США, коли заповнюють анкету чи форму? Їх не слід примушувати запам'ятовувати коди для подальшого використання. Надайте списки найпопулярніших предметів та документів, які легко вибрати без заповнення замовлення тощо.

Коли користувачі перебувають у режимі або використовують мишу, це впливає на екран. Екран повинен інформувати користувача про спосіб його присутності. Ви можете змінити вказівник, щоб вказати поточний режим або дію, а також увімкнути або вимкнути індикатор. Тест на візуальну обізнаність щодо продукту: вимкніть комп'ютер і поверніться до роботи пізніше під час виконання завдання. Зверніть увагу на візуальні інструкції інтерфейсу, які повинні інформувати вас про те, з чим ви працювали, де ви були і що робили.

Надайте функції для скасування останньої дії, її повторення та стандартних налаштувань.

Використовуйте свій комп'ютер для зберігання та отримання інформації про вибір користувачів та функції системи. Забезпечити багатоступеневі системи скасування та повторення команд, які забезпечують безпечну та тиху роботу програми.

Він пропонує ярлики інтерфейсу

Після того, як користувачі ретельно дослідять програмний продукт, вони відчують потребу в акселераторі. Не ігноруйте цю потребу, але дотримуйтесь стандартів при плануванні.

Активуйте синтаксис дій з об'єктами

Предметний синтаксис дозволяє людині зрозуміти зв'язок між об'єктами та діями програмного продукту. Користувачі можуть досліджувати та "обертати" інтерфейс, вибираючи елементи та переглядаючи доступні дії.

Використовуйте справжні метафори

Будьте обережні при виборі та використанні метафор інтерфейсу. Вибравши метафору, виправте її і послідовно дотримуйтесь її. Якщо метафора не відповідає своєму призначенню в усьому інтерфейсі, виберіть новий. Продовжуйте притчу, не перериваючи її.

Поясніть поняття та дії

Ніколи не забувайте про легкий доступ до часто використовуваних функцій та дій. Приховуйте непопулярні функції та властивості і дозволяйте клієнту використовувати їх за потреби. Вам не потрібно намагатися відображати всю інформацію в головному вікні. Використовуйте додаткові вікна.

Підвищення чіткості зору

Дизайнери комп'ютерної графіки та книг оволоділи мистецтвом подання інформації. Програмісти користувацького інтерфейсу також повинні мати цю можливість.

Правило 3: Зробіть інтерфейс сумісним

Сумісність є ключовим аспектом інтерфейсу. Однак поспішати з цим ніяк не слід. Однією з головних переваг послідовності є те, що користувачі можуть передавати свої знання та вміння зі старої програми, якою раніше користувалися.

Принципи створення сумісності інтерфейсу:

- 1) створити послідовний інтерфейс;
- 2) загальна сумісність усіх програм;
- 3) зберігати результати взаємодії;
- 4) естетична привабливість та цілісність;
- 5) сприяння навчанню.

Створіть послідовний інтерфейс

Користувачі повинні мати орієнтири під час навігації по інтерфейсу. Це заголовки вікон, навігаційні карти та сталі конструкції. Ще одним наочним посібником є миттєвий динамічний огляд місця. Користувач також повинен мати можливість виконувати завдання без зміни умов роботи або перемикання між стилями введення.

Загальна сумісність усіх програм

Вивчення однієї програми не повинно суттєво відрізнятися від вивчення такої програми. Коли подібні об'єкти не працюють однаково в різних ситуаціях, у користувачів є мінус. Це уповільнює вивчення програми і призводить до того, що користувач втрачає довіру.

Покращений інтерфейс та узгодженість

Дизайнери програмного забезпечення повинні знати про використання набутих знань та бути обережними при впровадженні нових. Якщо ми хочемо вдосконалити інтерфейс, користувачеві потрібно лише вивчити кілька нових методів взаємодії. Не змушуйте його повторювати і забувати довготривалі навички. Набагато важче “придушити” існуючі навички, ніж набути нових.

Зберігайте результати взаємодії

Якщо результати можуть відрізнятися від очікувань користувача, повідомте їх про це, перш ніж вживати заходів. Дайте йому варіанти кампанії, скасуйте кампанію або займіться чимось іншим.

Естетична привабливість та цілісність

Інтерфейс, який виглядає добре, не повинен приховувати відсутність функціональності програмного продукту. Користувачі не повинні бачити все одразу, вони повинні отримати гарний інтерфейс, який допоможе їм у роботі.

#### Сприяння навчанню

Інтерфейс сьогоднішнього та завтрашнього дня є більш інтуїтивним, передбачуваним, доброзичливим, привабливим. Вторинення продуктів CD / DVD-ROM та Інтернет-браузерів, домашніх сторінок та програм відкрило світ для користувачів комп'ютерів. Пора майже всім програмам стати зручними та привабливими.

#### Настанови

"Кількість рекомендацій зростає пропорційно кількості людей, залучених до процесу створення та розробки, а також використання комп'ютерних систем".

#### Джон Гулд

#### Для чого призначені інструкції?

Розглянуті вище принципи призначені для представників усього дизайну процесу. Інструкція стосується елементів подання та взаємодії інформації. Інструкції - це правила та визначення, які створюються під час створення елементів інтерфейсу, їх поведінки та зовнішнього вигляду.

Дотримання інструкцій з проектування, незалежно від уподобань користувача, зазвичай призводить до несправності інтерфейсу. Зручний та послідовний інтерфейс не буде створений, якщо ви сліпо дотримуєтесь інструкцій, не розуміючи механізму взаємодії. Джон Гулд дуже добре це описав: "Занадто багато інструкцій приділяє занадто багато уваги "розташуванню кнопок", а занадто мало - розумінню та навчанню". Отже, подання посібників та інструкцій - не єдиний критерій успіху.

Вказівки щодо створення інтерфейсу, викладені в інструкціях, не повинні зменшувати чи обмежувати творчу активність, а мають дозволяти користувачеві використовувати свої знання про реальний інтерфейс (наприклад, якщо користувач бачить набір кнопок з подібними перемикачами, він може використовувати свої знання про реальні кнопки комп'ютера).

Посібник зі створення інтерфейсів доступний для сучасних систем виведення та введення інформації. Вони включають такі технології, як використання пера, рукописного вводу та введення голосу. Однією з труднощів у розробці інструкцій, що стосуються нової технології, є розшифрування способів взаємодії з користувацькою системою, оскільки ступінь цієї взаємодії ще не визначений. Інструкції повинні базуватися на відповідях користувачів на нові функції та створюватися протягом певного періоду, при цьому користувачі повинні опановувати інтерфейс та формувати про нього думку.

Стандарти.

Рекомендації включають функції стандартів презентацій, поведінки та взаємодії з інструментами управління інтерфейсом.

Посібник з елементів інтерфейсу та елементів керування розповідає, коли їх виправляти, як їх "подавати" та як працювати з ними (наприклад, з клавіатурою чи мишкою). Весь набір посібників розкриває сутність кожного об'єкта та елемента інтерфейсу із способом та способом представлення на екрані, їх поведінкою, механізмом взаємодії користувача з ними.

Розробити існуючі рекомендації щодо проектування інтерфейсу.

Багато програмних продуктів призначені для роботи на різних платформах. Оскільки ці платформи мають різні операційні системи, інструменти та стилі інтерфейсу, дуже важко створити інтерфейс, який підходить для всіх платформ або працює на кожній платформі. Збірник підручників з промислового дизайну, був розроблений компанією Belcor. Він містить описи та інструкції для великих компаній та операційних систем, таких як IBM CUA, OSF, Microsoft Windows тощо.

Мета керівних принципів проектування зрозуміла: надати інформацію користувачам з будь-якої точки системи та у будь-якій формі, створити інтерфейс, який допоможе людям працювати та насолоджуватися. Добре розроблений інтерфейс дозволяє користувачам зосередитися на завданнях, а не на програмному забезпеченні та технічних характеристиках.

Використовуйте рекомендації.

Цілі та настанови щодо розробки інтерфейсу повинні бути реалістичними та зручними для користувача. Специфіка компанії накладає обмеження на це середовище. Слід також перевірити вказівки щодо дизайну інтерфейсу. Щоб продукт відповідав інструкціям, потрібна підтримка розробника. Дизайнери та розробники відповідають за розробку сумісного інтерфейсу.

Керівні принципи розвитку макро- та мікроінтерфейсів

При розробці рекомендацій щодо користувацького інтерфейсу важливо забезпечити аналіз розробки та простоти користувацького інтерфейсу з двох точок зору - мікро та макро.

Інструкції на мікрорівні обговорюють користувачів окремих елементів інтерфейсу (елементи керування - кнопки, поля виділення, текстові поля, смуги прокрутки тощо), а також способи взаємодії з ними.

Розробка інтерфейсу на макрорівні є шаблоном для інтерфейсу користувача - продукт повністю зібраний, і його концепція стає зрозумілою для користувачів під час взаємодії з ним.

### **1.3 Специфіка проєктування інтерфейсу веб-ресурсу державної установи**

Уряд запровадив єдині вимоги до оформлення офіційних веб-сайтів для виконавчої влади та доступу до інформації для користувачів із порушеннями слуху, зору, м'язів, мови та психічного здоров'я.

Відповідна Постанова Кабінету Міністрів України "Про внесення змін до деяких постанов Кабінету Міністрів України щодо функціонування офіційних веб-сайтів органів виконавчої влади" № 493, прийнята на засіданні Уряду 12 червня 2012 року.

Нею встановлюються уніфіковані вимоги до дизайну офіційних веб-сайтів органів виконавчої влади та доступність розміщеної інформації. Виконання цих вимог дозволить зробити веб-контент доступним та зручним для всіх користувачів.

Ця Постанова дозволяє вирішити наступні питання:

доступність веб-контенту для користувачів з порушенням зору, слуху, опорно-рухової системи, мови та інші;

зручність використання веб-сайтів;

підвищення якості та доступності електронних послуг для всіх користувачів.

Вона також усуває дискримінаційні елементи щодо осіб, які мають порушення зору, слуху, опорно-рухової системи, мови, шляхом забезпечення користувачів доступним та зручним сприйняттям інформації на офіційних веб-сайтах органів виконавчої влади.

Створюються умови, які в подальшому допоможуть реалізувати принцип єдиного вікна (“one-stop-shop”) для доступу фізичних та юридичних осіб до інформації про діяльність органів державної влади.

Бажано публікувати документи у текстовому форматі безпосередньо на веб-сайті та, при необхідності, завантажувати файли із зазначенням їх формату та розміру.

За відсутності інформації у текстовій формі, опублікованій безпосередньо на веб-сайті, доданий файл повинен містити опис короткого тексту та інформацію про його формат та розмір.

Статут та проект Положення публікуються на офіційному веб-сайті в текстовому форматі, який при необхідності може бути доданий до інших файлів.

Вимоги також існують безпосередньо до тексту документів.

Текст повинен бути простим і стислим. Його слід поділити на розділи з заголовками, абзацами, уникаючи довгих речень, складних слів і фраз, спеціальних виразів, а також рідкісних скорочень. Рекомендована довжина речення повинна складати від семи до дванадцяти слів.

Коефіцієнт контрастності основного тексту повинен бути не менше 4,5: 1 щодо фону.

Не допускається використання маленьких шпильок та декоративних шрифтів.

Крім того, офіційний веб-сайт повинен містити інформацію про права інтелектуальної власності, яка:

- знаходиться внизу кожної сторінки вмісту;



- заявляє, що матеріали, опубліковані на офіційному веб-сайті, підпадають під охорону законом прав інтелектуальної власності;

- описує умови, за яких певний матеріал може бути використаний іншими.

Зміст офіційного веб-сайту повинен вільно поширюватися.

Користувачі повинні мати можливість ділитися матеріалами з сайту в соціальних мережах. На кожній сторінці вмісту повинен бути блок із кнопками розподілу.

Зміст сторінки оптимізовано за допомогою пошукових систем для контекстного пошуку.

Структура сайту - це внутрішня структура, його "хребет", розташування сайту, розділи, підрозділи, додатковий матеріал. Основне завдання дизайнера - створити послідовний порядок хаотичного накопичення інформації.

Веб-сайти з поганою логічною структурою не тільки ускладнюють роботу відвідувачів, але й позбавляють бажання бути на них. Коли відвідувачі не можуть знайти потрібну інформацію, вони відчують, що її взагалі немає на сайті. Розчаровані відвідувачі не запам'ятовують цей ресурс, тим більше не хочуть його відвідувати.

Успішний дизайн повинен орієнтуватися насамперед на відвідувача, який може легко знайти та використати всю інформацію, розміщену на сайті. У той же час логічний макет сайту та логічний взаємозв'язок між окремими сторінками повинні бути цілком зрозумілими.

Правильна навігація означає, що відвідувачі точно знають, де вони перебувають, де знаходяться елементи сайту та як ними користуватися. Правильна структура інформації дозволяє відвідувачам продовжувати відвідувати сайт без страху. Вона надає можливість легко повернутися до раніше переглянутих сторінок. Турбота про структуру веб-сайту - завдання творця, а не відвідувачів.

Логічне проектування передбачає організацію інформації на сайті, побудову його структури та навігацію по сегментах.

При розробці вмісту веб-сайту та засобів навігації слід враховувати чотири основні принципи, засновані на сприйнятті людиною інформації:

- Використання символів

Вам потрібно використовувати слова та фрази, які є послідовними та зрозумілими для більшості відвідувачів сайту. Використання піктограм не дуже вдале, оскільки відвідувачі можуть неправильно їх інтерпретувати. Елементи навігації та посилання повинні мати сильне значення.

- Ефективність

Розділи веб-сайтів повинні бути заповнені інформацією, що стосується цього розділу або частини сторінки, бажано переміщувати або видаляти недоречні елементи.

- Єдність

Використовуйте ту саму навігацію сайтом з однаковим дизайном. Однакові елементи з боків повинні бути однакового розміру та в одному і тому ж місці. Якщо сторінки з якихось причин різні, відвідувач повинен чітко розуміти, чому так.

- Розділити на частини

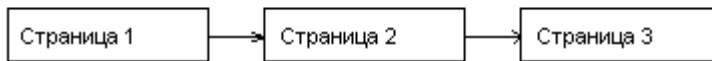
Людина не може одночасно зберігати великий обсяг інформації. Як правило, люди можуть сприймати інформацію, яка містить чотири-шість різних елементів. Як результат, відвідувачі сайту краще націлені і швидше знаходять необхідні їм матеріали, якщо візуально розділити їх на групи.

Організуючи дані у формі організованої структури, необхідно повідомити відвідувача, яку інформацію він знайде на сайті і де вона знаходиться. Структура веб-сайту схожа на зміст книги: якщо він добре складений, буде зрозуміло, чого чекати від книги перед читанням. Інформація повинна бути організована таким чином, щоб гість знав назву посилання, яке потім чекає на сторінці.

Організація інформації на веб-сайті може бути:

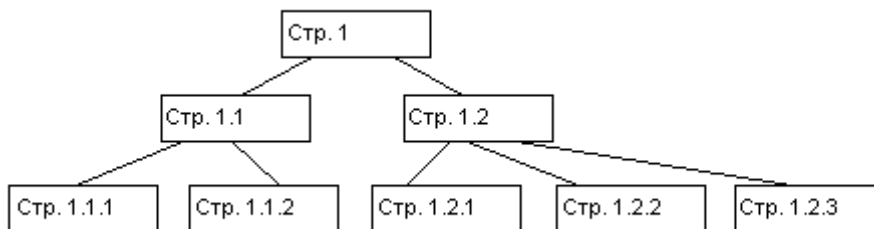
- Лінійна.
- Ієрархічна.
- Мережа.
- Комбінована.

## Лінійна структура



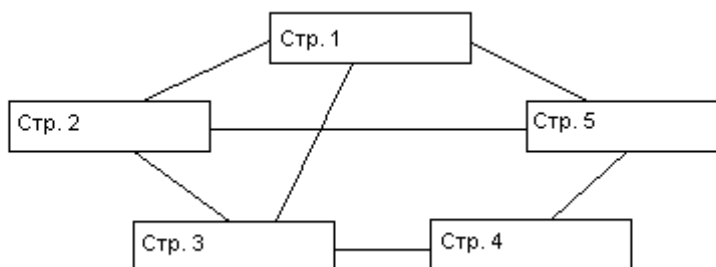
Сторінки в структурі цього сайту можна переглядати послідовно одну за одною. Ви можете переключитися з поточної сторінки на наступну або попередню сторінку. У чистому вигляді лінійна структура не використовується для організації веб-сайтів, оскільки її використання незручне. Зазвичай його використовують для поділу досить довгої сторінки на кілька коротких, логічно відокремлених частин. Наприклад, ви можете розібрати розділи книг під час відображення тексту книги.

## Ієрархічна структура



Дозволяє використовувати домашню сторінку, яка містить посилання на розділи сайту з меню на наступних сторінках. Розділи можуть містити посилання на підрозділи або іншу детальну інформацію.

## Мережна структура



Це найпоширеніша структура в Інтернеті. За допомогою такої організації ви можете переходити з будь-якої сторінки на будь-яку іншу сторінку, яка логічно пов'язана з нею. Яскравими прикладами такої структури є торгові сайти або великі інформаційні портали.

Згідно з планами розробника, немає необхідності пов'язувати всі сторінки з усіма. Посилання повинні бути максимально природними та відображати логіку сайту.

### **Комбінована структура**

Це поєднання кількох попередніх структур. Наприклад, сайт може мати ієрархічну структуру, яка в деяких місцях може включати покрокові дії (лінійна структура).

Спроба створити структуру сайту без урахування думки відвідувачів є великою помилкою, яка може лише зашкодити популярності сайту. Яким би приємним не був сайт, вирішальним буде питання, чи легко відвідувач знайде потрібну інформацію. Дизайн зроблений для зручності відвідувачів, а не власників сайтів.

Звичайно, логічну структуру веб-сайту можна створити лише інтуїтивно. Однак однією з найбільших помилок є використання схем компаній, які є прийнятними для цих компаній, таких як розподіл товарів за брендами або ідеальне збіг структури сайту зі структурою організації. В результаті ви можете створити веб-сайт, структура якого буде очевидна для розробників і абсолютно незрозуміла для відвідувачів.

Зручний веб-сайт повинен базуватися на законах ергономіки. Ергономіка - це наука, яка вивчає робочі процеси, щоб створити оптимальні умови праці, які допоможуть підвищити її продуктивність та зберегти силу, здоров'я та ефективність.

Найважливішим аналізатором сприйняття інформації "людина-комп'ютер" для системи є візуальний. Ергономічний веб-сайт базується на знанні будови та функції людського ока, який збирає інформацію при перегляді на екрані під обмеженим кутом.

Розглянемо ключові елементи ергономічного візуального інтерфейсу:

- Цей інтерфейс відображається на екрані монітора і включає найактивніші області. Одразу видно та виділяються найважливіші елементи сторінки.
- Колірна схема елементів інтерфейсу повинна базуватися на психофізіології кольору. Червоний, зокрема, є ознакою небезпеки, зелений - сигналом нормального стану, тощо.

- Розмір шрифту, шрифт та макет текстових блоків на сторінці слід вибирати відповідно до їх важливості для цільової аудиторії.

- Навігаційна система повинна бути зручною та інтуїтивно зрозумілою.

- Інтерфейс повинен адекватно моделювати реальні ситуації, що відбуваються в певній області програми. Він має бути незвичним, щоб відвідувач фокусувався на змісті веб-сайту, а не на його презентації.

Звичайно, для створення привабливих та інформативних веб-сайтів недостатньо вивчити ази дизайну. Цього можна досягти лише завдяки тривалій практиці. Але з самого початку бажано пам'ятати основи та дотримуватися основних правил:

- Колірна гамма сторінки повинна складатися з одного або двох кольорів;
- меню не повинно містити більше семи пунктів;
- Логотип має бути вгорі праворуч або ліворуч, простий і короткої форми;
- дизайн повинен бути простим, а текст коротким;
- горизонтальний зсув вікна не допускається;
- чорний колір має негативне (похмуре і гнітюче) навантаження;
- важко читати світлий текст на темному тлі;
- шрифти без засічок читаються легше, ніж шрифти із засічками;
- рамки на сайті небажані.

Кожне з цих правил можна оскаржити та довести необхідність, а також успіх використання “неправильного варіанту” на певному сайті. Тому їх не слід сприймати як безумовне табу. Однак потрібно пам'ятати, що для більшості сайтів все-таки найкраще дотримуватися наведених вище рекомендацій.

Слід зазначити, що Інтернет - це досить поширений сайт, незручний для відвідувачів. Ці незручності в основному проявляються кількома способами:

- Повільне завантаження. Чекати завантаження файлів і завершувати пошук - марна трата часу. Відвідувач сайту не буде чекати більше десяти секунд, поки потрібна сторінка завантажиться. Важливо, щоб ви отримали інформацію, яку

шукаєте, якомога швидше і простіше. Для цього забезпечте зручну структуру сайту та швидке завантаження кожної сторінки.

- Інформаційна невизначеність. В Інтернеті багато інформації, і відвідувачі часто стикаються з неправильною або недостовірною інформацією. Відвідувач відчуває більшу довіру до сайту, коли чітко вказані джерела інформації та дата їх оновлення. Якісний дизайн також допомагає зміцнити довіру.

- Ілюстрації слід вибирати успішно та виправдано. Зображення, які не стосуються текстового вмісту, просто привертають увагу. Також використовуйте менші файли для швидшого відображення.

- Жирний текст, який важко побачити, є дуже привабливим елементом веб-сайту. Більшість відвідувачів переглядають Інтернет, і лише якщо їм цікаво, вони прочитають їх більш докладно. Щоб полегшити процес, текст слід надавати шрифтом, який виділяє важливі моменти, точні ілюстрації тощо.

- Помилки навігації. На кожній сторінці має бути чітко зазначений веб-сайт, до якого вона належить, та гіперпосилання на домашню сторінку. Надавайте просту та зрозумілу адресу на головній сторінці сайту без використання спеціальних символів.

- Нестандартний дизайн гіперпосилань. Відвідані гіперпосилання не повинні позначатися іншим кольором. Гіперпосилання також мають бути виділеними.

- Нові вікна браузера не можна відкривати без попередньої згоди або попередження відвідувача. Винятком є робота з документом, який не є веб-сторінкою (наприклад, файл doc або pdf).

- Фіксована ширина сторінки або елементів на ній. Відвідувачі веб-сайту повинні мати можливість налаштувати шрифт відповідно до своїх потреб за допомогою налаштувань браузера. Тому найкраще вказувати розміри шрифтів не в абсолютних числах, а у відносних показниках.

- Неправильне використання форм. З веб-сайтів, що містять форми, потрібно видалити будь-які запитання, без яких можна обійтися, і не робити поля обов'язковими.

- Погані заголовки, особливо однакові для всіх сторінок сайту, занадто довгі або малоінформативні, щоб створити небажане враження на сайті.

- Ігнорувати загальні правила, нестандартний інтерфейс. Стійкість - один з головних принципів практичності розташування. Якщо однакові дії призводять до однакових результатів, відвідувач відчуває впевненість у своїх діях. Відвідувачі веб-сайту вже знають, на що чекати, де знайти меню та елементи керування, тому їх не змушують закривати вікно браузера через непорозуміння або неприємне подання інформації.

- Реклама та все подібне. Рухливі предмети привертають увагу людини і не дозволяють зосередитись на змісті сторінки. Відвідувачі навчилися не помічати різнокольорових банерів, рухомих анімацій чи тексту, і зазвичай вони закривали всі вікна, що були відчинені, без їх згоди. На жаль, вони можуть не помітити необхідної інформації, що міститься в такому елементі.

- Сучасні технології - це дуже приваблива можливість, яка дозволяє дизайнеру повністю реалізувати бачення власного сайту. Однак майте на увазі, що не всі відвідувачі готові чекати кілька хвилин, поки елемент завантажиться перед переглядом сторінки, тому пропустіть його. Відображення деяких інтерактивних елементів може бути вимкнено з міркувань безпеки, тому завжди потрібно надавати версію веб-сайту з мінімальними вимогами до комп'ютера користувача.

- Несумісність з браузерами. Різні браузери можуть відображати однакові веб-сторінки по-різному. Тому потрібно переглянути та перевірити ваш сайт в іншому браузері, перш ніж розміщувати його в Інтернеті. Слід враховувати особливі потреби відвідувачів, особливо у випадку зі слабким зором - створювати відповідні заголовки, використовувати альтернативні описи зображень тощо.

- Гіперпосилання на самій сторінці не мають посилатися самі на себе.

#### **1.4 Порівняльний аналіз інтерфейсів державних веб-ресурсів**

Одним із показників відкритості державної інформації, яка є предметом чіткого кількісного аналізу, є висвітлення діяльності державних органів в Інтернеті. На законодавчому та нормативному рівні всі процедури чітко визначені для забезпечення відкритості інформації про діяльність органів державної влади. Це

передбачає, що 20 видів обов'язкової інформації має публікуватися на веб-сайтах центральних органів виконавчої влади та 26 видів на веб-сайтах органів місцевого самоврядування.

Для моніторингу державних веб-сайтів можна встановити наступні критерії:

1) Частота оновлення інформації - кількість оновлень на день / тиждень / місяць або час останнього оновлення, якщо нова інформація не з'являлась протягом тривалого часу (один тиждень або більше).

2) Етап розвитку, на якому знаходиться ділянка

Враховуючи той факт, що технології електронного уряду впроваджуються у кілька етапів, цей критерій відображає ступінь актуальності веб-сайту Органу.

3) Рівень інформативності сайту

Цей критерій повинен містити перелік інформації, опублікованої на веб-сайтах виконавчої влади. Тут можна використовувати такі рейтинги для оцінки повноти та ефективності інформації:

- повний;
- цілком закінчений;
- досить неповний.

4) Наявність інтерактивних функцій

Опис засобів та можливостей для зворотного зв'язку можна знайти на відповідному веб-сайті. Цей критерій включає наявність форумів на веб-сайті, обговорення керівником відповідного органу актуальних питань з громадянами, швидке реагування на запитання громадян тощо.

5) Мови, якими інформація доступна на веб-сайті Органу

Інформація на веб-сайтах центральних органів виконавчої влади, обласних державних адміністрацій, Ради міністрів АР Крим, та міських державних адміністрацій м. Києва та Севастополя повинна подаватися українською та англійською мовами, а за необхідності - іншими мовами.

Порівняно з попередніми роками ситуація дещо покращилася.

За останні роки виконавча влада значно вдосконалила свій офіційний веб-сайт. Однак більшість джерел не відповідають вимогам щодо публікації повної та



актуальної інформації про діяльність відповідних органів та не пристосовані до потреб громадян.

Що стосується частоти оновлення інформації, останнім часом відбулися суттєві позитивні зміни, інформація постійно оновлюється, але часто недостатньо впроваджена, а саме нові нормативні акти відсутні або не опубліковані вчасно (особливо на регіональних національних веб-сайтах), недостатня інформація про цільові програми.

Каталог послуг є офіційним на більшості веб-сайтів, його зміст не відповідає існуючим вимогам, найчастіше відсутні зразки документації та інших матеріалів, необхідних громадянам.

Більшість веб-сайтів перебувають між першим та другим етапами розвитку електронного уряду, а це означає, що веб-сайт містить лише основну інформацію про орган, його структуру, нормативну базу, доступні лише основні інтерактивні послуги. Подекуди для зворотного зв'язку доступні адреси електронної пошти, використовуються гостьові книги, послуги електронної пошти для зареєстрованих користувачів, форми електронної пошти для посадових осіб із особливими питаннями іноді використовуються як "громадський (Інтернет) прийом". Деякі веб-сайти мають "гарячі лінії", розклад яких заздалегідь вказано для перегляду на веб-сайтах. Також існують форуми, на яких обговорюються актуальні теми, пов'язані з діяльністю цього органу, опитування відвідувачів сайту, як працює сайт, його дизайн, структура тощо. Такі розширені ресурси зворотного зв'язку містять дуже мало ресурсів, майже ніде не публікується, які заходи було вжито для задоволення відповідних потреб громадян.

Більшість проаналізованих веб-сайтів, доступні лише українською мовою, і лише веб-сайти Міністерства фінансів доступні трьома мовами. П'ять веб-сайтів міністерств доступні українською та англійською мовами. Деякі веб-сайти інших міністерств, хоча пропонують англійську та російську версії, є неактивними або майже повністю порожніми. Не найкраща ситуація в цьому питанні на веб-сайтах облдержадміністрацій.

Відкритість інформації органів державної влади в цілому залишається недостатньою. Потрібно досягти чіткого впровадження існуючих законів та

нормативних актів, що важко без активного впливу громадськості та конструктивної роботи дослідницьких центрів.

Відповідно до Закону України "Про доступ до публічної інформації" розпорядники інформації зобов'язані оприлюднювати інформацію, яка направлена на створення ефективного механізму гарантування максимального доступу до інформації, що знаходиться в органах публічної влади.

У статті 15 зазначеного Закону є перелік відомостей, які підлягають обов'язковому оприлюдненню, а також порядок і вимоги до такого оприлюднення. Цей перелік інформації, що підлягає оприлюдненню, є мінімальним – розпорядник може оприлюднювати іншу визначену ним інформацію.

Постановою Кабінету Міністрів України від 04.01.2002 № 3 "Про Порядок оприлюднення у мережі Інтернет інформації про діяльність органів виконавчої влади" з метою поліпшення умов для розвитку демократії, реалізації громадянами конституційних прав на участь в управлінні державними справами і на вільний доступ до інформації про діяльність органів виконавчої влади, а також забезпечення гласності та відкритості діяльності цих органів, визначено порядок та вимоги щодо розміщення і постійного оновлення міністерствами, іншими центральними та місцевими органами виконавчої влади інформації на офіційних веб-сайтах у мережі Інтернет.

Моніторинг інформаційного наповнення офіційних веб-сайтів органів виконавчої влади здійснюється на виконання постанови Кабінету Міністрів України від 29.08.2002 № 1302 "Про заходи щодо подальшого забезпечення відкритості у діяльності органів виконавчої влади" та згідно з Порядком проведення Держкомтелерадіо моніторингу інформаційного наповнення офіційних веб-сайтів органів виконавчої влади, затвердженого наказом Державного комітету телебачення і радіомовлення України від 08.06.2015 № 118, зареєстрованого в Міністерстві юстиції України 25.06.2015 за № 759/27204.

Двічі на рік (квітень - червень та жовтень - грудень) Держкомтелерадіо здійснює моніторинг інформаційного наповнення офіційних веб-сайтів органів виконавчої влади.

Метою моніторингу є оцінка змісту офіційних веб-сайтів органів виконавчої влади відповідно до нормативно-правових актів, що регулюють розміщення певних видів інформації на веб-сайтах.

Суб'єкти моніторингу: Офіційні веб-сайти міністерств, інших центральних органів виконавчої влади, районних та державних адміністрацій міста Києва.

Пошук, класифікація та аналіз інформації на веб-сайтах здійснюється за переліком параметрів, кожен з яких відображає певний тип інформації, обов'язок публікувати на веб-сайтах визначається законодавством.

Зміст кожного веб-сайту оцінюється за такими показниками:

Доступ до інформації, що визначає рівень доступу до інформації на сайті;

Якість інформаційного змісту, що враховує ступінь повноти, релевантності, навігаційну доступність інформації, опублікованої на веб-сайті, а також доступність функції пошуку та доступність сприйняття інформації людьми з вадами зору та слуху;

Інформаційна прозорість, яка вказує на рівень інформаційної прозорості енергетичного суб'єкта та обчислюється як середнє значення двох попередніх показників;

Динаміка прозорості інформації, яка показує, чи змінився рівень прозорості інформації енергетичного суб'єкта за звітний період порівняно з попереднім періодом, і визначається як різниця між значеннями відповідних показників прозорості інформації.

**Висновок до розділу:** провівши аналіз наявної наукової літератури, зроблено висновок що чіткої класифікації сучасних веб-ресурсів немає, вони характеризуються тим що водночас можуть об'єднувати в собі характеристики різних типів та видів.

За типами веб-ресурси поділяються за доступністю користувачам, за величиною та якістю аудиторії, за розміром, метою створення та функціональністю тощо.

За видами веб-ресурси бувають інформаційними, загально тематичними, новинними, блогами, корпоративними сайтами, іміджевими, промо-сайтами, форумами, тощо.

Було визначено що існує три основні типи структури веб-ресурсів: лінійна, ієрархічна та мережна структури.

Проаналізувавши класифікації та типи інтерфейсів, принципи та специфіку їх проектування, а також вивчивши особливості проектування інтерфейсу державної установи, можна зробити висновок, що все більшого поширення та стандартизації на державному рівні набувають доступні та прозорі інформаційні ресурси. Також було визначено, що веб-ресурс державної установи "Київський міський центр громадського здоров'я" належить до некомерційного веб-ресурсу; за доступністю є загальнодоступним, тобто відкритим; за функціональністю – динамічним; за розміром – великий веб-ресурс. Має типову структуру: присутні шапка, підвал, бічна панель, блоки основної інформації. Серед розглянутих функціональних компонентів для проектування інтерфейсу державної установи "Київський міський центр громадського здоров'я" вибрано наступні компоненти: новини, документи, галерея, інформаційні блоки, статті, пошук по сайту, хлібні крихти, мультимедіа, галерея зображень. Також додатково додана система допомоги для людей з обмеженими можливостями.

## РОЗДІЛ 2. ТЕХНОЛОГІЇ ПРОЄКТУВАННЯ ІНТЕРФЕЙСУ

### 2.1 Програмні засоби для проєктування інтерфейсу

В процесі розробки інтерфейсу можна виділити такі основні етапи, а саме дослідження, користувацькі сценарії, початкове проєктування, створення прототипу і тестування або модифікація прототипу. Фактично процес розробки, щоб бути успішним і безумовним, завжди повинен відбуватися в цій послідовності: проєктування, потім створення прототипу, потім нескінченні цикли тестування або модифікація до досягнення задовільного результату або до тих пір поки не буде досягнуто ідеалу. Тобто основним етапом виявляється не проєктування, та не власне дизайн, але поліровка вже зробленого дизайну.

На етапі дослідження проводиться збір інформації про продукт, клієнта, його конкурентів або близьких аналогів, збір статистики використання поточного інтерфейсу (наприклад, сайту або мобільного додатку), аналіз пристроїв передбачуваної цільової аудиторії.

Якщо вже відомо, хто буде втілювати інтерфейс в життя (розробники), то знайомимося з ними і з'ясуємо їх можливості і обмеження.

Цей етап допомагає зрозуміти для кого розробляється інтерфейс, з якими обмеженнями слід його робити, як не варто робити

. На основі наданого опису роботи інтерфейсу створюється список завдань, призначених для користувача сценаріїв, які може виконувати користувач в рамках інтерфейсу. Наприклад, оновити аватарку в профілі.

Всі завдання розписуються по кроках, які необхідно вжити для вирішення завдання. наприклад:

Зайти на сайт

авторизуватись

Перейти в профіль

Натиснути на аватарку

вибрати файл

Підтвердити або змінити кадрювання зображення

зберегти

Складені списки кроків для кожного завдання допомагають зрозуміти де шлях для вирішення занадто довгий щодо інших завдань. Етап користувальницьких сценаріїв найбільше підходить для скорочення шляху вирішення завдань користувачів в рамках інтерфейсу.

У більшості випадків ми робимо два схематичних прототипу: чорновий і фінальний. Винятки становлять невеликі інтерфейси: простенькі мобільні додатки або маленькі сайти.

Чернової прототип являє собою схематичні зображення екранів, пов'язані між собою. При чорновому варіанті на схемах зображені зони і опису цих зон. Наприклад, список новин або шапка сайту. Всі без деталей.

Чернової прототип допомагає більш наочно зрозуміти на скільки об'ємним буде сайт, як багато інформації буде на кожному екрані, як багато потрібно клікати, щоб дістатися до потрібної сторінки.

Наступним кроком йде фінальний прототип, в якому схеми сторінок все ще залишаються пов'язаними між один одним, але на сторінках вже видно всі кнопки, тексти, чекбокси, форми та інші елементи.

В прототипах планується функціонал, розташування елементів сторінок відносно один одного, але ніяк не оформлення. Кольори, зображення, іконки - це все етап оформлення. На етапі проектування неможливо сказати як вони будуть взаємодіяти між собою, як будуть виглядати разом, чи будуть перекрикувати один одного.

Кращі інструменти проектування призначеного для користувача інтерфейсу можуть підійти практично для кожного процесу проектування і, сподіваюся, задовольнити ваші творчі вимоги. Тепер саме час вивчити нові інструменти і подивитися, що можна спроектувати і продемонструвати свої проекти.

Для вайрфреймів частіше всього використовують або програму в якій буде проектуватися інтерфейс, проте іноді використовують і спеціалізоване програмне забезпечення таке як: MockFlow, Miro, Axure. Розглянемо їх детальніше:

MockFlow – це онлайн набір програм, які дуже корисні для ряду завдань у типовому процесі створення проекту інтерфейсу. В першу чергу, програма WireframePro - це хороший альтернативний інструмент для створення прототипів, особливо якщо ви перевіряєте нові ідеї. Вона безкоштовна проте має обмеження в один проект, який включає в себе всього 3 сторінки та дві ітерації виправлень

проекту. Проте там є і підписка на цей онлайн ресурс, який знімає усі обмеження та додає додаткові функції до безкоштовного плану.

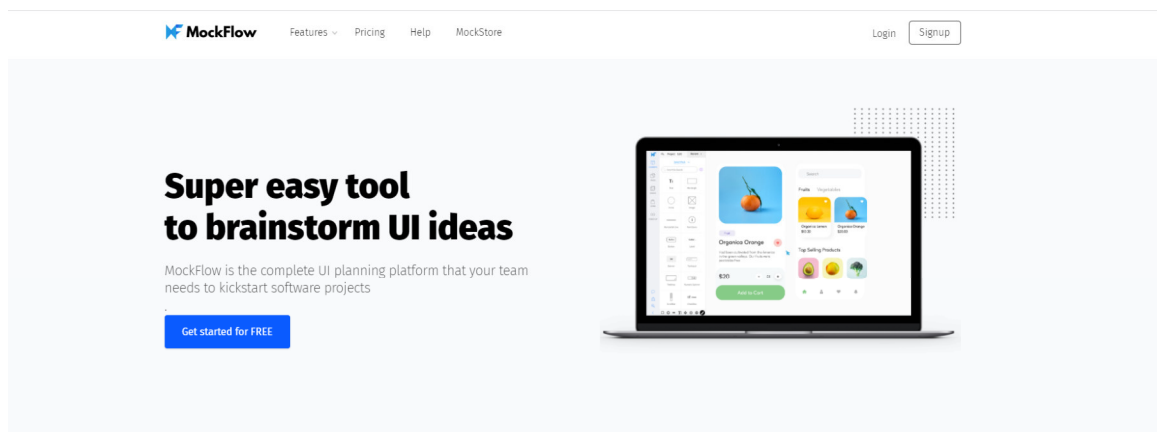


Рис 2.1 MockFlow

Miro також безкоштовний онлайн ресурс для роботи на початковому етапі створення інтерфейсу, і є прекрасним варіантом для команди з декількох чоловік, адже дає можливість одночасно працювати над одним проектом декільком користувачам одночасно. А також має багато додаткових функцій для проектування взаємодії користувачів у розроблюваному продукті.

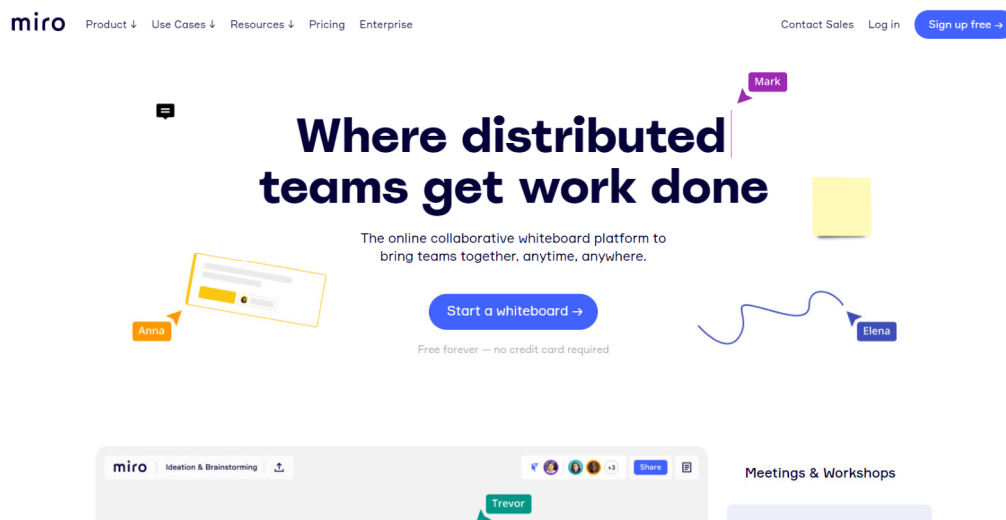


Рис 2.2 Miro

Ахиге завжди був одним з найкращих інструментів створення вайрфреймів на ринку і чудово підходить для більш складних проектів, що вимагають динамічних даних. За допомогою Ахиге ви дійсно можете зосередитись на прототипуванні над проектами, які є більш технічними та вимагають додаткової уваги, коли справа стосується структури та даних.

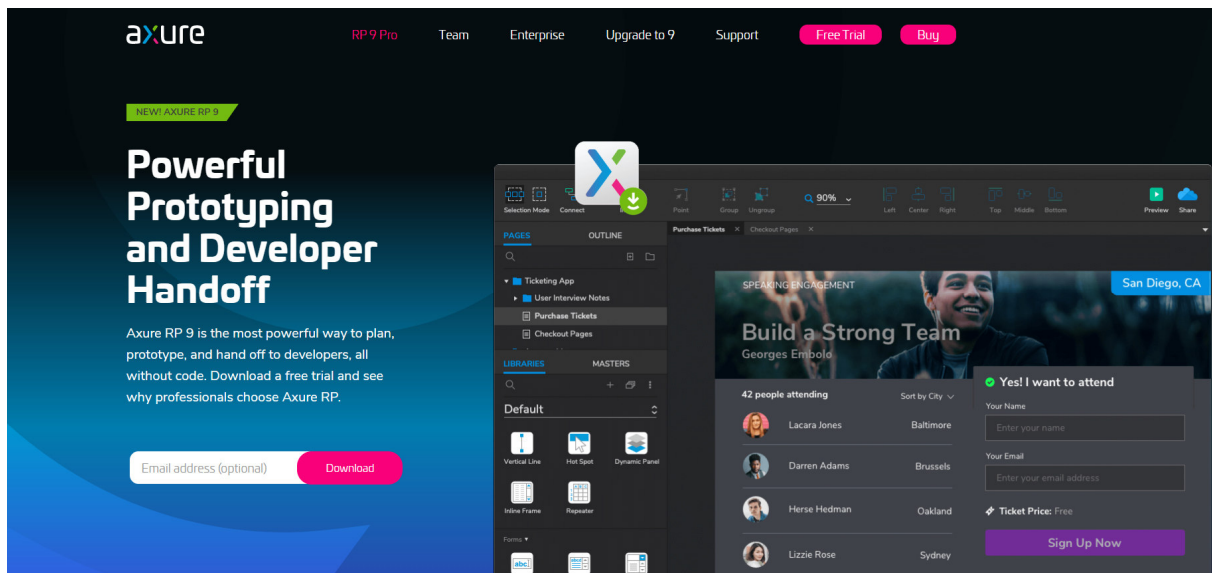


Рис 2.3 Axure

Після створення прототипів та проходження всіх етапів тестування вайрфреймів, їх редагування та полірування, наступає етап створення інтерфейсу, який створюється на основі фіналізованих екранів. Зазвичай для створення інтерфейсу використовують декілька найпопулярніших програм серед яких найбільше виділяють – Sketch, Figma.

Sketch - це редактор векторної графіки для macOS, він в основному використовується для користувальницького інтерфейсу та дизайну веб-сайтів та мобільних додатків, а також не включає функції дизайну друку. Ескіз нещодавно додав функції для створення прототипів та співпраці. Будучи доступним лише для macOS, програмне забезпечення сторонніх розробників та інструменти передачі можуть використовуватись для перегляду дизайну Sketch на інших платформах.

Sketch використовується в основному для проектування інтерфейсу та UX мобільних додатків та Інтернету. Файли, розроблені в Sketch, зберігаються у власному форматі файлу .sketch, хоча файли .sketch можна відкрити в Adobe Illustrator, Adobe Photoshop та інших програмах. Роботу також можна зберігати у популярних форматах PNG, JPG, SVG, PDF, TIFF, WebP тощо. Проекти, створені в Sketch, використовуються інженерами додатків для розробки мобільних додатків, а розробники веб-сайтів перетворюють їх у веб-сайти.

Незважаючи на те, що Sketch раніше продавався через App Store, розробники витягли програму з магазину в грудні 2015 року і натомість продали через власний веб-сайт. Причинами рішення вони називали суворі технічні вказівки Apple, повільний процес перевірки та відсутність ціни на оновлення. 8 червня 2016 року вони повідомили у своєму блозі, що переходять на нову систему ліцензування для Sketch. Ліцензії дозволять користувачам отримувати оновлення протягом 1 року,



після чого вони можуть продовжувати використовувати останню версію, опубліковану до закінчення терміну дії ліцензії, або поновлювати свою ліцензію, щоб продовжувати отримувати оновлення ще протягом року.

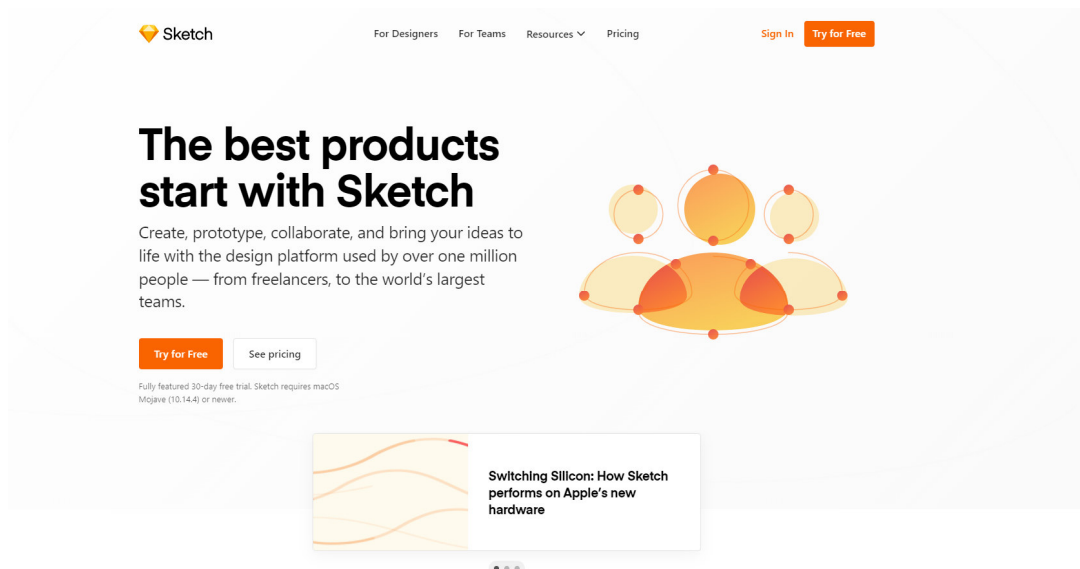


Рис 2.4 Sketch

Figma - це векторний графічний редактор та інструмент для створення прототипів, який в основному працює на Інтернеті, з додатковими функціями в автономному режимі, увімкненими робочими програмами для macOS та Windows. Супутні додатки Figma Mirror для Android та iOS дозволяють переглядати прототипи Figma на мобільних пристроях. Набір функцій Figma зосереджений на використанні в інтерфейсі користувача та дизайні користувачького досвіду, з акцентом на співпраці в режимі реального часу.

Figma дозволяє швидко складати та проектувати інтерфейси. Платформа Figma пишається тим, що є інструментом спільного проектування, завдяки якому кілька користувачів можуть одночасно працювати над проектом - це дуже ефективно, коли у вас є кілька зацікавлених сторін у проекті, які беруть участь у формуванні результатів. Це такий інструмент, який був би ідеальним, якщо у вас є живий проект, де розробнику, копірайтеру та дизайнеру, наприклад, потрібно працювати над чимось одночасно.

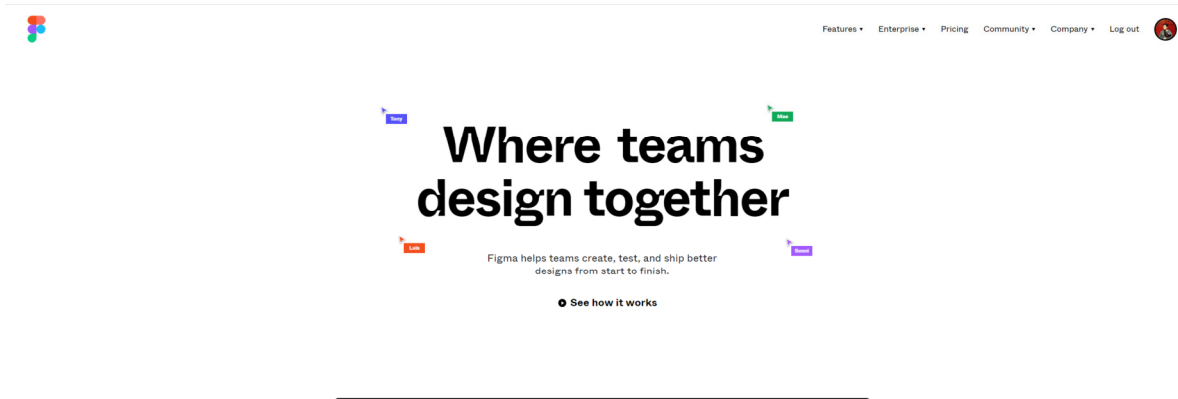


Рис 2.5 Figma

## 2.2 Принципи побудови архітектури веб-ресурсу та user-flow

В процесі розробки інтерфейсу можна виділити три основні етапи, а саме початкове проектування, створення прототипу і тестування або модифікація прототипу. Фактично процес розробки, щоб бути успішним і безумовним, завжди прагне відбуватися в цій послідовності: проектування, потім створення прототипу, потім нескінченні цикли тестування або модифікація до досягнення задовільного результату або до тих пір поки не зупинять. Тобто основним етапом виявляється не проектування, тобто власне дизайн, але поліровка вже зробленого дизайну.

З іншого боку, при ретельному проектуванні тривалого тестування зазвичай вдається уникнути - але, з іншого боку, при цьому проектування стає досить тривалим, так що невідомо ще, що краще скорочувати. Етап проектування сам по собі складається з декількох складових, причому кількість цих складових досить велике.

В процесі проектування вам знадобляться не тільки цифрові а й аналогові інструменти - а саме ручка і папір. Справа в тому, що використання комп'ютера само по собі повільно, по-перше, оскільки інтерфейс програм недосконалий, а по-друге, через те, що, використовуючи комп'ютер, ви будете підсвідомо намагатися зробити роботу красиво, а не просто будете фіксувати свою думку.

На етапі проектування будується багато теорії, гіпотез і умоглядних висновків, які належить підтвердити або спростувати. Ці висновки стосуються функціональності продукту і є наслідком питань: "Навіщо потрібен цей продукт?", "Кому він потрібен?", "Як з ним будуть працювати і вирішувати завдання користувачі?" і "Як він буде заробляти для своїх власників?".

Відповісти на більшу частину цих питань допоможе складання портрета цільової аудиторії (ЦА) - тих самих людей, для яких робиться продукт.

Головне завдання дизайнерів при вивченні аудиторії - включити емпатію на максимум і зрозуміти, як ця аудиторія думає, дихає, бачить, чує і діє. Цьому сприяють такі методи:

Коридорний метод. Зворотній зв'язок надходить від рідних, друзів і колег дизайнерів. Зібрати її легко, але цього недостатньо.

Розмова з вами. Справедливо передбачається, що ви як ніхто знаєте, що потрібно вашій аудиторії.

Польові дослідження. В рамках методу дизайнери йдуть в народ: спілкуються з людьми безпосередньо, якщо роблять продукт для місцевого ринку, або читають форуми, якщо для зарубіжного;

Проблемне інтерв'ю. Ставлячи користувачам питання про їхнє життя і місце проблеми в ній, дизайнери дізнаються, як ця проблема вирішується зараз і наскільки корисним виявиться їх продукт. Те, що він може виявитися марним - теж цінний результат: не доведеться витратити гроші на додаток, яким ніхто не буде користуватися.

Зібрану інформацію дизайнери переробляють і отримують, по-перше, ключові персони, а по-друге, призначені для користувача маршрути.

Ключові персони - це характерні представники ЦА. Вони можуть бути різними за професією, рівнем життя, мотивації користуватися додатком і іншим параметрам, але досвід, очікування і страхи кожної персони лягають в основу зовнішнього вигляду продукту і його функціональності. Наприклад, типовому користувачеві додатки "Кіноголік" для покупки абонементів в кіно 23 роки, він працює в ІТ-компанії і любить дивитися фільми англійською.

Така персона стає центром user story, або користувальницької історії. Це короткий, в кілька рядків, розповідь про персону і то, як вона працює з функціональністю програми та якої мети досягає. User story будується за шаблоном:

"Як <роль користувача>, я <щось хочу отримати> <з такою-то метою>"

Помістивши нашого 23-річного фаната оригінальних версій в цей шаблон, отримаємо:

"Як <23-річний любитель версій з оригінальною озвучкою>, я <шукаю через функцію" Пошук "фільми з субтитрами"> <щоб сходити фільм англійською>"

Компанія Intercom славиться не тільки комплексно вирішувати по впровадженню чатів в сайти і мобільні додатки, але і винаходом підходу Jobs To Be Done. В основі підходу лежить не особиста якість ключовий персони, а обставини і мотивація, які штовхають персону користуватися продуктом. "Роздуми" персони називаються Job story, а шаблон виглядає так:

"Коли <я опинився в такій ситуації", я хочу <щось зробити з деяких причин> <з такою-то метою> "

Ситуація з кінолюби в рамках такого підходу виглядає інакше:

"Коли <мене дратують відвідувачі кінотеатрів, жують попкорн і заважають дивитися фільм своїми розмовами>, я <шукаю в розкладі оригінальну версію фільму", "щоб посидіти в напівпорожньому залі, де ніхто не бубонить">

Детальніше про підхід Jobs To Be Done написала в своєму блозі платформа Tilda.

Від User story і Job story ми переходимо до User scenario. Це маршрут взаємодії користувача з продуктом і досягнення мети.

У гонитві за основною метою (покупка, додавання фотографії) користувач може вирішувати допоміжні завдання (визначає кур'єра замовлення, редагує фото) і досягати вторинних цілей (зручне отримання замовлення, фото з контрастують деталями); ці додаткові маршрути дизайнеру теж необхідно врахувати.

User flow дозволяє поглянути на взаємодію користувача і додатки очима користувача. Якщо повноцінні сайти надають достатньо місця для творчості, то розробка мобільних додатків завжди змушує шукати баланс між функціональністю і зручністю використання. З одного боку, користувачі повинні легко отримувати доступ до найбільш важливих функцій з головного екрану програми без безлічі додаткових кроків, легко натискаючи на великі кнопки. З іншого боку, коли ми говоримо про додатки, мова йде не тільки про кнопках.

Припустимо, ви створюєте додаток для музичного плеєра. З одного боку, ви хочете, щоб користувач міг відтворювати свої улюблені пісні одним дотиком. З іншого боку, їм потрібен спосіб визначати обрані треки, шукати нову музику (можливо, в декількох місцях), контролювати гучність і налаштовувати еквайзер, сортувати музику, шафл список, читати біографію виконавця, робити покупки і мати під рукою всі інші функції плеєра. Користувачеві захочеться ділитися рекомендаціями і плейлистами в соцмережах і бачити, що слухають їхні друзі.

Якщо користувач шукає в інтернет-магазині під час прослуховування плейлиста і вирішує пропустити поточний трек, йому не потрібно закривати магазин і переходити назад через кілька кроків меню - доступ до цієї кнопки пропуску повинен знаходитися на тому ж екрані.

Створити максимально функціональне додаток не так складно, але зібрати все це в найбільш привабливому і комфортному для користувача вигляді - набагато складніше завдання. Якщо компанія-розробник не впоралася з нею і не створила додаток інтуїтивним, то воно так і залишиться лежати незатребуваним в сторах. Користувачеві завжди легше завантажити більш зрозумілий додаток, ніж витратити час на читання мінлива і туторіали або без кінця бродити по ньому в пошуках сенсу.

Уявіть, що ви працюєте над продуктом в одній команді, де є Project Manager, Backend- і Web-розробники. Припустимо, є навіть якась технічне завдання від замовника, яке написано досить докладно і доброту. Залишилося просто взяти і зробити продукт.

Що повинен зробити дизайнер? Якщо судити по очікуваного результату - намалювати все потрібні екрани так, щоб кінцевий продукт був зручним і красивим, і віддати розробникам.

В цілому - так. Але є підводна частина айсберга. Там приховано безліч речей, про які мало хто замислюється. Всі ми любимо зручні та інтуїтивно зрозумілі інтерфейси, любимо, коли все працює логічно і результат будь-якої взаємодії передбачуваний. І щоб все це було, потрібно ретельно опрацювати кожен деталь.

Якраз для цього потрібен user flow - він допоможе визначити шляхи досягнення користувачами цілі, обчислити позитивні і негативні сценарії на обраному шляху досягнення мети наших потенційних користувачів. Він дає можливість зрозуміти, чи всі процеси в продукті мають логічне завершення і збудовані ефективно, так, щоб користувач витрачав мінімум часу для досягнення мети.

Важливо: чим більше проблема і чим більше функцій, тим більше сценаріїв необхідно опрацювати.

Таким чином, user flow допомагає виключити поширену помилку, коли створюють список необхідних функцій і на основі них просто проектують інтерфейси. Такий підхід, зрозуміло, можливий. Але це може привести до того, що сценарії взаємодії з інтерфейсом змусять користувача проходити занадто багато кроків для виконання простих операцій.

User flow допомагає на верхньому рівні погоджувати з замовником сценарій для роботи продукту або функціоналу. Особливо тоді, коли у замовників немає конкретних і детальних вимог (коли вони самі погано розуміють, чого вони хочуть). В цьому випадку user flow може дати якийсь напрямок для замовників, і вони вже підкажуть, в яких місцях вони хотіли б змінити логіку, надавши свою зворотний зв'язок - або у вигляді коментаря, або самостійно скорегувавши поточну схему. Незважаючи на те що створення user flow здається складним і тривалим за часом процесом, він допомагає виключити безлічі помилок при розробці продукту. З його допомогою, як то кажуть, "повільно запрягають, але швидко їдуть".

Для РМ на підставі user flow зручно заводити завдання для розробників - досить очевидно і прозоро, які зміни необхідно внести в поточну логіку або які нові функції будуть додані.

Тестировщики можуть готувати use cases (тести) і чек-листи, оскільки вони в зручному форматі демонструють призначені для користувача сценарії.

User flow дуже корисний для команди, особливо коли склад команди досить великий - не потрібно кожному окремо пояснювати логіку роботи. Таким чином, user flow залучає до процесу проектування замовника, мотивує команду, направляє весь робочі процеси в єдине русло, оскільки кожен член команди бачить картинку в цілому.

User flow може бути представлений у вигляді комбінацій з:

- **task flow** - просте уявлення того, що користувач робить на кожному кроці для виконання мети або завдання. По суті це класична блок-схема, яка визначає цю послідовність. Для складання схеми можна користуватися списком необхідних і достатніх елементів з класичних блок-схем, представлених на зображенні, або використовувати стандарт BPMN - він орієнтований як на технічних фахівців, так і на бізнес-користувачів.
- **wire flow (lo-fi)** - Wire flow - це об'єднання блок-схеми і wireframes. Wireframe - це нізкодеталізований начерк дизайну екрану, упор в якому робиться не на візуальну складову, а на розташування елементів, структуру і зміст екрану. У wire flow замість елементів блок-схеми представлені схематичні макети екранів, з якими взаємодіє користувач на шляху досягнення мети. Не потрібно зациклюватися на візуальних деталях і малювати кожную кнопку і іконку. Найчастіше акцент робиться на елементах навігації в дизайні кожної окремої сторінки. Wire flows особливо зручні при створенні мобільних екранів: відносно невеликі розміри мобільних екранів легко використовуються для заміни більш

абстрактних елементів блок-схем. Wire flows комфортні для формування і уточнення технічного завдання, але навряд чи підходять для етапу тестування.

- **screen flow (hi-fi)** - тут мова йде про детально опрацьованих екранах, які зрозумілі як користувачам, так і розробникам. Зазвичай робиться акцент на елементах навігації і деяких нюансах поведінки. Його можна використовувати як регламентує документ для затвердження дизайну макетів. Screen flow можна по суті назвати прототипом, якщо йому додати інтерактивність. Його особливістю є висока точність або ідеальне піксельний відповідність: в ньому враховується фізичний розмір екрану і представляються всі візуальні і типографические деталі продукту. Елементи screen flow - це фактично макети екранів готового додатка. Замовник програми може бути приємно здивований швидкістю розробки, коли ви покажете йому screen flow, і може подумати, що розробка близька до завершення. Ми, як правило, заздалегідь повідомляємо замовнику, що за screen flow не варто ніякого коду, але наявність screen flow дає замовнику гарантію, що за відмінно підготовленими UX і UI піде не менше ідеальний код.

Таким чином, user flow, охоплюючи весь продукт, систематизує і структурує процес його створення, максимально орієнтує продуктову команду на потреби користувача і на їх єдине розуміння, дозволяє створити гранично зрозумілий і інтуїтивний продукт за допомогою аналізу користувальницьких сценаріїв. Все це не тільки істотно прискорює швидкість розробки, а й поглиблює взаємодію з замовником. User flow змушує ІТ-компанію завжди думати про бізнес-вимогах замовника, щоб створювати додаток більш ефективним і успішним, ніж це можливо в рамках виділеного бюджету і часу без застосування user flow.

### 2.3 Wireframe як основа якісного UX

Вайрфрейм - це низько деталізований уявлення дизайну. Він чітко повинен показувати:

Основні групи вмісту. Що?

Інформаційну структуру. Де?

Опис взаємодії користувача з інтерфейсом і його приблизну візуалізацію. Як?

Вайрфрейм - не просто безглуздий набір сірих блоків, хоча це виглядає саме так. Вважайте це скелетом вашого дизайну і запам'ятайте, що в вайрфрейме повинні бути представлені всі важливі елементи кінцевого продукту.

"Представлені" - це ключове слово, яке допоможе вам знайти правильний баланс між рівнем деталізації і швидкістю створення. Не можна йти в деталі, але з іншого боку, потрібно створити цілісне уявлення кінцевого дизайну і не пропустити жодного важливого елемента. Ви описуєте фронт робіт по проекту для всіх задіяних осіб: розробників, дизайнерів, копірайтерів, менеджерів - всім їм потрібен добре спрацьований вайрфрейм. По суті, ви створюєте карту міста. Кожна вулиця має на ній бути, але в дуже спрощеному вигляді. Дивлячись на карту, ви можете оцінити планування міста, але не можете побачити його красу.

Для того, щоб почати роботу над будь-яким сервісом, потрібно розуміти весь список функціоналу, який потрібно для реалізації проекту. Потім потрібно визначити бізнес-цілі і творчі ідеї проекту, кількість сценаріїв і правила відображення елементів (зазвичай надаються з боку бекенд)

Wireframe повинен відповідати наступним принципам:

Real Size - необхідно використовувати реальні розміри робочої області, щоб розуміти, яка кількість елементів буде оптимальним для нього.

Storytelling - не варто забувати, що wireframing - це перш за все призначений для користувача шлях. Історія, яку ви доносите до користувача повинна бути проста і зручна.

Animation - на етапі вайрфреймінга варто відразу продумати, як будуть вести себе елементи інтерфейсу, переходи між екранами, інпут, дропдауни і т.д., де будуть тапи, а де свайпи.

Clear - wireframe повинен відповідати всім вимогам BDD (behavior-driven development), так як після закінчення роботи над вайрфреймом, він буде служити для користувача сценарієм для відділу розробки.

Вайрфрейми повинні створюватися швидко і більшу частину цього часу слід провести за обговореннями з командою і роздумами. Саме тягання сірих блоків по екрану повинно займати мінімум часу.

Зовнішній вигляд повинен бути естетичним, але дуже простим. Чорно-сіро-білий - типова палітра вайрфрейма (ви можете додати синій, щоб позначити посилання).



Якщо, наприклад, вибір піктограм або завантаження картинок займає надто багато часу, їх можна замінити заглушками - навхрест перекреслено прямокутниками з відповідним описом. Хороший вайрфрейм лягає в основу чистового дизайну і визначає напрямок роботи для всієї команди.

Зазвичай вайрфрейми використовуються як документація по проекту. Так як вони показують взаємодію користувача з інтерфейсом в окремих статичних моментах, їх потрібно супроводжувати текстовими коментарями: як короткими поясненнями, так і комплексної технічною документацією, при необхідності.

Однак вони можуть використовуватися менш формальним способом. Так як вони прості і швидко створюються, вони, як замальовки, добре підходять для обговорення всередині команди. Якщо розробники запитують, як щось має бути зроблено, відповідь може бути дан у вигляді швидко зробленого вайрфрейма.

Вайрфрейми складно пристосувати для юзабіліті-тестування, хоча вони можуть стати в нагоді для отримання відгуків під час початкового "партизанського" дослідження, коли вам терміново потрібно осяяння і вас не турбує методологічна чистота. У загальному процесі дизайну вайрфрейми можуть бути несподівано ефективні і, хоч в останні роки про них відгукувалися не дуже, вони залишаються важливим початковим етапом складних проектів.

Тобто wireframes - це організація повного функціоналу кінцевого продукту, у вигляді структури з поданням елементів інтерфейсу і навігації, їх взаємодії один з одним. Як правило, в wireframes не використовується типографіка, колір і будь-які інші графічні елементи оформлення, так як основна увага приділяється функціональності, поведінки і змістом. Wireframes фокусується на тому, як екрани взаємодіють між собою, а не на те, як вони виглядають. Основне правило вайрфреймінга - не сприймати його, як каркас, на який після буде натягнутий дизайн, це тільки набір обов'язкового функціоналу.

## **2.4 Етапи технології проєктування інтерфейсу**

Інтерфейс - система правил і засобів, що регламентує і забезпечує взаємодію декількох процесів або об'єктів. Інтерфейс (ПІ) - система правил і засобів, що регламентує і забезпечує взаємодію програми з користувачем. Типовий ПІ має пристрої введення і виведення. Отримавши від користувача команду, інтерфейс "відповідає" йому, виводячи різного роду інформацію. Подання інформації має на увазі вибір формату, носія, структури, композиції і візуальних прийомів. Виходячи зі стратегій розробки призначеного для користувача інтерфейсу, можна виділити

наступні етапи роботи над UX-проектом. Перед початком роботи над проектом формуються цілі і завдання, наступним пунктом є аналіз бажаної аудиторії і цільової аудиторії. На основі перших двох параметрів опрацьовуються варіанти можливої взаємодії з тим розрахунком, щоб максимально задовольнити попит. Уже виходячи з цього, проектується структура продукту і її зовнішній вигляд. UX / UI дизайн допомагає зробити додаток комфортніше для користувача.

Юзабіліті, як термін, прийшов в оборот під час розробки першого людино-орієнтованого інтерфейсу і визначається як ступінь ефективності, зручності, продуктивності, при якій певний користувач вирішує певні завдання в певному контексті.

У веб-технологіях цей термін прийнято розуміти як притаманну будь-якому проекту легкість використання. Легкість же використання виводиться на підставі конкретно взятого дизайну, навігації, архітектури та інших функціональних особливостей проекту. Багато хто помилково вважає, що "юзабіліті" - це всього лише простий і зрозумілий інтерфейс. Простота і зрозумілість - це тільки один з аспектів технологічних рішень, які використовують веб-майстри і програмісти для додання своїм продуктам характеристик високоякісних показників в плані юзабіліті.

Одним з основних аспектів, на яких слід загострювати увагу при створенні гарного юзабіліті, - це дизайн. Від того, наскільки якісним буде дизайн вашого продукту, залежить дуже багато чого. Саме вдалі дизайнерські рішення створюють один з елементів хорошого юзабіліті - професійний дизайн, який робить проект зручним для вивчення, навігації та інших дій користувача. Дизайн створеного продукту повинен відповідати певним стандартам. Звична схема розташовує до більшості користувачів, як наслідок, користуватися таким шаблонним інтерфейсом набагато зручніше, ніж "наворочений" або ж незвичним. У сучасному дизайні існують для протилежні напрямки: скевоморфізм і плоский дизайн, мають свої переваги і недоліки.

Скевоморфізм - це фізичний орнамент або елемент дизайну, який скопійований з форми іншого об'єкта, але виготовлений з інших матеріалів або іншими методами. Як протилежності, зворотною стороною монети можна вважати недавно став популярним "плоский стиль", головним прикладом якого є призначений для користувача інтерфейс Microsoft Metro. Плоский стиль втілює в собі візуальний мінімалізм, мінімум текстур і ефектів підсвічування для простих форм і плоских квітів. Однак, щоб бути більш точним, цей тренд не завжди стосується скевоморфізм, який часто має на увазі зв'язок з минулими втіленнями подібного дизайну, але досить часто пов'язаний з реалізмом: чисто візуальним стилем, який намагається імітувати

реальні матеріали і текстури, продемонстрованим надмірним позбавленим смаку використанням текстур в деяких додатках Apple. До появи iPhone реалізм в інтерфейсі був чимось незвичайним, за винятком відеоігор. З метою зберегти ефект занурення, розробники ігор давно почали робити інтерфейси під дерево, метал і камінь. Але в більш "серйозному" контексті реалізм все одно був обмежений. Єдиним примітним винятком були скляні кнопки "в стилі Web 2.0", успадковані з попередніх версій Mac OS X. Насамперед, неможливо ігнорувати візуальну привабливість реалізму. Складні текстури і анімація при перевертанні сторінок сьогодні здаються нудними, після того, як останні пару років ми користуємося ними постійно, але коли iPhone вийшов вперше, ніхто до цього не бачив такого візуального багатства в інтерфейсі системи (тим більше, на телефоні). Технічні обмеження середовища завжди будуть впливати на мистецтво, яке середовище може створювати, і це так само вірно щодо смартфонів, як і щодо олійної фарби. Дійсно, існує кілька практичних факторів, які змусили реалізм працювати на iPhone. Також існує проблема в неправильному розумінні скевоморфізм: наприклад, змушуючи щось виглядати як фізичний об'єкт, а не працювати як він. Для передбачення трендів в дизайні виникли цілі галузі в різних сферах. І, незважаючи на це, ніхто не очікував, що таке рух в дизайні з'явиться завдяки компанії Microsoft. Щоб сильніше відрізнятись від Apple або тому, що вони спочатку вважали це кращою філософією дизайну, Microsoft вибрали абсолютно іншу естетику для свого призначеного для користувача інтерфейсу в Metro-стилі. Пропали тіні, підсвічування, градієнти і текстури iOS-додатків. Замість цього Metro пропонує плоскі кольорові квадрати з великим шрифтом. Незважаючи на те, звертає чи основна аудиторія увагу на великі блоки з плоским кольором чи ні, нова філософія дизайну Microsoft виразно взяла за душу техносферу, вихваляючи особливу увагу Metro-стилю до шрифтів і квітів.

Вплив мобільного веб-простору на дизайн стало відчуватися ще в одному: чуйну дизайні. Чуйний (або адаптивний) дизайн - це концепція, яка має на увазі, що замість створення декількох окремих дизайнів для різних пристроїв, один і той же дизайн повинен адаптуватися під різні екрани. Він може бути в формі трансформації сітки з декількох колонок в одну колонку, зменшення великих шрифтів до більш дрібних, що випадають меню, які активуються у відповідь на дотик до екрану телефону. В першу чергу, як і будь-який інший тенденцією дизайну, їм будуть швидше зловживати, так як багато дизайнерів відразу ж перейдуть на сторону більшості, не замислюючись про свій вибір. Але, якщо основною жертвою реалізму був в основному хороший смак, то надмірне використання мінімалізму може привести до серйозних наслідків в юзабіліті. Користувачі звикли покладатися на велику кількість тонких натяків, які супроводжують їх через весь інтерфейс: невеликий градієнт на кнопках і закруглені кути, м'які внутрішні тіні полів для

заповнення, і "летять" над іншим контентом панелі навігації. Приберіть всі ці натяки, і у вас вийде плоский світ, в якому всі елементи раптово опинилися на одному тому ж рівні, збиваючи вас з пантелику. Візуальний стиль - це не більше ніж засіб досягнення мети. Якщо ситуація вимагає реалізму, розважайтеся з текстурами і тінями. З іншого боку, якщо плоска естетика краще допомагає дизайнерові досягти мети, тоді, напевно, варто відмовитися від градієнтів.

Наступний крок в початковому етапі проектування інтерфейсу: структурування контенту. Від того, якою буде архітектура наданої інформації, залежить багато чого. Теми, розділи, що випадають меню - ці деталі допомагають користувачеві орієнтуватися, а також здійснювати навігацію. При розробці користувацького інтерфейсу аналіз поведінки потенційного покупця, його дій і бажань є невід'ємною частиною початкового етапу проектування. Потенційні покупці (потенційні клієнти) - всі, хто має можливість і здатність купити товари або послуги жодного разу не купували їх, а також ті, хто припинив покупку товарів і (або) послуг з тих чи інших причин, але зберегли можливість їх придбання. Для збільшення числа користувачів продукту слід звертати увагу на їх поведінку: на чому вони затримуються, яким маршрутом вони переміщаються по навігації програми, чому можуть не доходити до цільових функцій. Для детальнішого перегляду поведінки користувача слід звернути увагу на призначені для користувача сценарії.

Сценарії взаємодії - це опис того, як повинні працювати функції системи. Вони можуть розповідати про суть і особливості роботи функцій, як в загальному вигляді, так і в докладному, алгоритмічній. Перший варіант потрібен для того, щоб зрозуміти, навіщо потрібна і що робить функціональність. Другий крок за кроком розписує всі можливі сценарії використання продукту - що може зробити користувач і як повинна відреагувати на його дії система. Діаграми варіантів використання описують функціональне призначення системи або те, що система повинна робити. Розробка діаграми переслідує такі цілі: визначити спільні кордони і контекст модельованої предметної області; сформулювати загальні вимоги до функціонального поведінки проектованої системи; розробити вихідну концептуальну модель системи для її подальшої деталізації у формі логічних і фізичних моделей; підготувати вихідну документацію для взаємодії розробників системи з її замовниками і користувачами.

Суть діаграми варіантів використання полягає в наступному. Проектована система представляється у вигляді безлічі сутностей або акторів, що взаємодіють з системою за допомогою варіантів використання. При цьому актором (actor) або дійовою особою називається будь-яка сутність, що взаємодіє з системою ззовні. Це може бути людина, технічний пристрій, програма або будь-яка інша система, яка може служити джерелом впливу на моделювану систему так, як визначить сам

розробник. Варіант використання служить для опису сервісів, які система надає актору. Діаграма варіантів використання може доповнюватися пояснювальним текстом, який розкриває зміст або семантику. Прикладом автоматизоване може бути докладний опис послідовності дій певної однієї або декількох функцій, які здійснює користувач для досягнення певних цілей. Метою складання автоматизоване є збір функціональних вимог. Сценарії взаємодії дозволяють не тільки перерахувати функції системи, а й у всіх подробицях розповісти про те, як вони працюють, оцінити якість спроектованого інтерфейсу, складових його компонентів. Існує чотири основних критерії якості будь-якого інтерфейсу, а саме:

1. Швидкість роботи користувачів.
2. Кількість людських помилок.
3. Швидкість навчання.
4. Суб'єктивне задоволення користувачів.

Швидкість виконання роботи є важливим критерієм ефективності інтерфейсу. У чистому вигляді цей критерій цінують досить рідко, але майже завжди він є вкрай бажаною складовою цілого. Будь-яка спроба якось збільшити продуктивність праці завжди зустрічається позитивно. Тривалість виконання роботи користувачем складається з наступних складових частин:

тривалості сприйняття вихідної інформації; тривалості інтелектуальної роботи (час осмислення користувачем, що він повинен зробити); тривалості фізичних дій користувача; тривалості реакції системи.

Тривалість сприйняття вихідної інформації особливих коментарів не потребує, користувач повинен уявити собі, яка інформація про виконувану задачу у нього існує, і в якому стані знаходяться кошти, за допомогою яких він буде вирішувати це завдання. Основний час тут піде на зчитування показань системи.

Тривалість інтелектуальної роботи оцінюється тим, що взаємодія користувача з системою (не тільки комп'ютерної) складається з семи кроків:

1. Формування мети дій.
2. Визначення загальної спрямованості дій.
3. Визначення конкретних дій.
4. Виконання дій.

5. Сприйняття нового стану системи.
6. Інтерпретація стану системи.
7. Оцінка результату.

З цього списку стає видно, що процес роздуми займає майже весь час, протягом якого користувач працює з комп'ютером, у всякому разі, шість з семи етапів повністю зайняті розумовою діяльністю. Відповідно, підвищення швидкості цих роздумів приводить до істотного поліпшення швидкості роботи. На жаль, істотно підвищити швидкість власне мислення користувачів неможливо. Проте, зменшити вплив чинників, які ускладнюють і, відповідно, що уповільнюють процес мислення, цілком можливо. Наприклад, застосуванням методу безпосереднього маніпулювання і застосування в інтерфейсі ефективних методів при втраті фокуса уваги. Втрата фокусу уваги призводить до втрати користувачів, необхідно проектувати інтерфейс так, щоб користувачі якомога менше про

нього думали. Для продовження роботи користувач повинен знати, на якому етапі він зупинився; які команди і параметри він вже дав системі; що саме він повинен зробити на поточному кроці; куди було звернуто його увагу на момент відволікання.

Будь-яка фізична дія, що здійснюється за допомогою мускулатури, може бути або точним, або швидким. Разом точність і швидкість зустрічаються виключно рідко, оскільки для цього потрібно виробити істотну ступінь автоматизму. Пояснюється це фізіологічними факторами: при різкому русі неможливо швидко зупинитися, відповідно, точне рух має бути плавним і уповільненим.

Таким чином, щоб фізична дія користувача було швидким, воно не повинно бути точним. Користувач, як правило, керує комп'ютером двома способами, а саме мишею і клавіатурою. Клавіатура не вимагає особливої точності рухів - неважливо, швидко натиснули кнопку або повільно, так само як сильно або слабо. Миша, навпаки, інерційна - є різниця між повільним її переміщенням і швидким,

сильним прикладеним зусиллям і слабким. Саме тому оптимізація використання миші в системі може істотно підвищити загальну швидкість робіт. Можливі помилки: помилка визначення дії; втрата мети; модальна помилка.

Очевидно, що призначений для користувача інтерфейс безпосередньо залежить від розв'язуваних програмним забезпеченням завдань, вхідних і вихідних даних; однак при цьому існують значна свобода в тому, в якому вигляді всі ці дані будуть представлені користувачеві. Від того, наскільки призначений для користувача інтерфейс буде функціональний, зрозумілий і зручний кінцевому користувачеві,

багато в чому залежить успішність вирішення поставленого при проектуванні програмного забезпечення.

На процес проектування призначеного для користувача інтерфейсу неабиякий вплив надають суб'єктивні уявлення проектувальника про зрозумілості, зручності і красу. Тому, велике значення має проблема оцінки якості призначеного для користувача інтерфейсу. Проводячи такі оцінки на ранніх етапах процесу проектування, можна уникнути великої кількості помилок, прорахунків, неприйняття ПО кінцевими користувачами. Хоча оцінка якості призначеного для користувача інтерфейсу процес досить суб'єктивний і важко формалізується, можна з упевненістю стверджувати, що хороший інтерфейс повинен забезпечувати ефективну і продуктивну роботу користувача. Існує також і ряд критеріїв, яким повинен задовольняти якісний інтерфейс: краще той інтерфейс, при якому час виконання завдання менше; краще той інтерфейс, в якому число мимовільних помилок користувача менше, неоднозначність в розумінні інтерфейсу повинна бути мінімальна (це сприяє самонавчання користувачів і робить їх поведінку передбачуваним); необхідна висока стандартизація інтерфейсу (вона полегшує навчання користувачів); обсяг введеної користувачем інформації повинен прагнути до мінімуму (одні і ті ж дані не повинні вводитися кілька разів), простота і візуальна привабливість (зручність використання не менш важливо, ніж функціональність).

Прототипування полягає в створенні широкого набору макетів (прототипів) майбутнього призначеного для користувача інтерфейсу, які піддаються порівняльний аналіз. Як правило, прототип містить реалізацію лише самого інтерфейсу, без його функціонального наповнення. Мета прототипування полягає у визначенні, наскільки те чи інше рішення перспективно, і подальшої реалізації одного з найкращих можливого. Цей підхід дозволяє заощадити час і ресурси, що витрачаються на проектування та розробку. Найбільш доцільно застосовувати цей підхід на ранніх етапах проектування, що допомагає вибрати правильний напрямок розробки, проте можливо і створення "локальних" прототипів для окремих елементів призначеного для користувача інтерфейсу. Таким чином, даний підхід охоплює як проектування інтерфейсу як цілого, так і проектування його частин. Для створення прототипів приваблюють не тільки фахівців, але і кінцевих користувачів, при цьому корисні будь-які думки, пропозиції та графічні начерки; основне завдання - створити 5 - 7 варіантів інтерфейсу, вирішального одну і ту ж задачу. При створенні прототипів потрібно виходити з розумного балансу між такими ключовими факторами: необхідний обсяг ресурсів для створення прототипу; плановане час життя прототипу (призначений він для вирішення короткострокової, локальної проблеми або для тривалого, глибокого аналізу); ризик зміни цілей проектування (зосередження уваги

нема на вирішенні проблеми за допомогою прототипів, а на створенні самих прототипів).

Створені прототипи піддаються порівняльній аналіз, в зв'язку з чим необхідно визначити критерії оцінки. Відправною точкою у визначенні таких критеріїв служить та проблема, задля вирішення якої були створені прототипи. Їй може бути час введення даних користувачем, час прийняття користувачем рішення на основі наданої інформації, суб'єктивна оцінка якості інтерфейсу за деякою

шкалою і т.п. Проектування інтерфейсів на папері в 21 столітті - найшвидший, дешевий, багатоплатформовий і загальнодоступний метод. Значна частина проектувальників не використовує паперові прототипи і скетчінга внаслідок недовіри. Більшість фахівців помилково вважають, що такий простий і швидкий шлях не допоможе отримати достатньо інформації для роботи над проектом. Життя поза монітора навчила нас думати, що по-справжньому ефективні інструменти повинні бути дуже дорогими і витрачати багато ресурсів. Для проектування інтерфейсів з'явилося безліч інструментів прототипування. З їх допомогою можна отримати хороший результат, але він не можна порівняти з часом, витраченим на опрацювання і тестування макетів. Буквально, три-чотири паперових скетчу головної сторінки цілком достатньо для того, щоб отримати інформацію про те, якою буде поведінка користувача і наскільки інтерфейс інтуїтивно зрозумілий. Паперові прототипи значно економлять бюджет.

Існує гіпотеза, довести яку може кожен, спробувавши в своїй роботі прототипування на папері. Суть гіпотези в наступному: "Чим раніше ми отримаємо інформацію про юзабіліті інтерфейсу, тим якісніше буде підсумковий результат". Юзабіліті розроблюваної системи підвищиться, якщо на самій ранній стадії будуть виявлені і виправлені проблеми в інтерфейсі. Всі знають, що тестування юзабіліті на кожному етапі проекту є обов'язковим, але чим раніше проблеми будуть виявлені, тим простіше їх можна вилікувати. На етапі скетчів і начерків все ще можна, а в деяких випадках навіть потрібно, робити фундаментальні зміни. Робити добре хочуть багато, але з різних причин не роблять. Цільовий користувач (Актор) являє собою будь-яку зовнішню по відношенню до модельованої системі сутність, яка взаємодіє з системою і використовує її функціональні можливості для досягнення певних цілей. При цьому актори служать для позначення узгодженого безлічі ролей, які можуть грати користувачі в процесі взаємодії з проектованою системою. Кожен актор може розглядатися як якась окрема роль щодо конкретного варіанту використання. Стандарним графічним позначенням актора на діаграмах є фігурка чоловічка, під якою записується ім'я актора. У деяких випадках актор може позначатися у вигляді прямокутника класу з ключовим словом "актор" і звичайними складовими



елементами класу. Імена акторів повинні записуватися великими літерами і слідувати рекомендаціям використання імен для типів і класів моделі.

Прикладами акторів можуть бути: клієнт банку, банківський службовець, продавець магазину, менеджер відділу продажів, пасажир авіарейсу, водій автомобіля, адміністратор готелю, стільниковий телефон і інші сутності, що мають відношення до концептуальної моделі відповідної предметної області. Так як в загальному випадку актор завжди знаходиться поза системою, його внутрішня структура ніяк не визначається. Для актора має значення тільки те, як він сприймається з боку системи.

Актори взаємодіють з системою за допомогою обміну повідомленнями з варіантами використання. Повідомлення являє собою запит актором певного сервісу системи і отримання цього сервісу. Ця взаємодія може бути виражена за допомогою асоціацій між окремими акторами і варіантами використання або класами. Крім цього, з акторами можуть бути пов'язані інтерфейси, які визначають, яким чином інші елементи моделі взаємодіють з цими акторами. Два і більш актора можуть мати загальні властивості, тобто взаємодіяти з одним і тим же безліччю варіантів використання однаковою чином. Така спільність властивостей і поведінки представляється у вигляді відношення узагальнення з іншим, можливо, абстрактним актором, який моделює відповідну спільність ролей. Структура інтерфейсу служить для специфікації параметрів моделі, які видимі ззовні, без вказівки їх внутрішньої структури. У мові UML інтерфейс є класифікатором і характеризує тільки обмежену частину поведінки модельованої сутності. Стосовно діаграм варіантів використання, інтерфейси визначають сукупність операцій, які забезпечують необхідний набір сервісів для акторів. Графічний символ окремого інтерфейсу з'єднується на діаграмі суцільною лінією або пунктирною лінією зі стрілкою з тим варіантом використання, який його підтримує. Суцільна лінія вказує, що пов'язаний з інтерфейсом варіант використання повинен реалізовувати всі необхідні для нього сервіси. Пунктирна лінія зі стрілкою означає, що варіант використання призначений для специфікації тільки того сервісу, який необхідний для реалізації даного інтерфейсу.

**Висновок до розділу:** проаналізовано найпопулярніші сучасні інструменти для прототипування та розробки інтерфейсу. Більшість з них є безкоштовними, а користувачі не виділяють з них явних фаворитів. Для розробки інтерфейсу державної установи "Київський міський центр громадського здоров'я" обрано засобом для прототипування та систематизації аналітичних даних інструмент Miro, так як він має

найбільший набір інструментів для вирішення поставлених завдань. Для реалізації фінального варіанту інтерфейсу на основі вайрфреймів вибрано ПЗ Figma, через його зрозумілість інтерфейсу, велику кількість плагінів які пришвидшують роботу, та за допомогою яких можливо прискорити розробку інтерфейсу.

Визначено та описано основні етапи розробки інтерфейсу державної установи. Визначено що найдоцільніше проектування інтерфейсу виконувати за таким алгоритмом:

- збір інформації;
- опрацювання матеріалів;
- створення lo-fi прототипів та user-flow
- тестування
- допрацювання lo-fi прототипів
- створення hi-fi прототипів
- тестування
- допрацювання hi-fi прототипів
- розробка дизайн системи
- проектування інтерфейсу
- тестування

Детально досліджено етап вайрфреймінгу, за допомогою якого можливо однозначно визначитись з основними напрямками вдосконалення досвіду взаємодії з інтерфейсом.

## РОЗДІЛ 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ ПРОЄКТУВАННЯ ІНТЕРФЕЙСУ "КИЇВСЬКИЙ МІСЬКИЙ ЦЕНТР ГРОМАДСЬКОГО ЗДОРОВ'Я"

### 3.1. Дослідження та аналітика потреб користувачів веб-ресурсу

Найважливішим етапом підготовчих робіт до початку проектування інтерфейсу є дослідження та аналітика потреб користувачів певного інтерфейсу, адже від нього залежить чи отримає користувач необхідну інформацію.

Спочатку проводилося вивчення вимог до розробки інтерфейсу державної установи, та були проаналізовані велика кількість подібних інтерфейсів. Спочатку вони були зібрані на одній сторінці для дослідження спільних паттернів та архітектурних принципів побудови досвіду використання.



Рис 3.1 Дослідження спільних паттернів

Також був отриманий доступ до якісних та кількісних метрик веб-ресурсів, та досліджено які саме паттерни найефективніші, та які рішення в проектуванні інтерфейсу державної установи є оптимальними для використання в продукті.

Наступним етапом йде інтерв'ю з замовниками розробки інтерфейсу державної установи "Київський міський центр громадського здоров'я", для досягнення взаєморозуміння у тому які саме з досліджених у попередньому етапі паттернів найкраще підійдуть для вирішення поставленої задачі. Також були проведені додаткові інтерв'ю з технічними спеціалістами предметної області, щоб поглибити свої знання для оптимального використання доступних ресурсів, та підвищити якість розробки продукту.

На основі зібраних даних була створена таблиця зі зведеними даними проведених інтерв'ю, та проаналізовані всі отримані данні, та визначено основні напрямки проектування інтерфейсу.

Country/Region	Link	Region	Target Audience	Platform	Advertising	Message	Feedback (Notes)
Країна/Регіон	Ссылка на веб-сайт/урок/курс	Географія (регіони)	Пользователи	Платформа	Реклама	Жаба	Обратная связь и замечания
Основной курс на Zilliox	<a href="https://www.zilliox.com/learn/online-course/learn-english">https://www.zilliox.com/learn/online-course/learn-english</a>	Во всем мире	Успешные студенты колледжа и студенты, а также все профессиональные образовательные и корпоративные организации	Видео и протипный курс, возможность получить сертификаты, онлайн и офлайн способы взаимодействия и получения курса данных	<a href="https://www.zilliox.com/">https://www.zilliox.com/</a>	Возможности	Обратная связь, в виде комментариев под курс и возможность для фидбека учителя и на основе полученных данных улучшить интерфейс
Основной курс на Udemy	<a href="https://www.udemy.com/courses/learn-english/">https://www.udemy.com/courses/learn-english/</a>	USA, India, UK, Turkey и другие страны по всему миру	Успешные и студенты, а также все профессиональные организации и корпоративные организации	Видео уроки, в которых рассказывается о филологических операциях и операциях перевода	<a href="https://www.udemy.com/">https://www.udemy.com/</a>	30 млн. по всему TWorld	
Обучение онлайн курсом на edX	<a href="https://www.edx.org/learn/online-course/learn-english">https://www.edx.org/learn/online-course/learn-english</a>	Обучение базе или возможность развиваться курсы для профессионалов или курсы	Обучение базе или возможность развиваться курсы для профессионалов или курсы	Видео уроки, в которых рассказывается о филологических операциях и операциях перевода	<a href="https://www.edx.org/">https://www.edx.org/</a>	Возможности, сертификаты EdX и др.	в курсе, в колонках и фидбек учителя и на основе полученных данных улучшить интерфейс
Основной курс на Coursera	<a href="https://www.coursera.org/learn/online-course/learn-english">https://www.coursera.org/learn/online-course/learn-english</a>	USA	Курс рассказывается на виде уроки, в которых рассказывается о филологических операциях и операциях перевода	Обучение на видеороликах онлайн уроки, в которых рассказывается о филологических операциях и операциях перевода	<a href="https://www.coursera.org/">https://www.coursera.org/</a>	Возможности	
Основной курс на Zilliox	<a href="https://www.zilliox.com/learn/online-course/learn-english">https://www.zilliox.com/learn/online-course/learn-english</a>	Во всем мире	Успешные студенты колледжа и студенты, а также все профессиональные образовательные и корпоративные организации	Видео и протипный курс, возможность получить сертификаты, онлайн и офлайн способы взаимодействия и получения курса данных	<a href="https://www.zilliox.com/">https://www.zilliox.com/</a>	Возможности	Обратная связь, в виде комментариев под курс и возможность для фидбека учителя и на основе полученных данных улучшить интерфейс

Рис 3.2 Зведена таблиця результатів інтерв'ю замовника та технічних спеціалістів

Наступним кроком стало користувацьке дослідження, в якому були виявлені шаблони, логіка та мотиви поведінки користувачів, поняття проблематики предметної області, та потреб користувача, що допоможе прийняти обгрунтоване рішення у питаннях проектування цифрового продукту. Була виділена цілова аудиторія, та зроблені гіпотези на основі попередніх даних, які будуть перевірятися під час користувацьких інтерв'ю.

Підготувавши гіпотези, та додаткові питання, та підібрані користувачі, які входять до цільової аудиторії, були проведені користувацькі інтерв'ю, на яких було зібрано великий масив даних.

Рис 3.3 Результати користувацьких інтерв'ю

Проаналізувавши данні, було визначено які гіпотези підтвердились, та за якими принципами та паттернами користувачам буде краще користуватись цифровим продуктом.

Після проходження всіх етапів дослідження, збору та аналізу даних було визначено, якою є цільова аудиторія продукту, були підтверджені гіпотези побудовані на основі користуцязьких досліджень методом інтерв'ю, проведено дослідження теми та предмету цифрового продукту, а також визначено за допомогою яких паттернів та принципів буде проектуватися користувацький інтерфейс.

### 3.2. Проектування wireframes на основі аналітичних даних

Перед початком роботи над вайрфремами, необхідно завершити декілька додаткових завдань, на сонові дослідження та отриманих аналітичних даних була створена мапа веб-ресурсу, з приблизною архітектурою необхідних сторінок майбутнього інтерфесу.

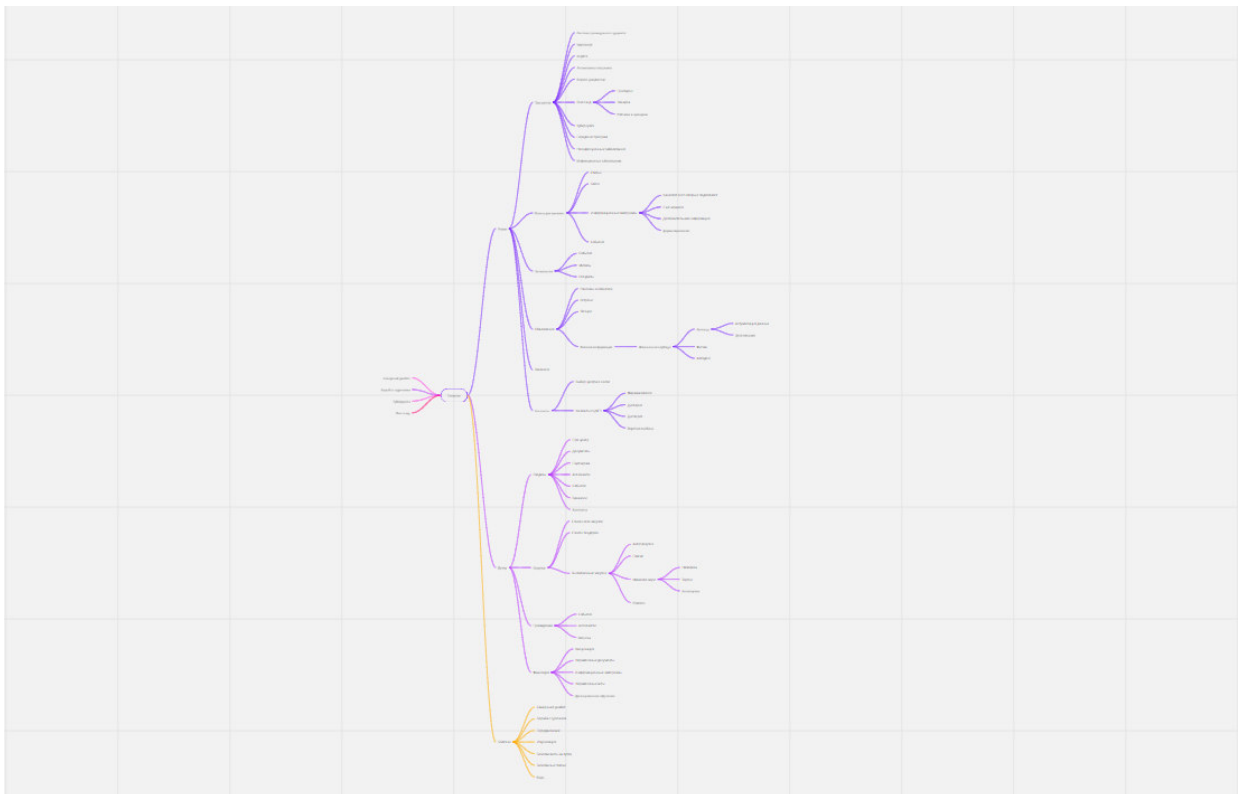


Рис 3.4 Мапа веб ресурсу

В мапі веб-ресурсу було відображено всі необхідні сторінки та за допомогою яким шляхом користувач буде потрапляты до необхідної інформації. На мапі було

описано як клієнтську частину веб-ресурсу, так і адміністративну панель, для наповнення інформацією веб ресурсу.

Головною задачею в проектуванні інтерфейсу реалізувати не тільки гарний дизайн, але створити функціональний продукт, яким буде зручно користуватися, та швидко орієнтуватися, тому наступним етапом стала розробка lo-fi вайрфреймів, та на їх основі проектування user-flow.

При створенні lo-fi вайрфреймів використовуються лише прості геометричні фігури, для визначення приблизного росташування елементів на екрані, щоб визначитися як саме досвід використання продукту буде найефективнішим та найпростішим. Візуалізація вайрфреймів йде згідно мапи сайту, та в першу чергу створюються основні сторінки, які необхідні для мінімально працюючого прототипу, по якому будуть проводити тестування.



Рис 3.5 Lo-fi вайрфрейми головних сторінок

Створивши всі вайрфрейми які необхідні для реалізації продукту, необхідно реалізувати user-flow, щоб визначити яким шляхом найоптимальніше реалізувати навігацію між сторінками, а також каталогізацію навігації. Адже система є складною,

та багаторівневою, необхідність створення такої діаграми є головним критерієм успішності реалізації та подальшого використання інтерфейсу.

User flow - це наочні матеріали, які ілюструють весь шлях користувача в продукті цілком. Це гібрид класичної блок-схеми і елементів візуального інтерфейсу. Основою user flow є завдання, які виконує користувач, а також можливі альтернативні ситуації які можуть виникнути, та методи їх вирішення. Тому при створенні такої схеми було взято до уваги всі аналітичні дані які були зібрані на попередньому етапі, та використані як база для її реалізації. Кожен потік у цій діаграмі робиться окремою діаграмою для кращого розуміння, та візуалізації інформації.

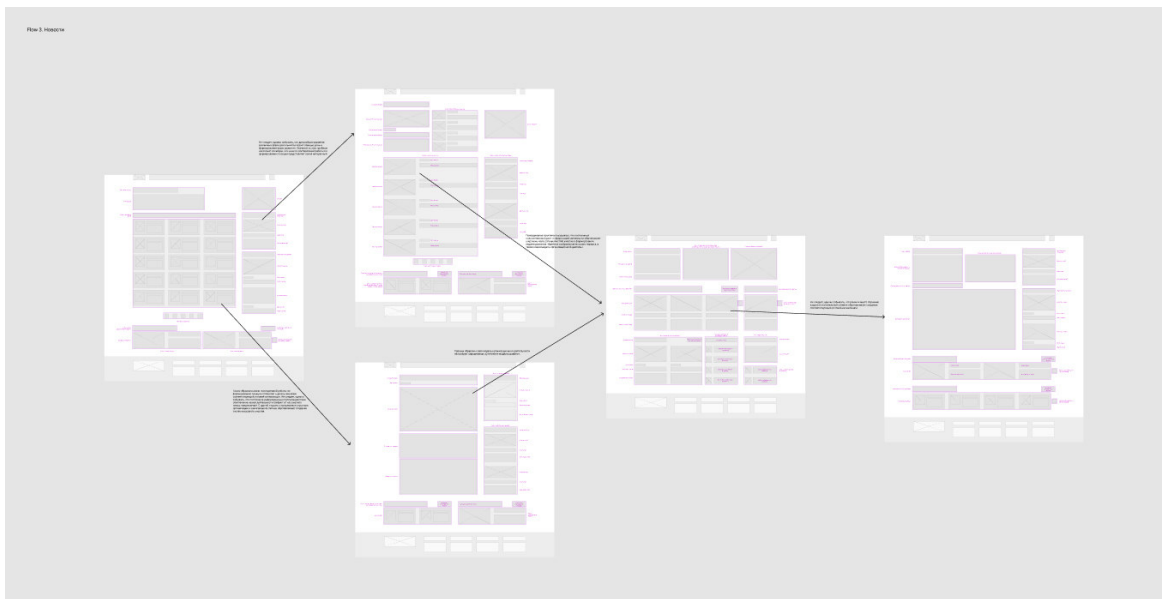


Рис 3.6 Вигляд одного етапу user-flow

Після завершення етапу прототипування lo-fi вайрфреймів, та завершення побудови user-flow, було проведено тестування створених матеріалів, та внесено корективи для покращення досвіду використання продукту. Повівши всі етапи тестування з цільовою аудиторією, та технічними спеціалістами предметної області була створена таблиця з усіма результатами, по яким вносились зміни.

Наступним етапом було створення hi-fi вайрфреймів, які є макетом дизайну інтерфейсу, які відображають зовнішній вигляд продукту на останніх стадіях процесу проектування.

Hi-fi вайрфрейми включають в себе фактичний зміст, шрифти, розміри зображення та елементи брендингу, які будуть використовуватись у фінальному варіанті інтерфейсу продукту.

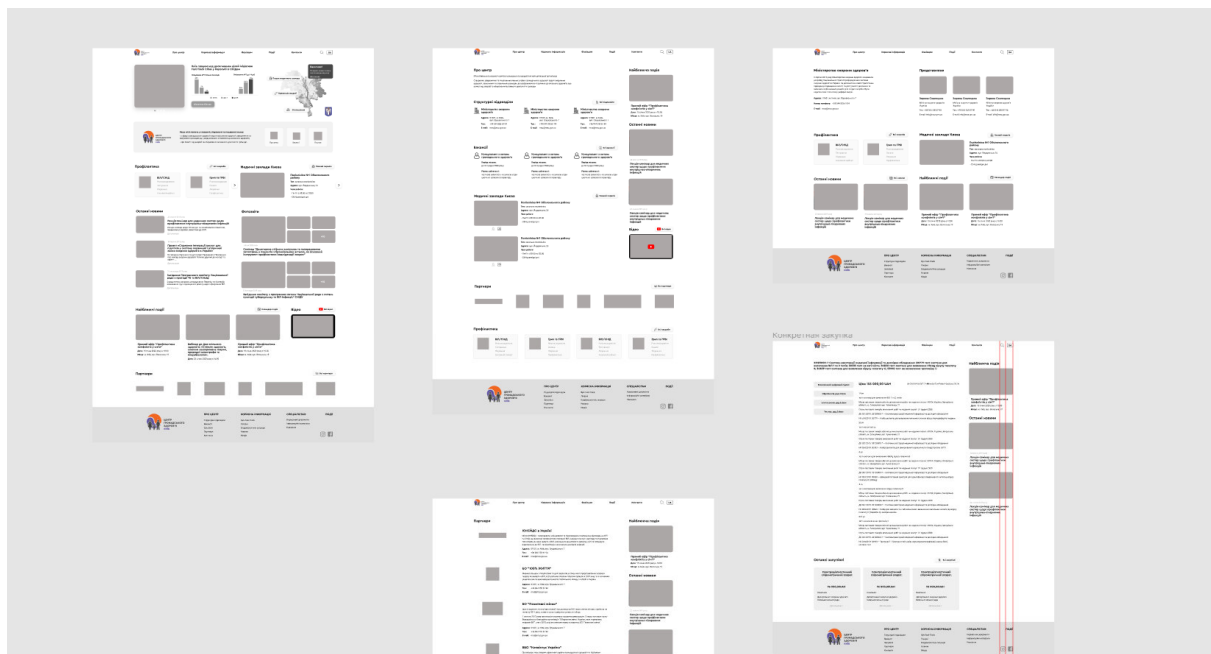


Рис 3.7 Hi-fi вайрфрейми

Реалізувавши всі вайрфрейми, необхідні для реалізації продукту було створено інтерактивний прототип, за допомогою якого проводилося тестування цільової аудиторії, для визначення кількісних та якісних показників метрик, по яким визначається успішність проходження шляху користувача по інтерфейсу, та оптимальності користувацького досвіду за використання інтерфейсу. Провівши дослідження була зібрана інформація, яка була проаналізована, та внесено додаткові корективи до прототипів. Після чого тестування повторювалося поки не було досягнуто успіху у всіх необхідних пунктах. Та почато підготовку до реалізації дизайну інтерфейсу на основі готових вайрфреймів.

### 3.3. Розробка дизайну інтерфейсу на основі wireframes

Найпершим етапом для створення дизайну інтерфейсу є підготовка дошок настрою, які використовуються для затвердження графічного стилю інтерфейсу, з замовником. Дошка настрою збиралась на платформі Behance.net, де були відібрані найоптимальніші інтерфейси, для демонстрації замовнику. Дошка настрою дає



можливість доступно та лаконічно продемонструвати замовнику, яке враження буде складатися від продукту за допомогою дизайну.

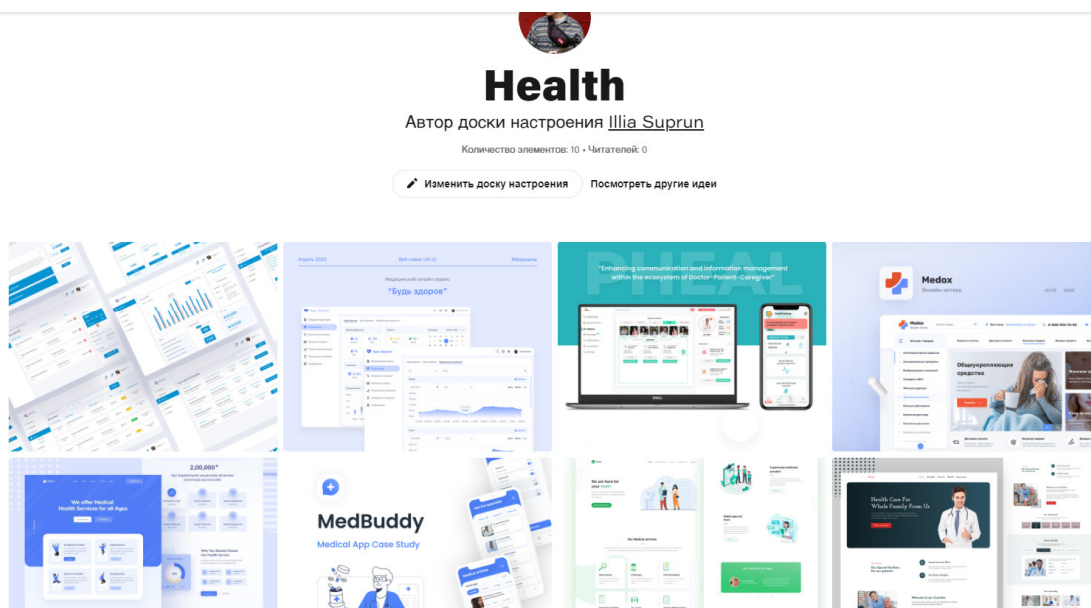


Рис 3.8 Дошка настрою

Обравши концепцію яка найяскравіше зображує ідеологію державної установи "Київський міський центр здоров'я", та обравши елементи які допоможуть донести цю ідеологію до користувачів почалася робота зі створення дизайну інтерфейсу.

Далі необхідно створити дизайн систему, яка дозволить дотримуватись однорідності інтерфейсу, зробіть досвід від використання продукту постійним та послідовним, а також спростить прийняття рішень з точки зору візуального дизайну інтерфейсу.

Були обрані шрифти, які будуть використовуватись в інтерфейсі, також прописано де вони будуть використовуватись, розмір кегелю, міжбуквенний та міжстроковий інтервали. Прописані відступи, які використовуються в ситсемі, та описано в яких ситуаціях який відступ буде використовуватись. Та додатково були прописані ситемні записи.

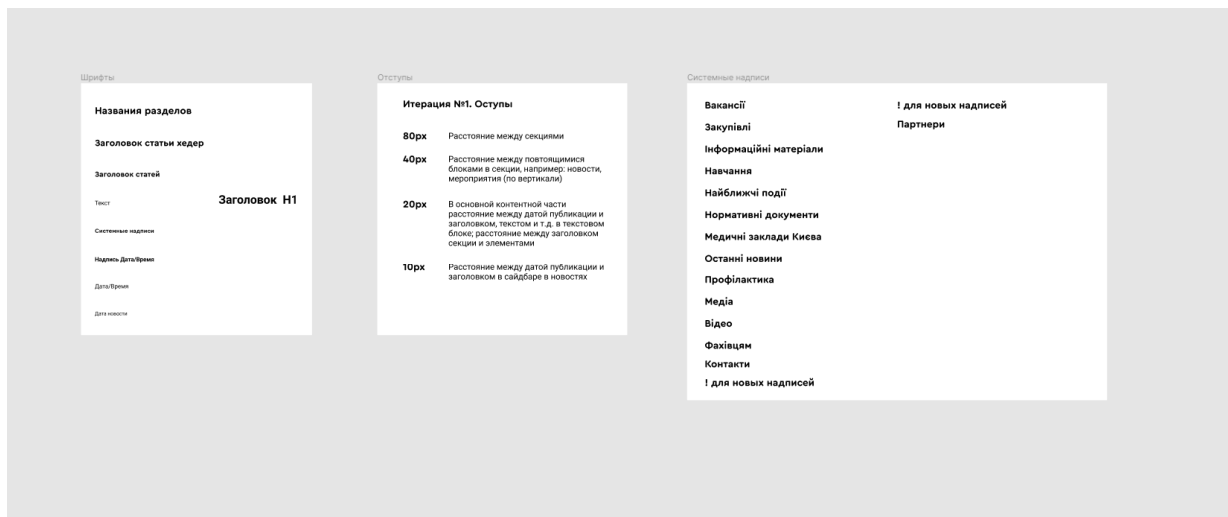


Рис 3.9 Шрифти, відступи та системні записи

Наступним кроком були додані системні кольори згідно брендбуку замовника, та перевірена їх доступність для людей з вадами зору, і додано контрастніші кольори для таких людей. І хоча більша частина інтерфейсу буде чорно-білою, було додано кольорові акценти для додаткової інформації, яка буде висвітлена в інтерфейсі.

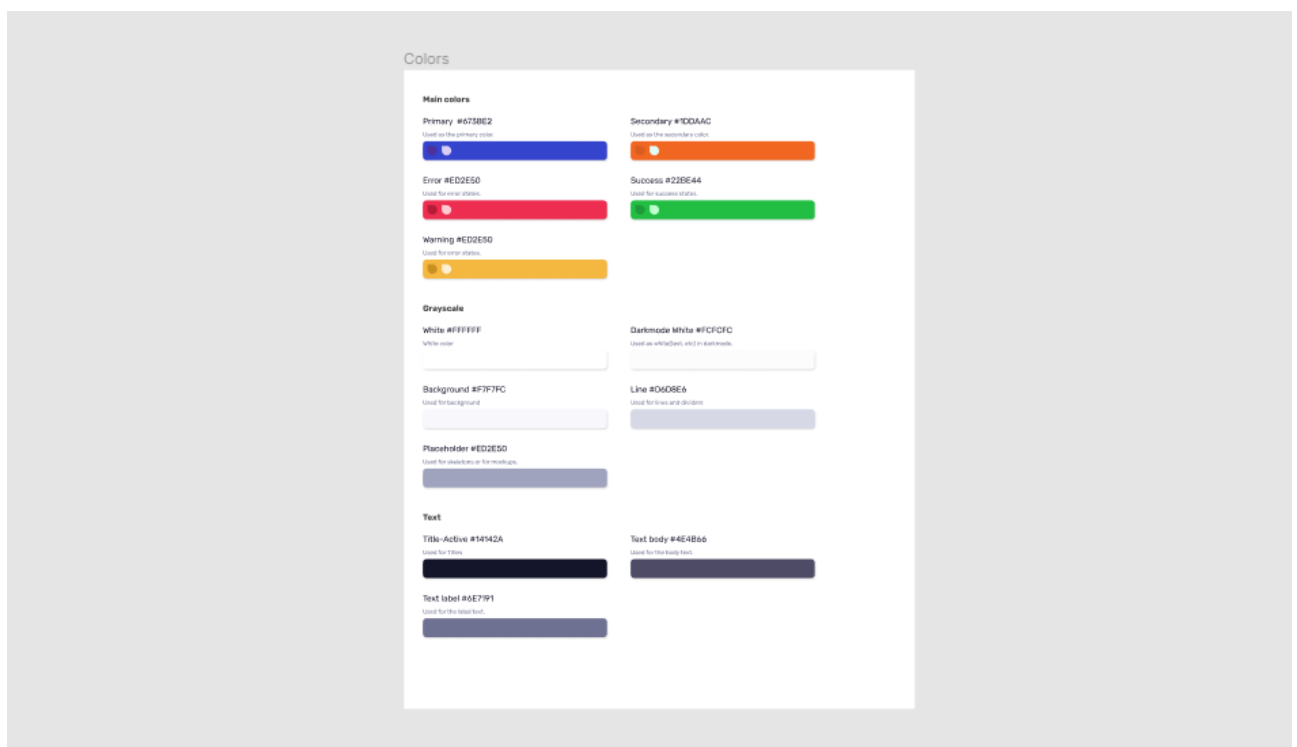


Рис 3.10 Кольорова схема

Визначившись з кольоровою схемою, та типографікою, наступним етапом було створення біль складних елементів для автоматизації створення інтерфейсу. Серед таких елементів можна виділити такі як хедери, футери, медіа, та медіаблоки, окремо

новини та блоки новин, пошук, документи, блоки хворіб, інформаційні матеріали та контакти клінік. Всі ці блоки часто використовуються на сторінках інтерфейсу, і для пришвидшення роботи з проектування дизайну, їх було додатково винесено у дизайн систему.

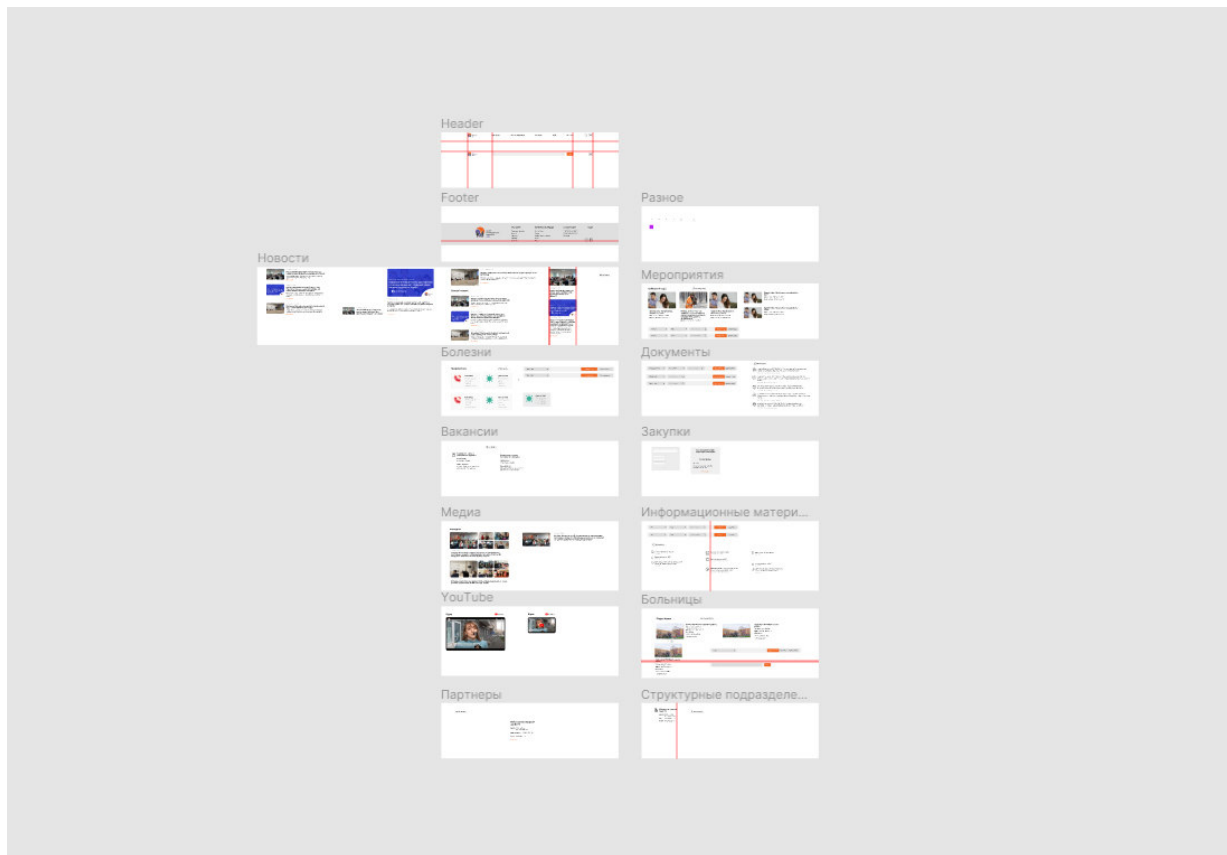


Рис 3.11 Додаткові елементи в дизайн системі

Завершивши з дизайн системою починається робота над проектуванням інтерфейсу, першою сторінкою взятою до роботи була головна сторінка, на якій відображається зібрана інформація з більшості сторінок доступних користувачеві, та доступне швидке переміщення на кожну з них. Основною інформацією яка є на головній сторінці є новини по Fast-track, та важлива інформація по місту Києву. Також відображена інформація по профілактиці хвороб, медичним закладам Києва, останні новини, фотозвіти з фотогалеерею і найближчі події. Вся ця інформація розміщується на сторінці згідно з вайрфреймами, та результатами досліджень, проведених раніше.

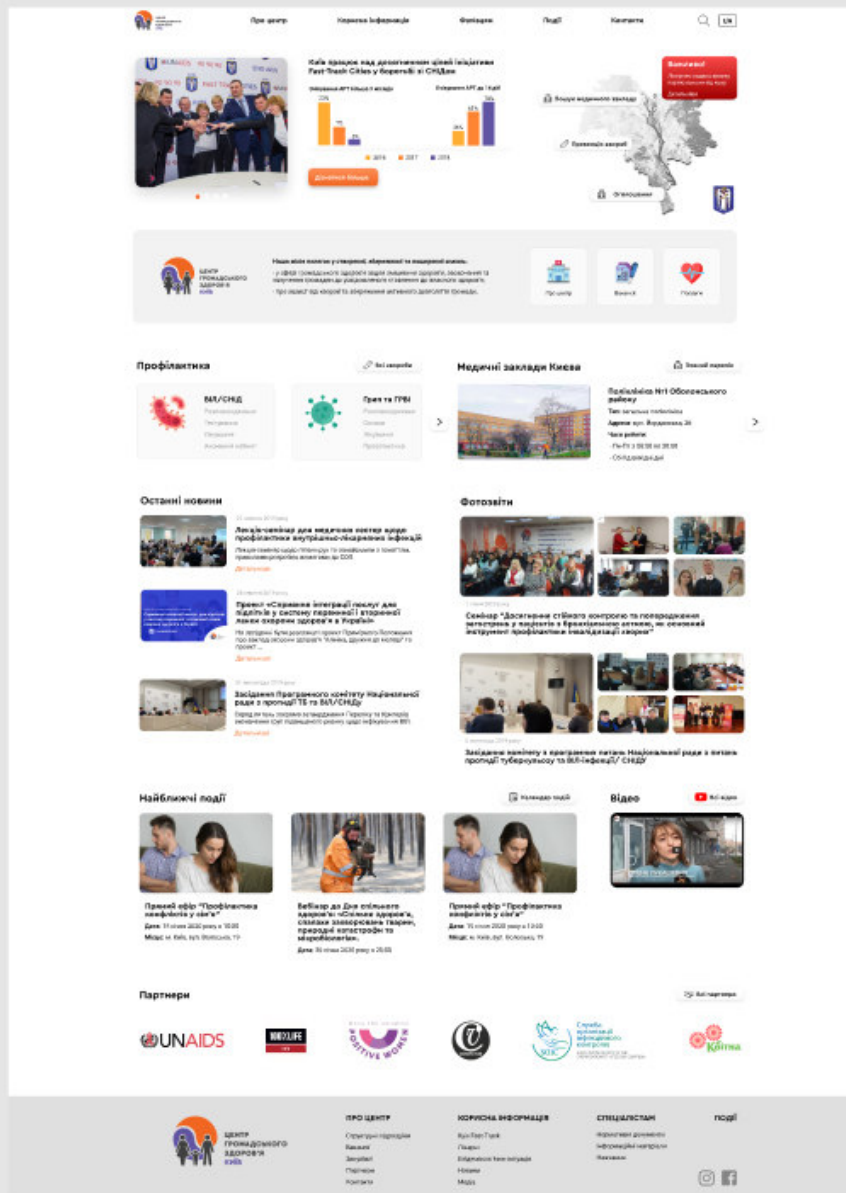


Рис 3.12 Головна сторінка

Для ініціативи Fast-track була створена окрема сторінка, на якій користувач може дізнатись інформацію по успішності імплементації ініціативи до системи охорони здоров'я, на ній приведені графіки успішності імплементації, та публікації по данній тематиці. Спочатку була розміщена найважливіша інформація для користувача, нажче додаткова інформація, для поглибленого вивчення прогресу по імплементації ініціативи.

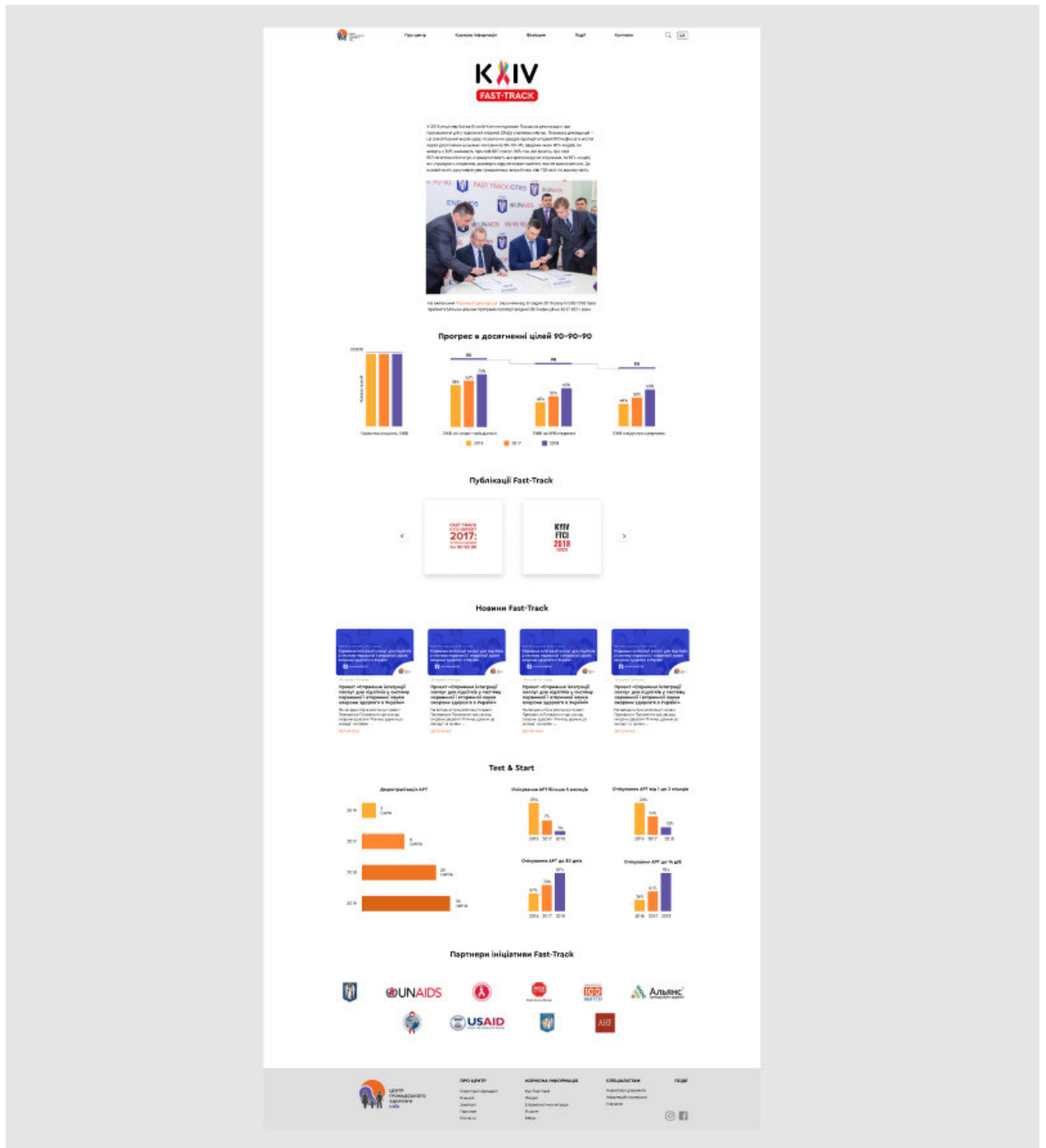


Рис 3.13 Сторінка ініціативи Fast-track

Наступним кроком було створення сторінки зв'язаної з спецпроект з приводу зупинки розповсюдження Covid-19, основною задачею цієї сторінки було оперативне донесення інформації до населення, та поширення фактів як убезпечити себе від цієї хвороби. Проаналізувавши надану інформацію, було розроблену спеціальну сторінку під цей спецпроект, та виділену основну інформацію для цільової аудиторії. Сторінка виконана в спільній стилістиці, згідно з дизайн системою, та

використовує спільні елементи інтерфейсу, для дотримання ідеї нерозривності дизайну інтерфейсу.

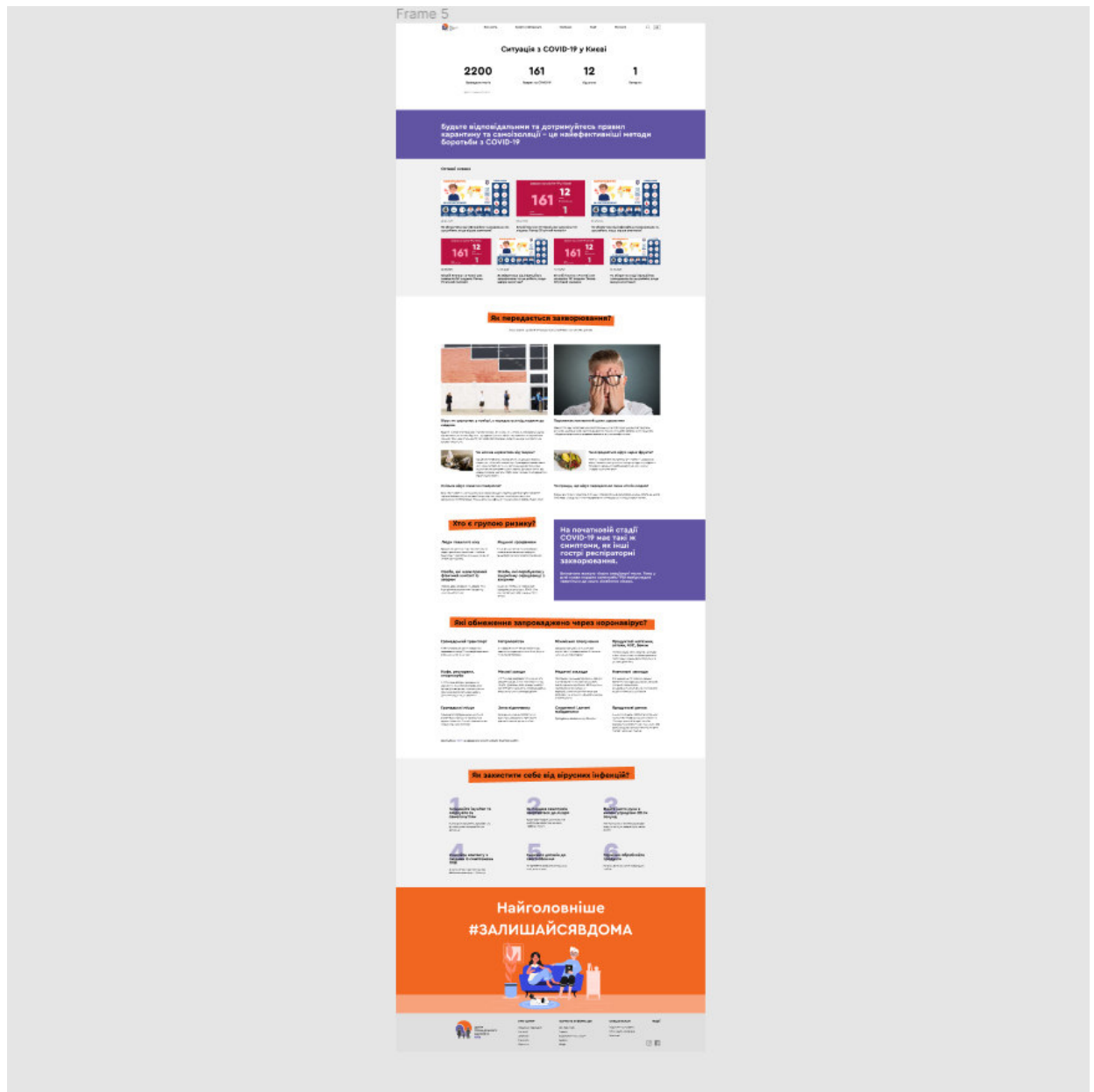


Рис 3.15 Сторінка присвячена COVID-19

Завершивши роботу над сторінкою наступним етапом роботи було створення інтерфейсу сторінки яка надає користувачам інформацію про "Київський міський центр громадського здоров'я". На ній згідно вайрфреймів повинні бути інформація про керівництво центру, цінності, місія, доступні вакансії та контакти державної установи. А також додаткова інформація про найближчі події, і останні новини, що допоможе користувачеві швидко та оптимально переміщуватись по веб ресурсу по веб-ресурсу, згідно мапи ресурсу.

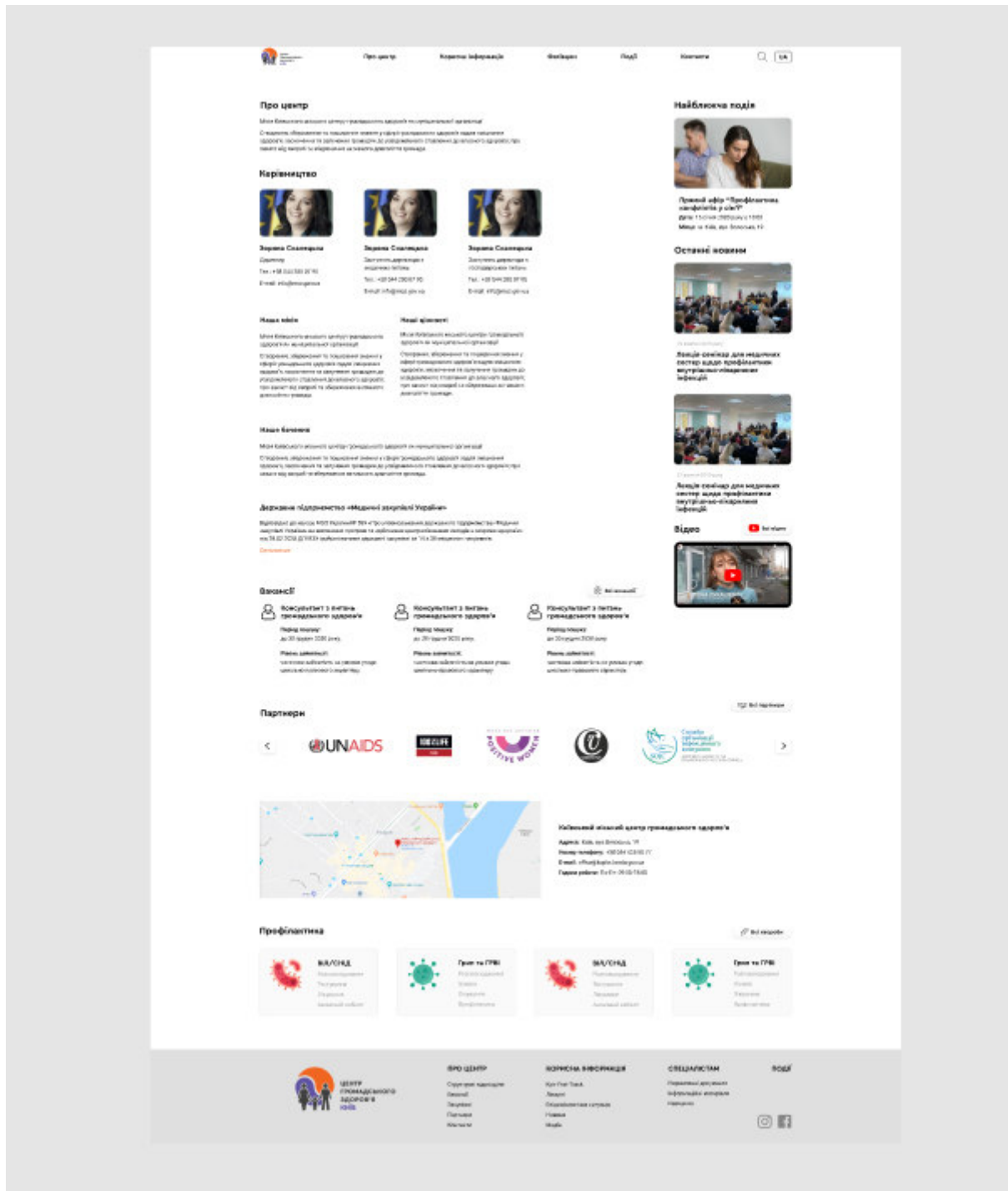


Рис 3.15 Сторінка інформації про державну установу

Сторінка присвячена профілактиці захворювань була підготовлена за допомогою створеної дизайн системи. На ній відображена коротка інформація про деталі контент на сторінці, та посилання на всі хвороби які відстежуються державною установою. Також додано фільтр сортування та відображення інформації по заданим параметрам.

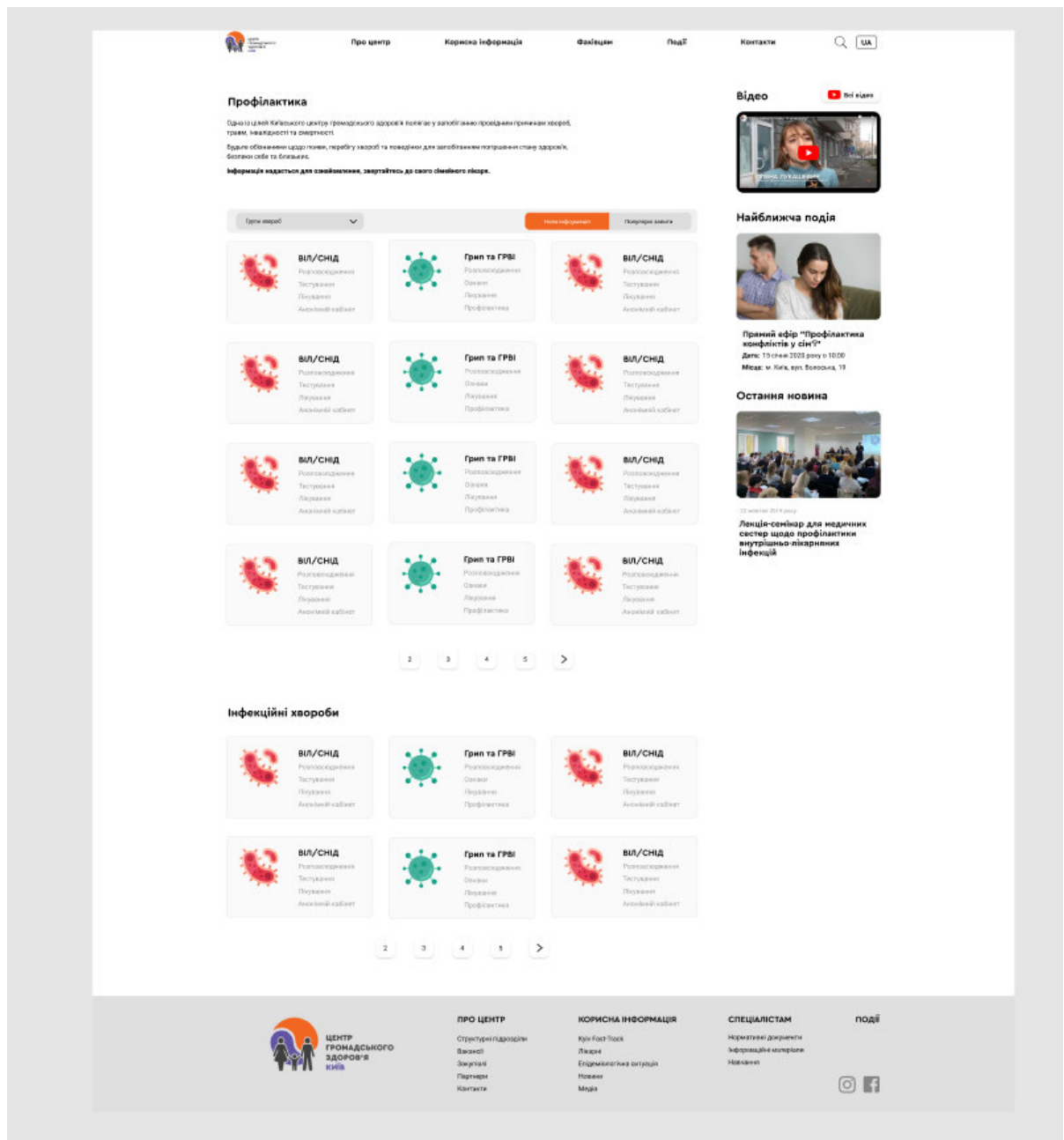


Рис 3.16 Профілактика захворювань

Зі сторінки профілактики можливо перейти до кожного захворювання, та дізнатися детальну інформацію про певну хворобу, а також вивчити епідеміологічну ситуацію, та дізнатися останні новини по темі. Додатково виділені нормативні документи пов'язані з хворобою, та найближчі події.

Додатково була створена сторінка більш детальної інформації по захворюванню, та розширена інформація з нормативними актами та документами. А також додані посилання на інші захворювання для кращої навігації по веб ресурсу.



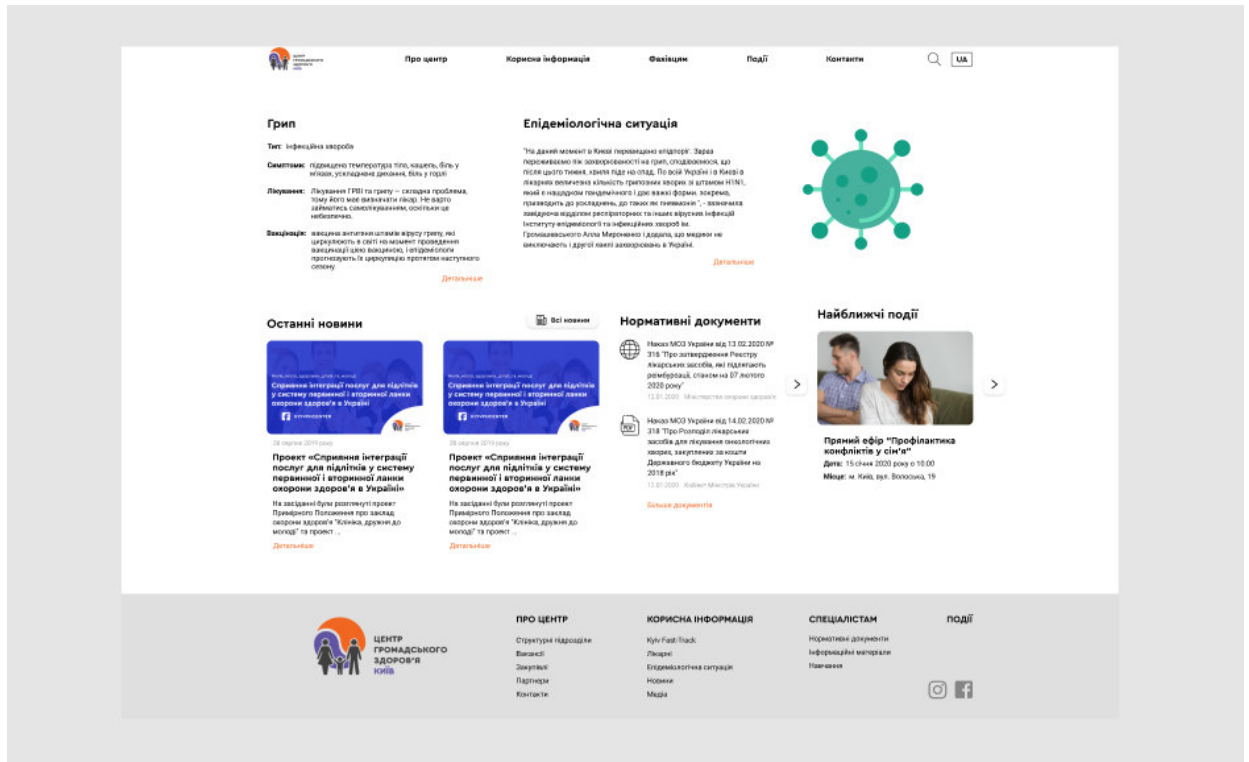


Рис 3.17 Коротка інформація про захворювання та епідеміологічна ситуація

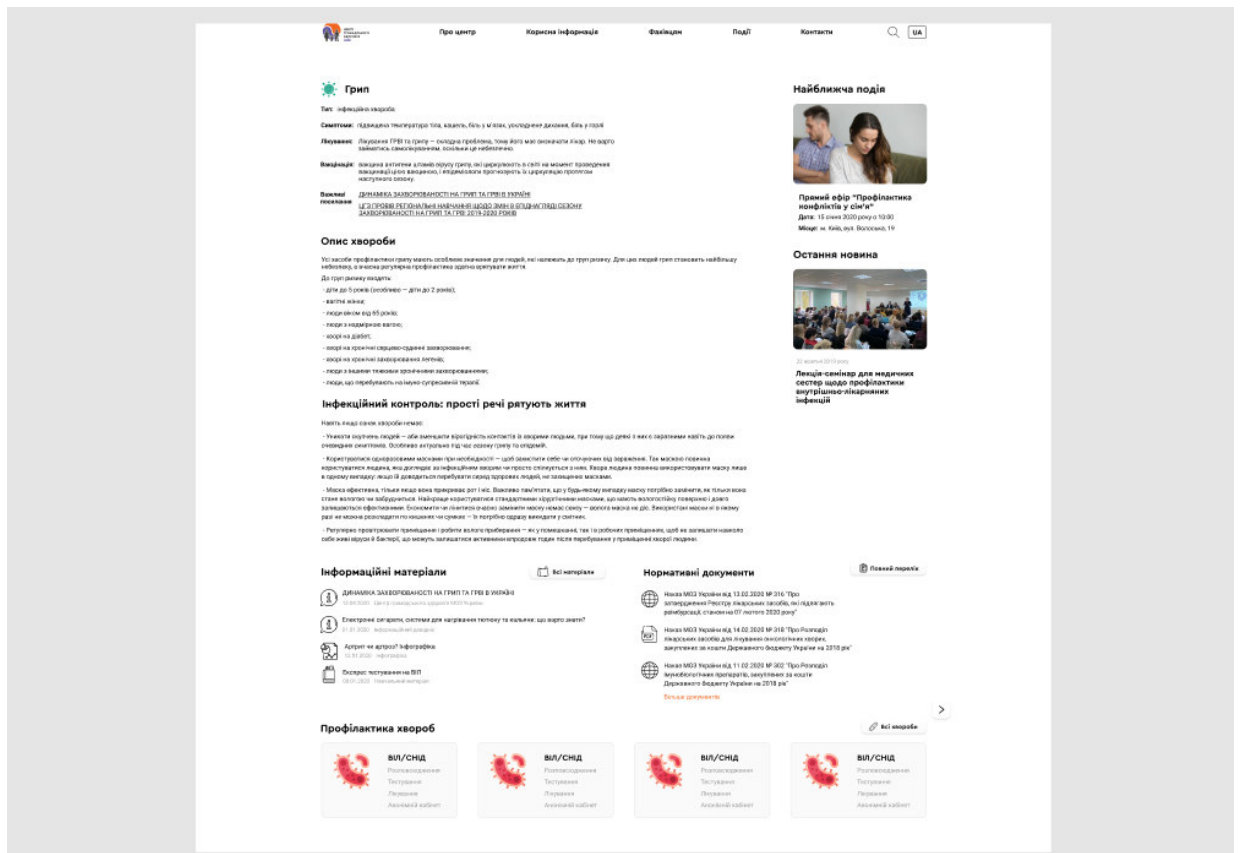


Рис 3.18 Сторінка інформації про захворювання

Важливою частиною інтерфейсу веб-ресурсу державної установи є частина з останніми новинами, та інформацією пов'язаною з "Київський міський центр громадського здоров'я". Було вирішено виводити на екран десять останніх новин, а також інформацію про найближчі події та навігацію по профілактиці захворювань. Для перегляду інших новин була додана пагінація, до дозволяє переміщуватись по блоку новин, та підвищує досвід використання інтерфейсу для користувачів.

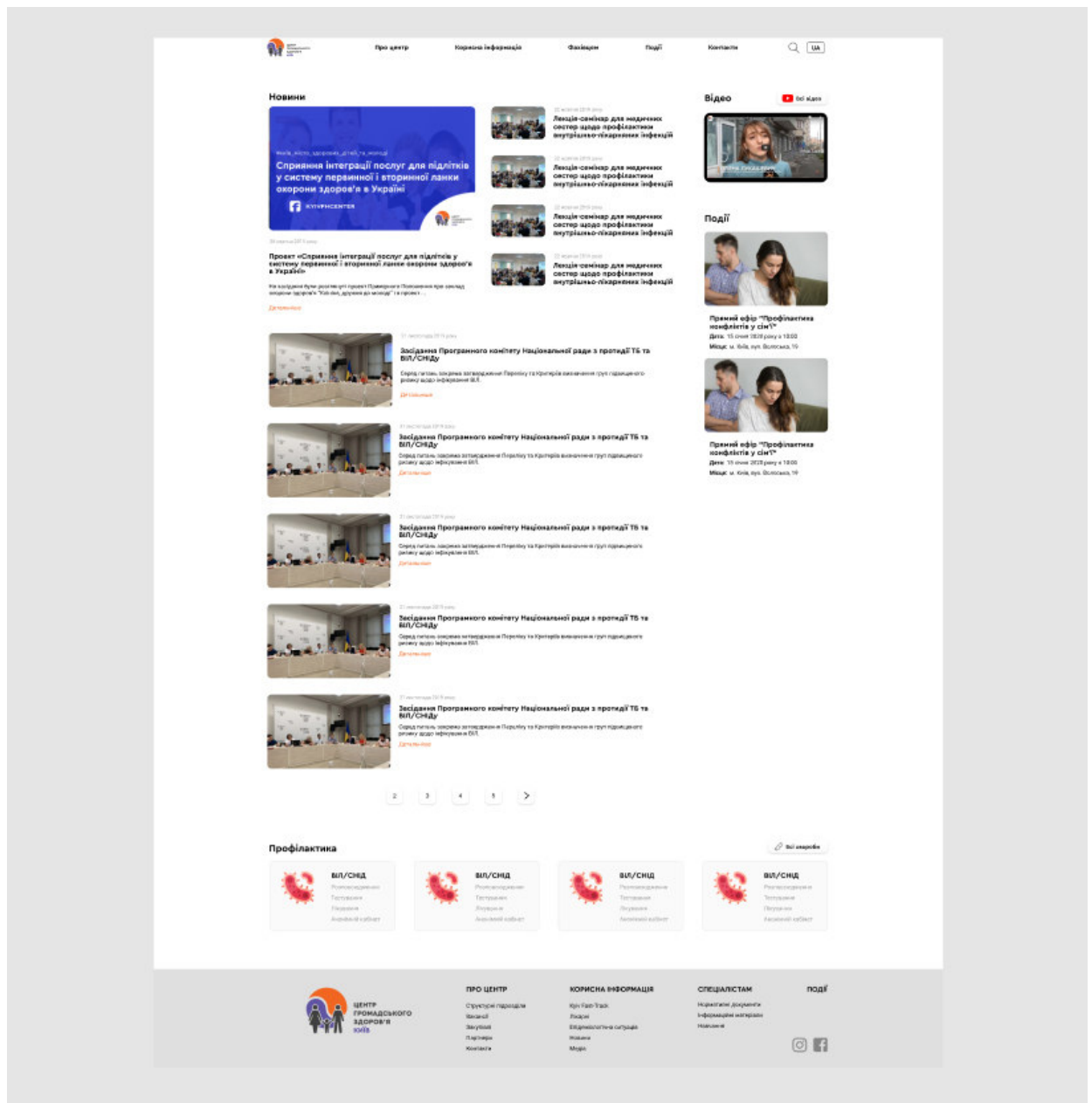


Рис 3.19 Сторінка усіх новин

Перейшовши до сторінки певної новини на ній відображається обкладинка новини у вигляді фотографії, та текст новини. Додатково за потреби додається можливість прикріплення інформаційних матеріалів, які користувач має можливість

завантажувати на свій пристрій, та детальніше вивчати. Задля поліпшення навігації по веб-ресурсу додано також інформаційні блоки про останні новини та найближчу подію, а також інформація про профілактику захворювань.

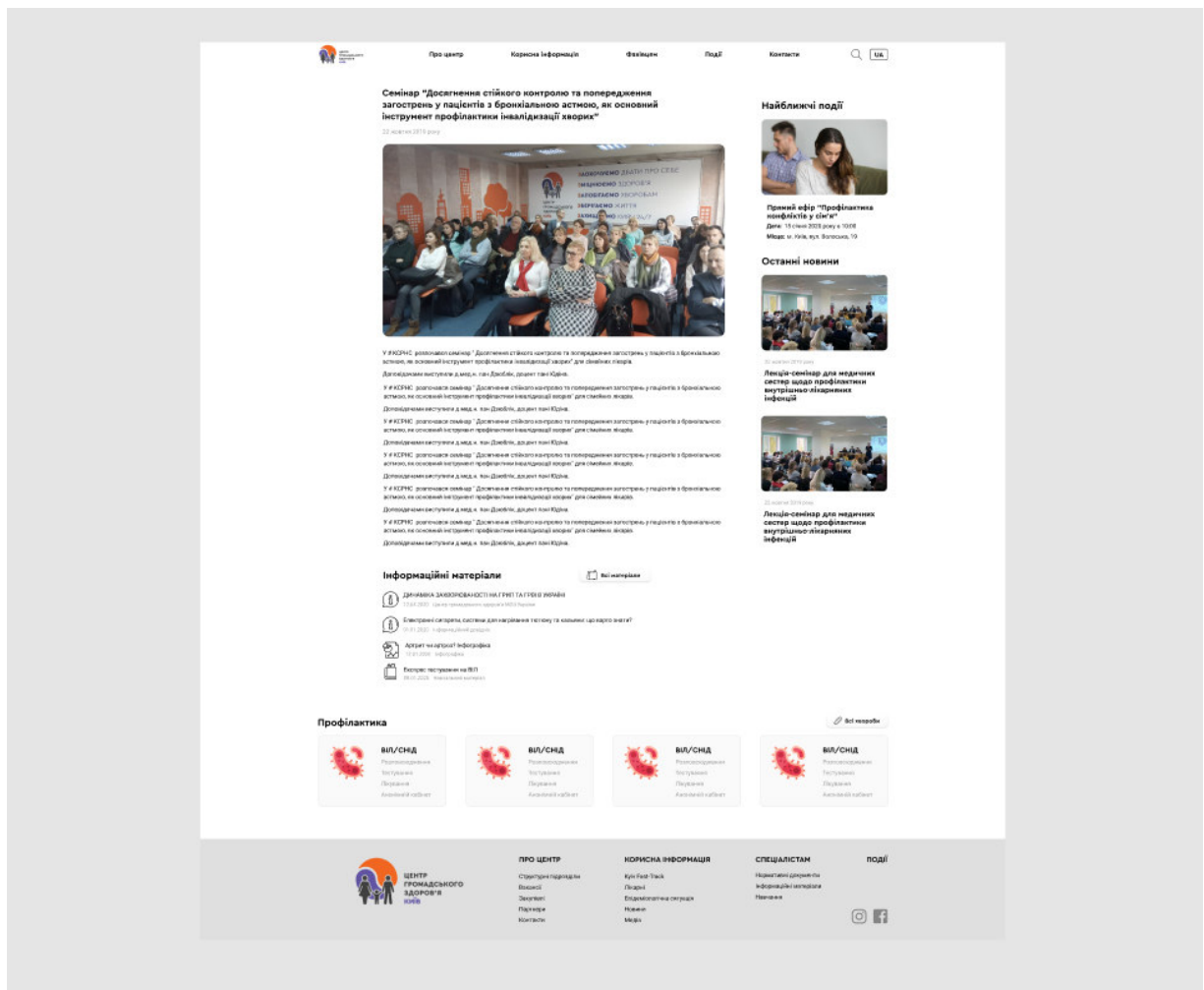


Рис 3.20 Новина

Одним з найважливіших елементів інтерфейсу є сторінка присвячена для фахівців, які за допомогою веб ресурсу зможуть швидко отримувати необхідну інформацію. Згідно вайрфреймів та аналітичних даних був розроблений інтерфейс сторінки з такими елементами як нормативні документи, інформаційні матеріали для фахівців, останні новини, навчальні матеріали та профілактика захворювань.

При переході на сторінку нормативних документів є можливість провести пошук щоб знайти необхідну інформацію, та певні нормативні акти, чи вивчити всі доступні документи на відкритій сторінці. Окремо на сторінці виділені документи Всесвітньої організації охорони здоров'я, та нормативні документи Кабінету міністрів України, для підвищення UX сторінки, за результатами тестувань вайрфреймів.

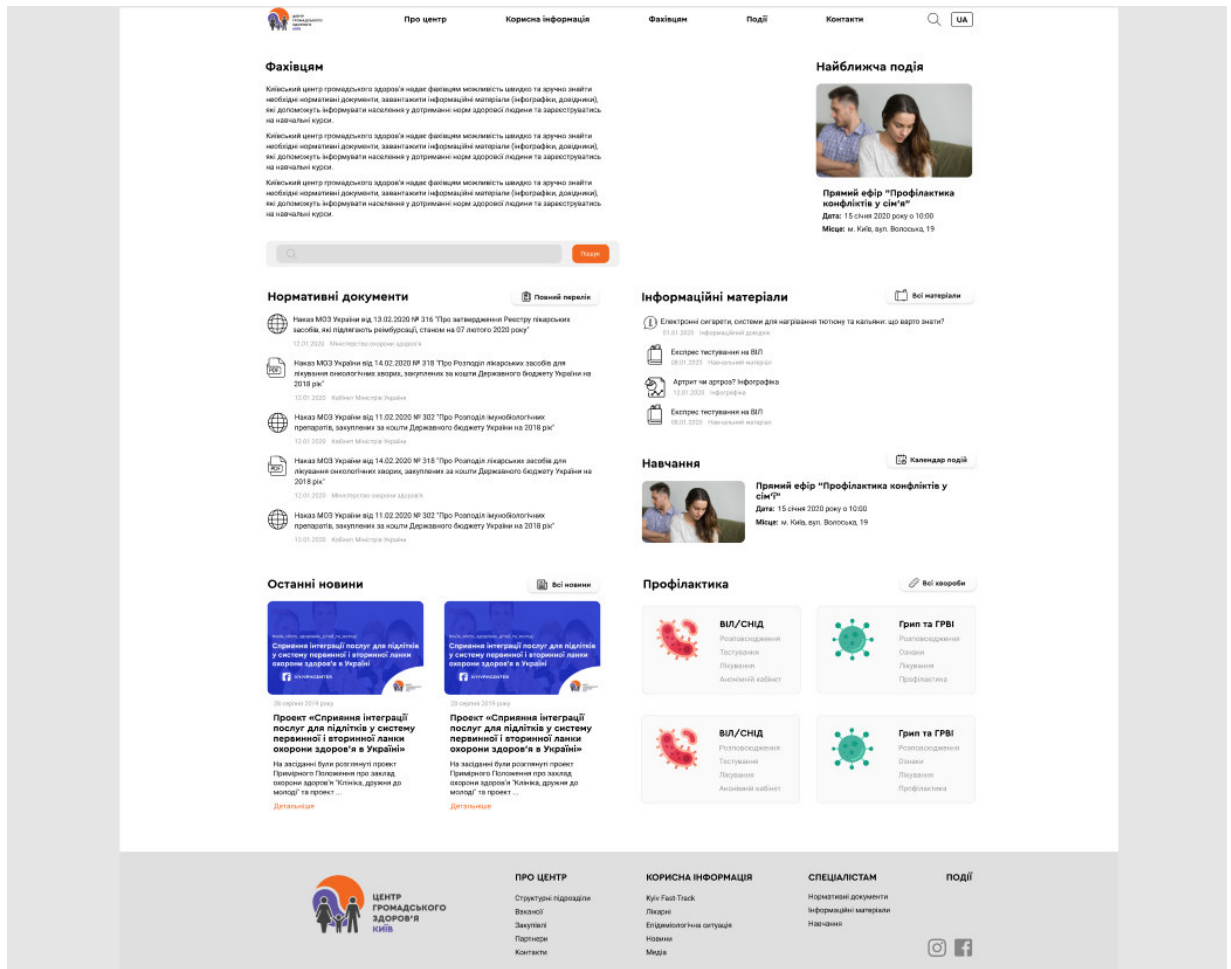


Рис 3.21 Сторінка для фахівців

Було розроблено зручний інтерфейс пошуку по нормативним документам для фахівців.

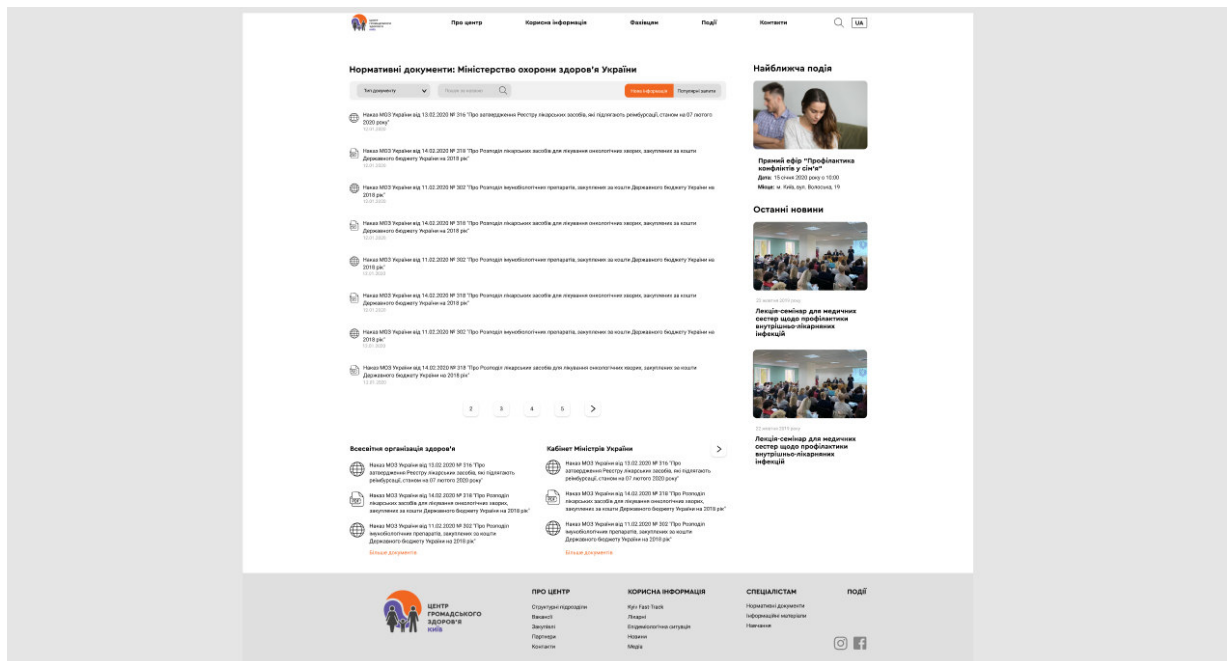


Рис 3.22 Результати пошуку нормативних актів та документів

Задля оптимізації та пришвидшення орієнтації користувача по інтерфейсу, було розроблено систему навігації за допомогою фільтрів та каталогів.

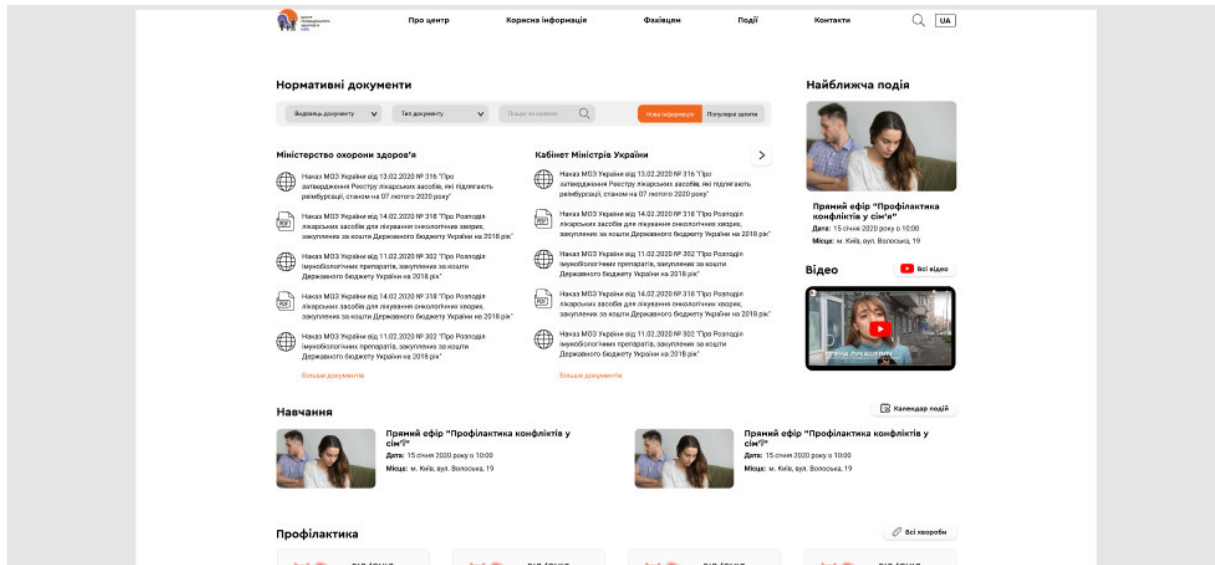


Рис 3.23 Сторінка нормативних документів

Відкривши нормативний документ, користувач може його переглянути онлайн, для більш детального ознайомлення, дізнатися хто видавець документу, та документи пов'язані з цією темою від інших організацій охорони здоров'я, та державних установ.

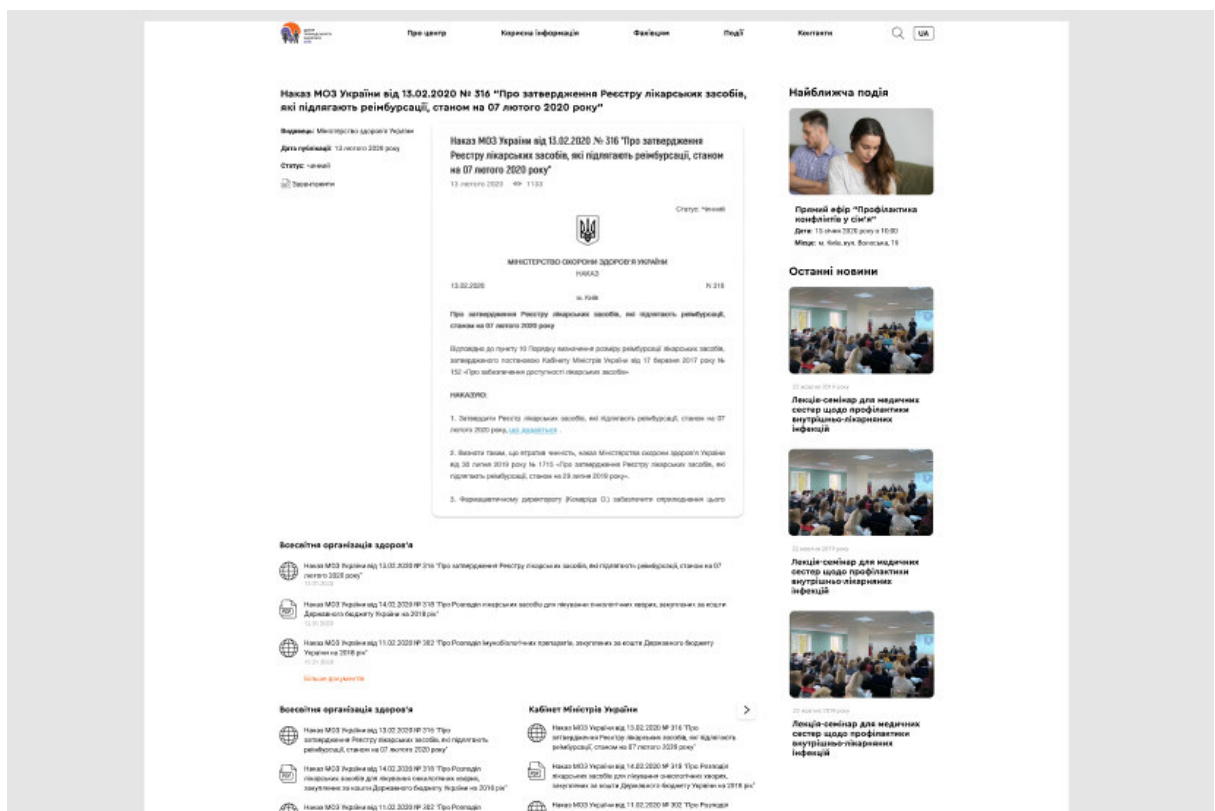


Рис 3.23 Сторінка нормативного документу

Закінчивши роботу над попередньою сторінкою, був покладений початок розробки дизайну інтерфейсу на основі вайрфремів та аналітичних даних, зібраних на попередньому етапі дослідження, додаткових сторінок інтерфейсу. Були створені інтерфейс сторінок новин, статей, подій, контактів та інших. Реалізовано інтерактивний прототип та проведено тестування інтерфейсу на цільовій аудиторії.



Рис 3.24 Всі сторінки

**Висновок до розділу:** в розділі було досліджено та проаналізовано потреби користувачів веб-ресурсів. Визначено вимоги до інтерфейсу державної установи, зібрано аналітичні дані та проведено дослідження потреб користувачів. Створено вайрфрейми та user-flow, проведено опитування та тестування на кожному етапі створення інтерфейсу, і спроектовано інтерфейс державної установи.

Ціллю проектування інтерфейсу державної установи є створення інформаційно-комунікаційного простору для користувачів та фахівців сфери охорони здоров'я. Беручи до уваги цілі продукту було визначено функції інтерфейсу, та розроблено його архітектуру.

Реалізовано інтерфейс державної установи "Київський міський центр громадського здоров'я", який складається з інформаційної складової (інформаційна структура, аналітичні дані та прототип) та концептуальної складової(стиль: модульна сітка, колірна гамма, типографіка, дизайн система; макет: дизайн інтерфейсу, функціональних блоків, піктограм, іконок).

Обрано каскадно-циклічний підхід до роботи(від загального до конкретного з тестуванням на кожному етапі) для проектування інформаційної та концептуальної складової структури інтерфейсу державної установи "Київський міський центр громадського здоров'я". Вайрфрейми інтерфейсу створені за допомогою додаткового програмного забезпечення.

Для проектування інтерфейсу обрано мінімалістичний стиль з елементами Apple гайдлайнів. Інтерфейс створений за допомогою модульної сітки яка складається з 12 колонок. Колірна схема виконана у монохромному стилі з використанням акцентів, згідно брендбуку державної установи. Створено інтерфейс державної установи, іконки та інтерактивний прототип.

## ВИСНОВКИ

У ході дипломної роботи досягнуто поставлено мету, задачі розв'язано. Отримано наступні результати.

1. Проведено аналіз та визначено класифікацію та характеристики інтерфейсу веб-ресурсів. За типами веб-ресурси поділяються за доступністю користувачам, за розміром та якістю аудиторії, за розміром, за метою створення та за функціональністю тощо. За видами веб-ресурси бувають інформаційними, загальнотематичними, новинними, блогами, корпоративними сайтами, іміджевими, промо-сайтами, форумами, тощо. Визначено ключові вимоги до інтерфейсу державних установ, та детально проаналізовано потреби до реалізації.

2. Описано основні принципи проектування інтерфейсу та прототипування на різних рівнях. Визначено етапи проектування інтерфейсу. Досліджено основні характеристики та принципи побудови user flow та user story. Детально досліджено етап вайрфреміну, який є одним з найважливіших кроків у досягненні та вдосконаленні досвіду взаємодії з інтерфейсом для користувачів. Визначено головні етапи технології проектування інтерфейсу.

3. Вибране та обгрунтоване програмне забезпечення для проектування інтерфейсу державної установи "Київський міський центр громадського здоров'я". Для проектування інтерфейсу вибрано онлайн сервіс Miro, функціонал якого інтуїтивно зрозумілий та містить в собі всі необхідні інструменти. Для створення макетів інтерфейсу державної установи обрано програму Figma, яка має широкий функціонал та зручна в експлуатації, можлива робота з компонентами та шарами, а також віддаленим доступом з будь-якого пристрою. InVision використовується для створення інтерактивного прототипу та тестування на кожному етапі створення інтерфейсу.

4. Визначено складові інтерфейсу державної установи. Проведено попередній аналіз інформації, створено мапу сайту. Було визначено потенційних користувачів та цільову аудиторію, проведено аналіз інтерфейсів державних установ та проведені інтерв'ю з потенційними користувачами та фахівцями області. Створено lo-fi вайрфрейми та побудовано user-flow. Побудовано hi-fi вайрфрейми, та проаналізовано результати тестування даних вайрфреймів на цільовій аудиторії.



5. За допомогою програм Figma та InVision створено інтерфейс державної установи "Київський міський центр громадського здоров'я", та реалізовано інтерактивний прототип, на основі аналітичних даних та реалізованих вайрфреймів.

Практична значимість роботи полягає у комплексному дослідженні розробки інтерфейсу державної установи, як засобу для розширення можливостей доступу до інформації, та її обміну, також вдосконалення технологій процесів проектування інтерфейсу державної установи, та може бути використано при проведенні практичних занять зі студентами, для поглибленого вивчення курсу розробки та проектування інтерфейсу.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

- 1 Круг С. Don't Make Me Think / Стів Круг., 2001.
- 2 Норман Д. Дизайн привычных вещей / Дон Норман., 2018. – 500 с.
- 3 Купер А. About Face: The Essentials of Interaction Design / Алан Купер.. – 720 с..
- 4 The Best Interface Is No Interface: The simple path to brilliant technology (Voices That Matter) / Golden Krishna., 2015. – 243 с.
- 5 Гото К., Котлер Э. Веб-редизайн: книга Келли Гото и Эмили Котлер, 2-е издание/ К.Гото, Э. Котлер, СПб.: Символ-Плюс, 2006. – 416 с.
- 6 Бородаев Д. Веб-сайт как объект графического дизайна / Д. Бородаев, Харьков: ООО МПФ «Септима ЛТД», 2006. – 288 с.
- 7 Гусарь Ю. Какие бывают типы вёрстки сайта [Электронный ресурс]. Режим доступа: <https://impuls-web.ru/kakie-byvayut-tipy-vyorstki-sajta/>.
- 8 Интернет-грамотность по сайтам Из чего состоит сайт, или основные структурные блоки сайта [Электронный ресурс]. Режим доступа: <https://www.inetgramotnost.ru/sajt/iz-chego-sostoit-sajt-ili-osnovnyie-strukturnye-bloki-sajta.html>.
- 9 Макнейл П. Веб-дизайн. Книга идей веб-разработчика / П. Макнейл, под ред. В. Черник, СПб.: Питер, 2014. – 288 с.
- 10 Класифікація інтерфейсів [Електронний ресурс] – Режим доступу до ресурсу: [https://ua-referat.com/%D0%9A%D0%BB%D0%B0%D1%81%D0%B8%D1%84%D1%96%D0%BA%D0%B0%D1%86%D1%96%D1%8F\\_%D1%96%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81%D1%96%D0%B2](https://ua-referat.com/%D0%9A%D0%BB%D0%B0%D1%81%D0%B8%D1%84%D1%96%D0%BA%D0%B0%D1%86%D1%96%D1%8F_%D1%96%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81%D1%96%D0%B2).
- 11 Поморова О. В. Проектування інтерфейсів користувача [Електронний ресурс] / О. В. Поморова, Т. О. Говорущенко. – 2011. – Режим доступу до ресурсу: <http://elar.khnu.km.ua/jspui/bitstream/123456789/1415/2/Rozdil1.pdf>.
- 12 Дизайн Рівню Представлення [Електронний ресурс] – Режим доступу до ресурсу: <https://learn.ztu.edu.ua/mod/book/view.php?id=278&chapterid=86>.

13 Порядок функціонування веб-сайтів органів виконавчої влади [Електронний ресурс]. – 2015. – Режим доступу до ресурсу: <https://zakon.rada.gov.ua/laws/show/z1022-02#Text>.

14 Каліщук Н. О. Веб-сайт як документально-інформаційна система: функцій, структура, зміст [Електронний ресурс] / Наталія Олександрівна Каліщук – Режим доступу до ресурсу: <http://dspace.wunu.edu.ua/bitstream/316497/17608/1/%d0%9c%d0%90%d0%93.%20%d0%9a%d0%9b%d0%86%d0%a9%d0%a3%d0%9a.pdf>.

15 Державний комітет телебачення і радіомовлення України [Електронний ресурс]. – 2019. – Режим доступу до ресурсу: <http://ratadmin.gov.ua/press-tsentr/item/7397-derzhavnyi-komitet-telebachennia-i-radiomovlennia-ukrainy>.

16 Нильсен Я. Web-дизайн: удобство использования Web-сайтов / Я. Нильсен, Х. Лоранжер, под ред. В.С. Иващенко, М.: ООО «И.Д. Вильямс», 2009. – 368 с.

17 Орлов Л. Web-сайт без секретов / Л. Орлов, М.: Бук-пресс, 2006. – 512 с.

18 Самсонов В. Методи та засоби Інтернет-технологій: навч. посібник / В. Самсонов, А. Єрохін, Х.: Компанія СМІТ, 2008. – 264 с.

19 Фельке-Моррис Т. Большая книга веб-дизайна / Т. Фельке-Моррис, М.: Эксмо, 2012. 608 с.

20 Шигина Н.А. Web- дизайн Учебное пособие / Н.А. Шигина, Пенза: Пенз. гос. технол. ун-та, 2015. – 157 с.

21 MDN web docs Структура документа и веб-сайта. [Електронний ресурс]. Режим доступу: <https://developer.mozilla.org/ru/docs/Learn/HTML/>.

22 RebusMedia Основные структурные компоненты страницы сайта [Електронний ресурс]. Режим доступу: <http://rebusmedia.ru/osnovnyie-strukturnyie-komponentyi-stranitsyi-sayta>.

23 WebStudio2U Веб-дизайн: логическая структура сайта [Електронний ресурс]. Режим доступу: <http://webstudio2u.net/ru/design-web/403-structure-models.html>.

- 24 Овчинников Р. Корпоративный веб-сайт на 100 %. Требуйте от сайта большего! / Р. Овчинников, С. Сухов, СПб: Питер, 2009. – 320 с.
- 25 Лебедев С. Web-дизайн. Учебное пособие по созданию публикаций для Интернет, 3-е изд. / С. Лебедев, М.: Бином-пресс, 2004. – 736 с.
- 26 Мак-Дональд М. Создание Web-сайта. Недостающее руководство / М. Мак-Дональд, 3-е изд., СПб: БХВ-Петербург, 2013. – 624 с.
- 27 Интернет-издательство «АртВебМедиа» Модули сайта [Электронный ресурс]. Режим доступа: <http://www.artwebmedia.ru/service/making-sites/moduli/?q=383>.
- 28 Мальцев И. Проектирование сайтов / И. Мальцев, SelfPub, 2018. – 118 с.
- 29 Супрун I. A. Fillin — financial literacy app, design process step-by-step [Электронный ресурс] / Илья Анатолійович Супрун. – 2020. – Режим доступа до ресурсу: <https://uxplanet.org/fillin-financial-literacy-app-design-process-step-by-step-15b1026e25ad>.
- 30 Texiera F. Storyframes before wireframes [Электронный ресурс] / Fabrico Texiera. – 2017. – Режим доступа до ресурсу: <https://uxdesign.cc/storyframes-before-wireframes-starting-designs-in-the-text-editor-ec69db78e6e4>.
- 31 Monteiro M. Design Is a Job / М. Monteiro, Е. Spiekermann. – New York: A Book Apart, 2012. – 135 с.
- 32 Кемпкенс О. Дизайн-мышление. Все инструменты в одной книге / Оливер Кемпкенс. – Киев: Форс, 2020. – 224 с.
- 33 Беркун С. Сделано. Проектный менеджмент на практике / Скотт Беркун. – Москва: Манн, Иванов и Фербер, 2019. – 560 с.