

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ**

**Факультет кібербезпеки та програмної інженерії  
Кафедра інженерії програмного забезпечення**

ДОПУСТИТИ ДО ЗАХИСТУ  
Завідувач кафедри  
Катерина Нестеренко

“ \_\_\_\_ ” \_\_\_\_\_ 2023 р.

**КВАЛІФІКАЦІЙНА РОБОТА  
(ПОЯСНЮВАЛЬНА ЗАПИСКА)**

**ВИПУСНИКА ОСВІТНЬОГО СТУПЕНЯ  
“ МАГІСТРА ”**

**Тема:** “Методика та програмний застосунок психологічної допомоги населенню”

**Виконавець:** Василенко Максим Вікторович

**Керівник:** к.т.н. доцент Микола Федорович Радішевський

**Нормоконтролер:** ст.в Гололобов Дмитро Олександрович

Київ 2023

# НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

**Факультет** кібербезпеки та програмної інженерії

**Кафедра** інженерії програмного забезпечення

**Освітній ступінь** магістр

**Спеціальність** 121 Інженерія програмного забезпечення

**Освітньо-професійна програма** «Програмне забезпечення систем»

ЗАТВЕРДЖУЮ

Завідувач кафедри

Катерина Нестеренко

" \_\_\_ " \_\_\_\_\_ 2023 р

## ЗАВДАННЯ

на виконання кваліфікаційна роботи студента  
Василенка Максима Вікторовича

1. Тема кваліфікаційної роботи: «Методика та програмний застосунок психологічної допомоги населенню»  
затверджена наказом ректора від 29.09.2023 р. № 1994/ст
2. Термін виконання кваліфікаційної роботи: з 02.10.2023 р. до 31.12.2023 р.
3. Вихідні данні до кваліфікаційної роботи: розробити програмний продукт на основі .NET 6.0 з використанням Angular
4. Зміст пояснювальної записки:
  1. Аналіз сучасного стану психологічної допомоги населенню.
  2. Методика взаємодії лікаря-психолога з клієнтами.
  3. Структура інформаційної системи.
  4. Прототип програмного застосунку психологічної допомоги населенню.
5. Перелік обов'язкових слайдів презентації:
  1. Описання ідеї та новаторства теми.
  2. Аналіз сучасного стану психологічної допомоги населенню.
  3. Завдання розробки.
  4. Безпека даних користувачів.
  5. Структура архітектури.
  6. Прототип програмного застосунку психологічної допомоги населенню.

## 6. Календарний план-графік

№ пор	Завдання	Термін виконання	Відмітка про виконання
1.	Ознайомлення з постановкою задачі та вивчення літератури. Написання 1 розділу, представлення керівнику	02.10.23 – 08.10.23	
2.	Попередній друк 1 розділу та допоміжних сторінок (черновик) - титульної, завдання, графіка, реферат, список скорочень, зміст, вступ, список джерел. 1-ий нормо-контроль.	09.10.23– 15.10.23	
3.	Підготовка остаточного плану та узгодження його з керівником	16.10.23– 22.10.23	
4.	Обговорення та редагування першого розділу з керівником, підготовка до написання другого розділу	23.10.23– 29.10.23	
5.	Написання другого розділу, представлення керівнику, підготовка до написання третього розділу	30.10.23 – 05.11.23	
6.	Написання 3 розділу, представлення керівнику, форматування та виправлення існуючого тексту.	06.11.23 – 19.11.23	
7.	Загальне редагування та друк пояснювальної записки, графічного матеріалу. Отримання відгуку керівника, рецензії та заповнення листа про добросовісність. Розробка доповіді	20.11.23 – 03.12.23	
8.	Завершення написання ПЗ. Проходження нормо-контролю. Друк ПЗ. Отримання відгуку керівника. Підготовка презентації та доповіді на перед захист.	04.12.23 - 10.12.23	
9.	Передзахист каліф. Роботи. Отримання рецензії	11.12.23 - 17.12.23	
10.	Підготовка документів до захисту та здача їх секретарю ДЕК	18.12.23 - 24.12.23	
11.	Захист дипломної роботи перед ЕК	25.12.23 - 31.12.23	

Дата видачі завдання 02.10.2023

Керівник: к.т.н. доцент Микола РАДШЕВСЬКИЙ

Завдання прийняв до виконання: Максим ВАСИЛЕНКО

Дата 02.10.2023

## РЕФЕРАТ

Пояснювальна записка до дипломної роботи «Методика та програмний застосунок психологічної допомоги населенню»: 97 с., 39 рис., 0 табл., 32 інформаційних джерел.

**МЕТОДИКА ПСИХОЛОГІЧНОЇ ДОПОМОГИ, ВЗАЄМОДІЯ З НАСЕЛЕННЯМ, АВТОМАТИЗАЦІЯ ПСИХОЛОГІЧНОЇ ДОПОМОГИ, ПОКРАЩЕННЯ СТАНУ ЛЮДИНИ**

**Об'єкт розробки** – програмний засіб для надання інформаційного сервісу роботи з населенням для покращення психологічного здоров'я

**Мета роботи** – покращення психологічного стану населення та її автоматизація взаємодії.

**Метод дослідження** – спостереження сучасного стану психологічної допомоги населенню.

В процесі роботи був досліджений великий спектр впливу на психоемоційний стан людини, порівняння існуючих практик для його покращення.

**Результати** роботи можуть бути використанні під час проектування застосунку на темі психоемоційного стану людини та покращення автоматизації взаємодії людського зв'язку опираючись на різні фактори, ситуацію та тенденції в сучасному світі.

Розробка та дослідження проводилися під управлінням ОС Windows 10. Розробка програми проводилася у середовищі .NET Core, а саме 6-ої версії, на мові програмування C#.

## ABSTRACT

Explanatory note to the diploma work "Methods and software application of psychological assistance to the population": 97 p., 39 figures, 0 tables, 32 information sources.

METHODS OF PSYCHOLOGICAL ASSISTANCE, INTERACTION WITH THE POPULATION, AUTOMATION OF PSYCHOLOGICAL ASSISTANCE, IMPROVEMENT OF HUMAN CONDITION

**Object of development** - a software tool for providing an information service for working with the population to improve psychological health

**Purpose** - to improve the psychological state of the population and automate its interaction.

**Research method** - observation of the current state of psychological assistance to the population.

In the course of the work, a wide range of influences on the psycho-emotional state of a person was studied, as well as a comparison of existing practices to improve it.

**The results** of the work can be used when designing an application on the topic of the psycho-emotional state of a person and improving the automation of human interaction based on various factors, situations and trends in the modern world.

The development and research were carried out under the control of Windows 10. The program was developed in the .NET Core environment, namely version 6, in the C# programming language.

## ЗМІСТ

<b>ВСТУП.....</b>	<b>10</b>
<b>РОЗДІЛ 1 АНАЛІЗ СУЧАСНОГО СТАНУ ПСИХОЛОГІЧНОЇ ДОПОМОГИ НАСЕЛЕННЮ .....</b>	<b>8</b>
1.1 Аналіз сегменту психоемоційного стану людини та інтеграція методики до програмного забезпечення .....	8
1.2 Ідея, актуальність і новаторство розробки інформаційної системи .....	9
1.3 Дослідження реалій ринку та аналіз існуючих проектних аналогів за темою методика та програмний застосунок психологічної допомоги населенню .....	11
Висновки.....	28
<b>РОЗДІЛ 2 МЕТОДИКА ВЗАЄМОДІЇ ЛІКАРЯ-ПСИХОЛОГА З КЛІЄНТАМИ</b>	<b>29</b>
2.1 Сутність та основні етапи розвитку психологічної підтримки.....	29
2.1.1 Важливість дослідження та аспекти тематики психологічної підтримки...	30
2.1.2 Ціль та мета психологічної допомоги людині.....	32
2.1.3 Сутність проблематики у галузі психологічної допомоги та шляхи вирішення .....	33
2.1.4 Актуальні проблеми у галузі психологічної підтримки в Україні та їх вирішення .....	34
2.2 Сучасні психологічні виклики, з якими стикається населення.....	35
2.2.1 Вплив та наслідки війни на психоемоційний фон людини.....	37
2.2.2 Найкращі практики у виявленні та вирішенні психологічних проблем населення.....	39
2.3 Вимоги до застосунку з психологічної допомоги населенню.....	41
2.3.1 Бізнес вимоги до програмного застосунку .....	42
2.3.2 Функціональні вимоги до програмного застосунку .....	43
2.3.3 Користувацькі вимоги до програмного застосунку.....	44

2.3.4 Системні вимоги до програмного застосунку .....	46
2.3.5 Нефункціональні вимоги до програмного застосунку .....	48
Висновки .....	52
<b>РОЗДІЛ 3 СТРУКТУРА ІНФОРМАЦІЙНОЇ СИСТЕМИ .....</b>	<b>53</b>
3.1 Моделі сучасного проектування .....	53
3.2 Проектування архітектури систем програмного забезпечення.....	56
3.2.1 Специфіка архітектури при побудуванні web-застосунків.....	56
3.2.2 Монолітна архітектура при побудуванні застосунку .....	58
3.2.3 N-tier архітектура при побудові застосунку .....	60
3.2.4 Сервісно-орієнтована архітектура при побудові застосунку.....	62
3.2.5 Мікро-сервісна архітектура при побудові застосунку .....	63
3.3 Архітектура у проектуванні API застосунків .....	65
3.3.1 Побудова на базі MVC .....	66
3.3.2 Побудова на базі MVP .....	67
3.3.3 Побудова на базі MVVM .....	68
3.4 Побудова ПЗ та інтеграція з системою контролю версій Git.....	69
3.5 Розділення моделі користувача .....	69
3.6 Проектування програмного засобу з психологічної допомоги.....	70
3.7 Специфікація автентифікації .....	72
3.7.1 Історія типів автентифікації їх розвиток.....	72
3.7.2 Детальніше про cookie-файли .....	73
3.7.3 Детальніше про Web Tokens та JWT .....	74
Висновки .....	81
<b>РОЗДІЛ 4 ПРОТОТИП ПРОГРАМНОГО ЗАСТОСУНКУ ПСИХОЛОГІЧНОЇ ДОПОМОГИ НАСЕЛЕННЮ .....</b>	<b>83</b>
4.1 Побудова прототипу для психологічної допомоги населенню.....	83
4.1.1 Створення прототипу .....	83
4.2 Аналіз прототипу .....	84

4.2.1	Опис функціоналу .....	85
4.2.2	Представлення інтерфейсу користувача .....	86
4.3	Покращення системи .....	88
	Висновки .....	89
	<b>ВИСНОВКИ.....</b>	<b>90</b>
	<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....</b>	<b>91</b>



## **ПЕРЕЛІК ПРИЙНЯТИХ СКОРОЧЕНЬ**

API – Application Programming Interface

REST - REpresentational State Transfer

GUI – Graphical User Interface

UI - User Interface

CEO – Chief Executive Officer

HTTP – Hypertext Transfer Protocol

SOAP – Simple Object Access Protocol

SOA – Services-oriented architecture

JWT – JSON Web Token

JSON – JavaScript Object Notation

HMAC - Hash-based message authentication code

SHA256 - Secure Hash Algorithm 256-bit

RSA - Rivest–Shamir–Adleman

ASCII - American Standard Code for Information Interchange

VCS - Version Control System

## ВСТУП

Головна ідея дипломної роботи полягає в розгляді основної сутності людини, яка в стремлінні до поліпшення свого життя та загальної тенденції автоматизації оточуючого середовища призвела до швидкого розвитку цифрової технології в усіх сферах, що стосуються людського життя. Знаходячись в епоху цифровізації, ця тема залишається актуальною і в наш час, і на перспективу.

З урахуванням того, що кожна особа має свої психологічні потреби, пов'язані як з власною психологічною стійкістю, так і з наданням психологічної допомоги оточуючим, цифрова трансформація цієї сфери стає надзвичайно важливою. Створення програмного засобу, спрямованого на покращення психологічного стану особистості, може виявитися надзвичайно корисним в сучасному світі.

Актуальність реалізації такого проекту підкреслюється тим, що, незважаючи на зміни в світі, люди завжди будуть шукати рішення для покращення свого психічного стану, зробленого доступним цифровими засобами. Таким чином, головною метою створення цього програмного засобу є розробка сервісу, який залишатиметься актуальним завжди, допомагаючи людям подолати складнощі сучасного життя.

До того ж, під час моєї роботи був проведений ретельний аналіз ринку, ідентифіковані основні проблеми існуючих аналогів, які можна вирішити шляхом удосконалення. Також під час практичної діяльності було вивчено актуальність та проведений аналіз предметної області, що дозволило сформулювати основну ідею програмного засобу для надання психологічної допомоги населенню.

# РОЗДІЛ 1

## АНАЛІЗ СУЧАСНОГО СТАНУ ПСИХОЛОГІЧНОЇ ДОПОМОГИ НАСЕЛЕННЮ

Оскільки в сучасному світі інформаційні технології занадто сильно почали впливати на наше життя, настільки що вже не можеш уявити свій день без них. Кожен день відбувається мільярд або більше операцій в цих системах. Це може бути звичайний запит цікавої для вас книжки, чи перегляд карт для пошуку маршруту, можуть бути онлайн покупки чи перевірка балансу в сервісі онлайн-банкінгу, тощо.

Також враховуючи останні тенденції в світі, такі як цифровізація, збільшення комфорту для людей, але з проблемами які присутні у людей в будь-який час та які викликають вони ж між собою – конфлікти. Така тенденція є звичною для людини та буде завжди продовжуватись протягом росту людства, окрім цього продовжується і все більша цифровізація. Тому однією з таких актуальних тем є автоматизація всього чим користуються люди, особливо якщо це те з чим людям потрібно взаємодіяти практично весь час. Одним з таких тем є автоматизація лікарень, а саме лікарень покращення стану населення та збереження дорогоцінного часу з обох боків, як пацієнта, так і лікаря.

Отже, на фоні цих факторів, мною було винесено ідею для створення такого програмного засобу для покращення стану людей, оскільки тема є дуже обширною, то було прийнято рішення звузити обсяг до дослідження та проектування засобу для взаємодії та покращення з психоемоційним станом людей на фоні останніх подій.

Для цього в цьому розділі було розглянуто та проаналізовані вже існуючі на ринку послуги з цієї галузі.

### **1.1 Аналіз сегменту психоемоційного стану людини та інтеграція методики до програмного забезпечення**

Людина, як і будь-яка інша істота живе та розвивається, запускаючи прогрес. Оскільки люди є набагато складнішим організмом який до сих пір не є досягнутий

повністю. Отже, кожен день відкриваються все більше та нові галузі пов'язані з людьми.

Однією з базових завжди була галузь здоров'я, а саме турботи про свій організм, зародилося це ще далеко за наших часів, до нашої ери. Спочатку наші пращури використовували для лікування та профілактики рослини які їх оточували, так століттями люди з покоління в покоління покращували та передавали свої знання, пізніше були винайдені перші таблетки і так все йшло разом з людиною до прогресу.

Під час цих етапів також розвивались та з'являлися інші галузі, які були пов'язані з здоров'ям, але по при це розділені за типом і однією з таких є психологія. Хоча спочатку це була не дуже зацікавлена галузь, але тепер можна сказати інакше, адже практично всі люди, хоча б трохи, але хочуть пізнати себе для покращення освіченості та покращення свого ж життя. [5]

Оскільки час змінюється і ще не декілька століть, а всього століття було складно уявити, що людина буде настільки пов'язана з прогресом, що відбудеться інтеграція технологій у все можливі аспекти життя. Ще століття тому телефони, інтеграція життя з мережею були утопією, а сьогодні реальністю, ще двадцять років тому інтелект і роботи здавались також утопією, але тепер вони також інтегруються в життя, хоча цей процес лише на самому початку.

Отже, інтеграція психоемоційних методик до програмного продукту є в край важливою, адже сучасна людина не може витратити час задарма, вона, як покращена стадія сотень еволюцій хоче здобувати якомога більше інформації у якомога короткий термін, тому подібна інтеграція буде корисна для людства.

## **1.2 Ідея, актуальність і новаторство розробки інформаційної системи**

В умовах геополітичних конфліктів і війни, стає очевидною необхідність у створенні інноваційного психологічного додатку, який забезпечує доступ до психологічної підтримки для військових та громадян, які постраждали від цих подій. Цей додаток необхідний для покращення емоційного стану та психічного здоров'я тих, хто пережив важкі травматичні ситуації.

Подібно до того, як інформаційні технології змінили наше сприйняття світу та перетворили спосіб, яким ми взаємодіємо з інформацією, цей психологічний додаток може перетворити спосіб, яким ми реагуємо на стресові ситуації та травми. Цей інструмент може надавати доступ до різноманітних психологічних тестів, методів релаксації та консультацій, допомагаючи користувачам зміцнити свою психіку та відновити психологічну стійкість.

Проте, особливістю цього застосунку є те що, кожен користувач зможе зменшити витрати часу на рутинні речі, наприклад, при відвідуванні лікаря, він як завжди запитає важливі, але банальні речі: «Що вас турбує?», «Які є скарги?», «Який діагноз підозрюєте?», тощо. Всі подібні запитання є вкрай важливими, але часозатратними, тому як би у лікаря був листок з відповіддю на всі цікаві для нього запитання – це зекономить час на них, тому більше пацієнтів буде за один день, а тому в місяць це вже сотні вилікуваних пацієнтів.

Розвиток цього додатку може викликати інтерес інвесторів, які вбачають в ньому можливість забезпечити необхідну психологічну підтримку та лікування, для тих, хто переживає війну та її наслідки. Ці інвестиції можуть допомогти створити інструмент, який буде активно залучений до підтримки та одужання військових та постраждалих, і сприятиме формуванню їхнього психологічного благополуччя. Поява такого додатку може стати важливим кроком в зміні підходів до психологічного здоров'я в умовах військових конфліктів та стресового середовища. Він може стати важливим інструментом для підтримки емоційного здоров'я і відновлення психологічного балансу у людей, які пережили важкий та травматичний досвід війни.

Головною метою кваліфікаційної роботи є створення програмного засобу що зміг би допомогти людям витримати ці складні часи для більшості людей. Актуальність реалізації цього проекту полягає в тому що не зважаючи на ситуацію в світі люди в будь-якому разі знайдуть причину для переживання чи інших психологічних проблем, які, на жаль, можуть переходити у дещо більше. Тобто головна ціль створення додатку для покращення психоемоційної взаємодії.

### 1.3 Дослідження реалій ринку та аналіз існуючих проектних аналогів за темою методика та програмний застосунок психологічної допомоги населенню

Для правильного побудування та аналізу ринку, мною було вирішено провести аналіз існуючих додатків за схожою тематикою та виділити їх позитивні та негативні рішення для врахування цих рішень у своєму проекті.

#### Проект «UA Mental Help»

Це благодійний організація у вигляді веб-додатку, яка має декілька безкоштовних консультацій з психологами-волонтерами для осіб, що відчують симптоми депресії, розпачу, апатії та інших тривожних станів. Ця організація пропонує підтримку та допомогу тим, хто потребує, і допомагає їм знайти шлях до поліпшення психічного стану. [3]

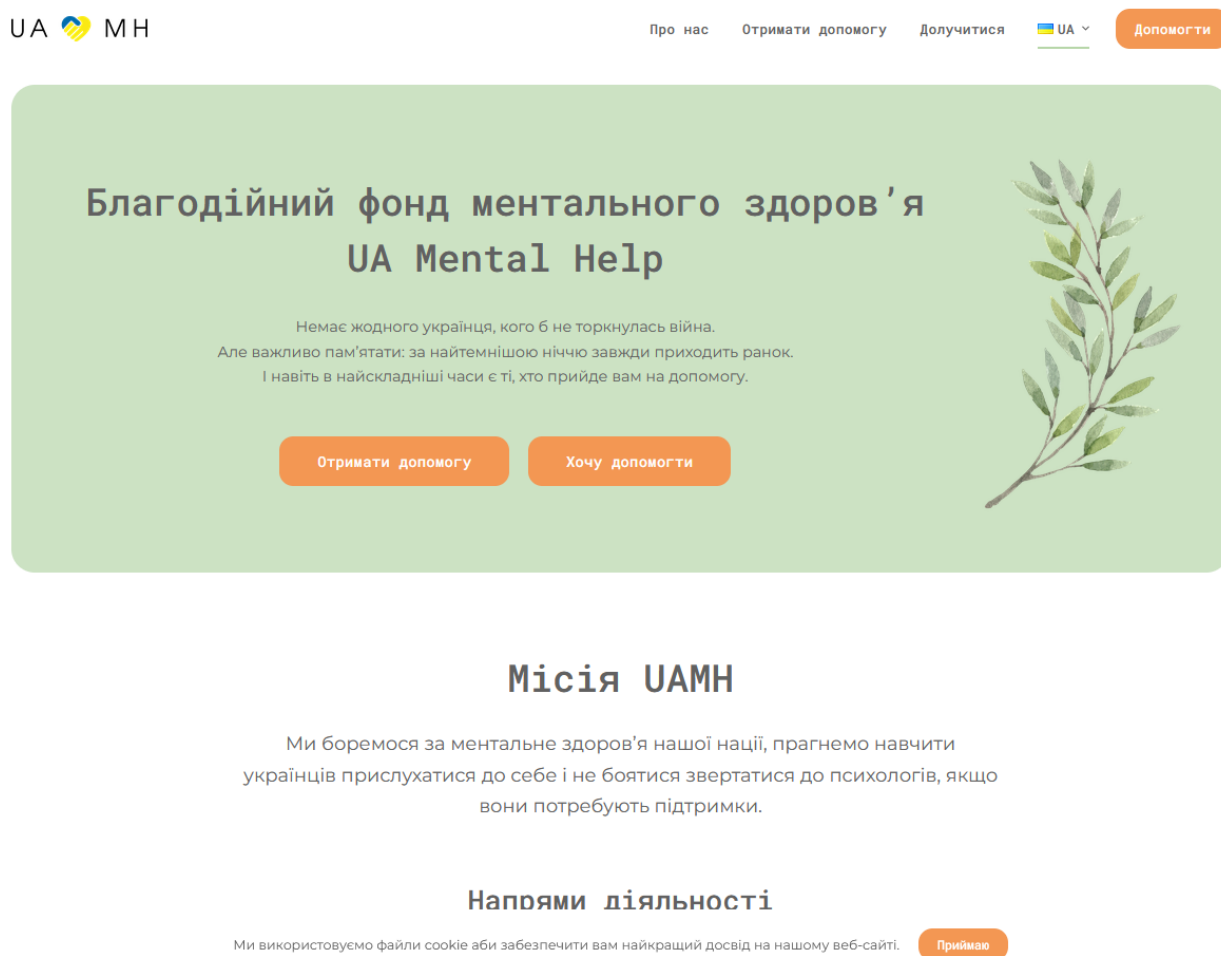


Рис 1.1 Знімок екрану головної сторінки веб-застосунку «UA Mental Help»

Даний програмний засіб призначений на людей, котрі потребують психологічної допомоги. Як зазначено на малюнку вище, кожен користувач може приєднатися до цієї організації, оскільки вона є благодійною, тобто можна як отримати допомогу, так і надавати, це може бути гарним проявом для волонтерів, що збільшує кількість користувачів цієї системи.

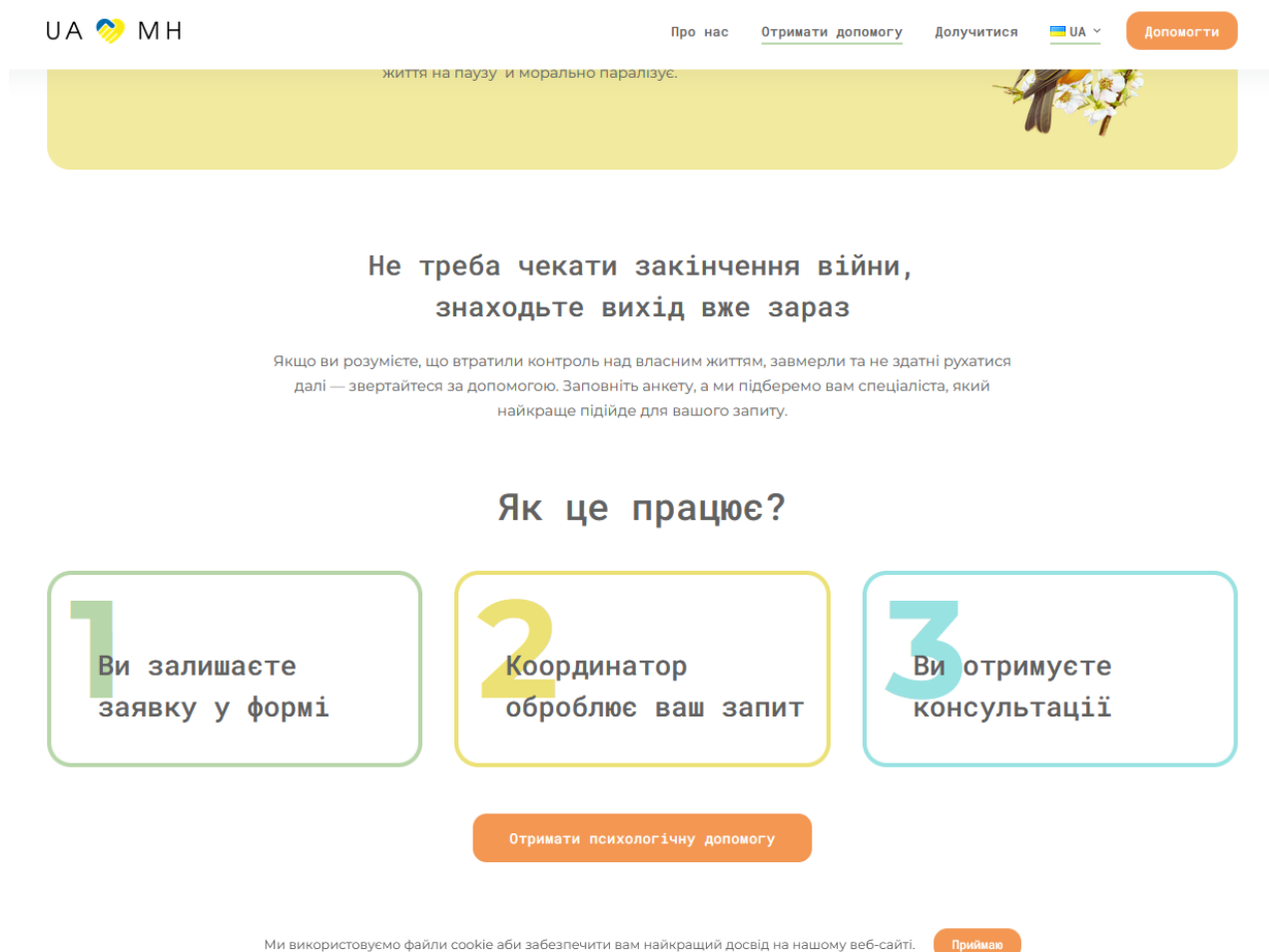


Рис 1.2 Знімок екрану програмного застосунку з використання ПЗ

**Використання системи:** цей програмний застосунок працює за наступним алгоритмом, що зазначений на рисунку (Див. Рис. 1.2). Отже, кожен користувач, який захоче отримати психологічну допомогу повинен буде заповнити гугл-форму, після чого буде оброблений менеджером, у цього підходу є як негативні так і позитивні риси, наприклад, користувач не зможе негайно почати роботу з лікарем,

що трохи погіршує враження користувача від системи. Також, оскільки цей проект працює повністю безкоштовно – користувачам надається можливість допомогти проекту грошима (Див. Рис. 1.3), так він не залежить від інвесторів та може фокусуватися на своїй основній задачі – психоемоційна допомога.

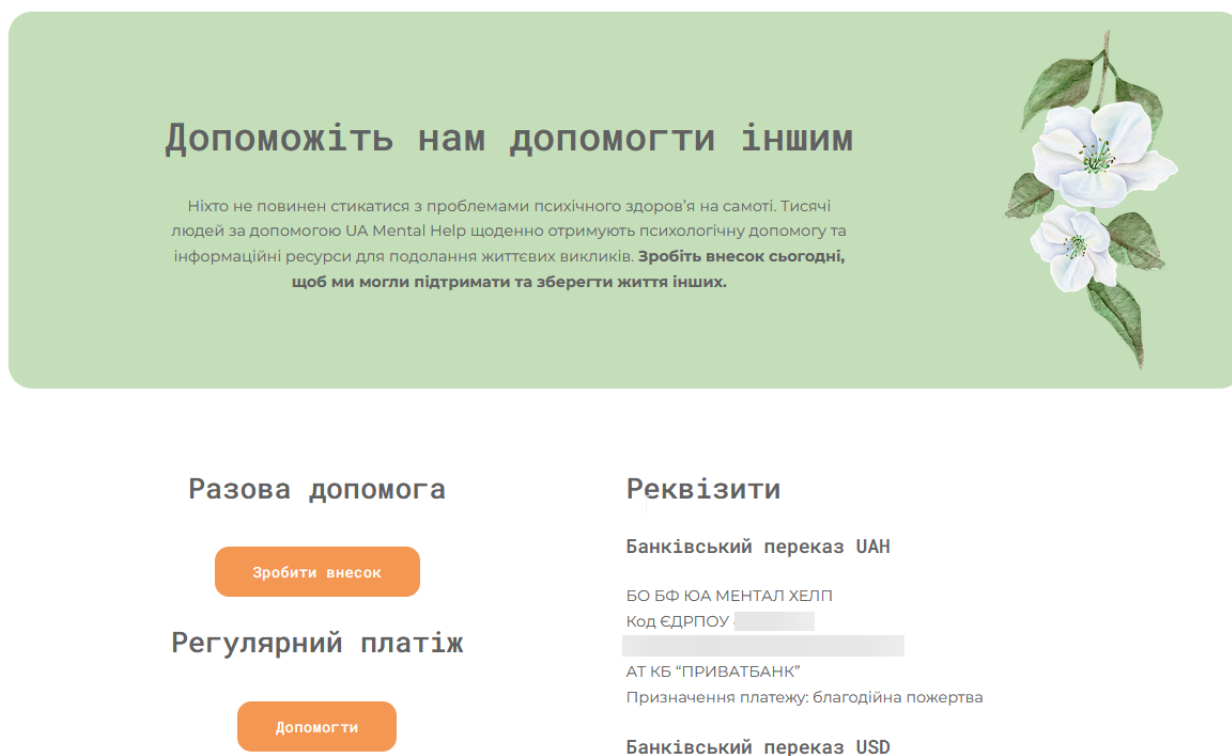


Рис 1.3 Знімок екрану програмного застосунку для допомоги проекту з фінансуванням і залученням людей

Отже, перерахуємо основні позитивні та негативні рішення в додатку. До позитивних можна віднести наступні:

- Легкий та не навантажений інтерфейс;
- Можливість отримання допомоги без реєстрації в системі;
- Має велике охоплення користувачів (за рахунок волонтерів);
- Має декілька безкоштовних консультацій.

Звертаючись до негативних рішень я звернув увагу на:

- Система реалізовує мінімальні можливості;
- Користувач не може одразу приступитися до системи;



– Використання сторонніх сервісів для опитування.

## Проект «Pleso»

Це інноваційна платформа для психотерапії онлайн, яка допомагає знайти саме того психотерапевта, який відповідає вашим потребам та підходить до вашого стилю життя. Наша мета - створити безпечне та підтримуюче середовище для тих, хто шукає допомогу та розробити індивідуальну програму терапії для кожного клієнта. [1]

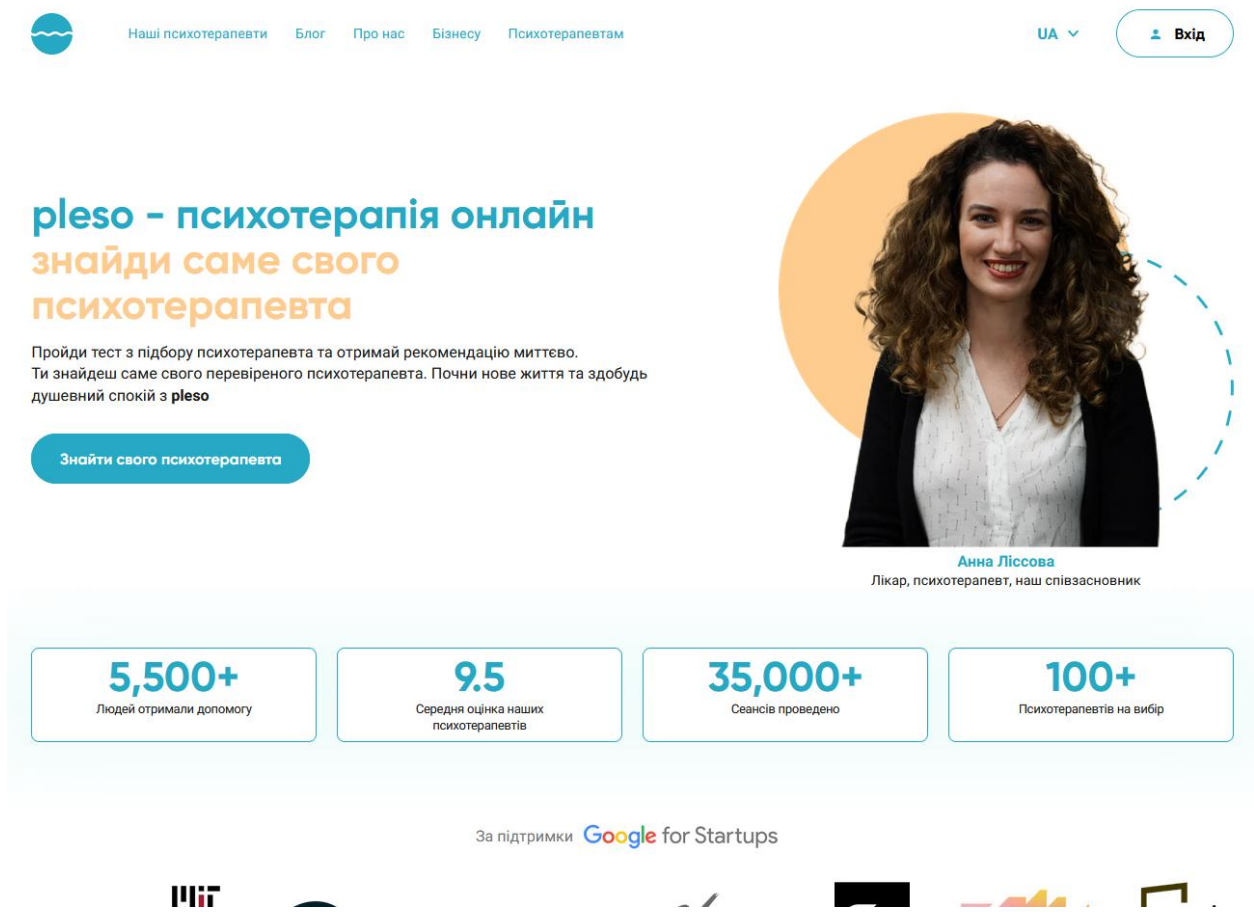


Рис 1.4 Знімок екрану головної веб-сторінки застосунку pleso

Як і попередній програмний засіб цей орієнтований на підтримку населення, хоча він має здебільшого серйознішу реалізацію. Оскільки цей проект є комерційний, отже підхід також інший. Здебільшого, тут є реєстрація користувача і хоча тут не можна зареєструватися як лікар чи волонтер, але ця система спрямована

на гнучку роботу з користувачем. При спробі зареєструватися вперше користувачу буде запропонований легкий тест, для підлаштування під нього (Див. Рис. 1.5)

**З ким тобі буде комфортніше працювати?**

Без різниці

З чоловіком

З жінкою

**З терапевтом якого віку тобі комфортніше працювати?**

До 30

30-40

40+

**В який час тобі буде комфортніше працювати з терапевтом?**

Робочі дні - Денний час

Робочі дні - Вечірній час

Вихідні дні - Денний час

Вихідні дні - Вечірній час

**Далі**

Рис 1.5 Знімок екрану тесту для нових користувачів у застосунку «pleso»

Можна зазначити, що подібне опитування справляє на користувача дуже приємне враження, окрім того такий підхід автоматизує взаємодію між людьми, зберігає час як користувачу, так і лікарю, окрім того понижує ризик неприємних випадків, які іноді можуть траплятися.

Після проходження цієї форми сформується рекомендації, щодо вашого можливого лікування. Отже виглядатиме це наступним чином (Див. Рис. 1.6)

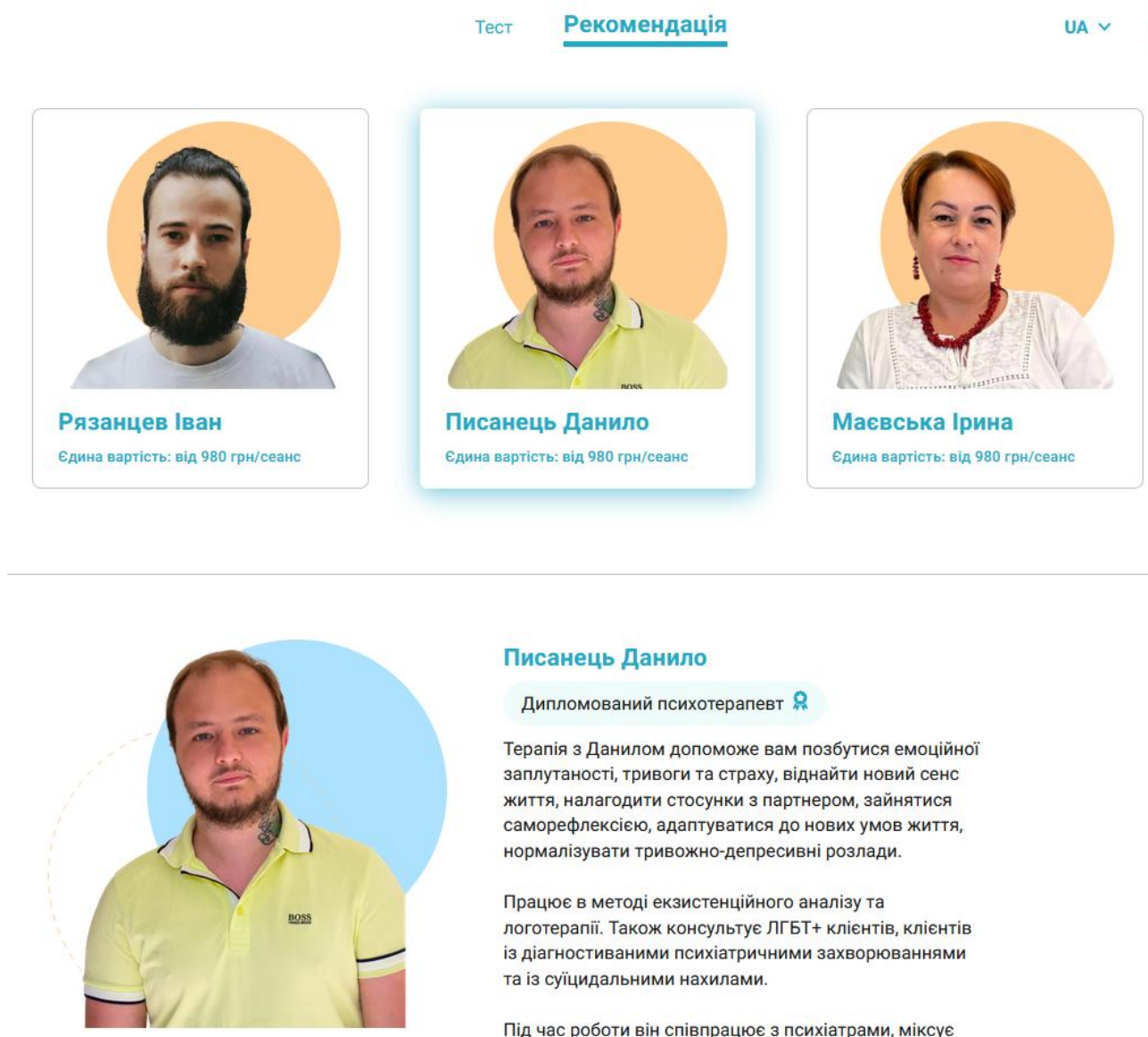


Рис 1.6 Знімок екрану рекомендацій щодо вашої анкети

Перше, що хочеться відмітити, дані рекомендації будуть завантажені одразу, оскільки це відбувається одразу це створює гарне враження для користувача та збільшує його використання цієї системи. В рекомендаціях користувач може обрати

одно з трьох запропонований лікарів, почитати про них інформацію та дізнатися їх методи, оскільки система не є відкритою та є комерційною, всі лікарі – це підтвердженні дипломами та досягненнями лікарі. На сторінці кожного з них є можливість переглядати дипломи та діагнози з якими працює чи категорично ні – лікар (Див. Рис. 1.7)

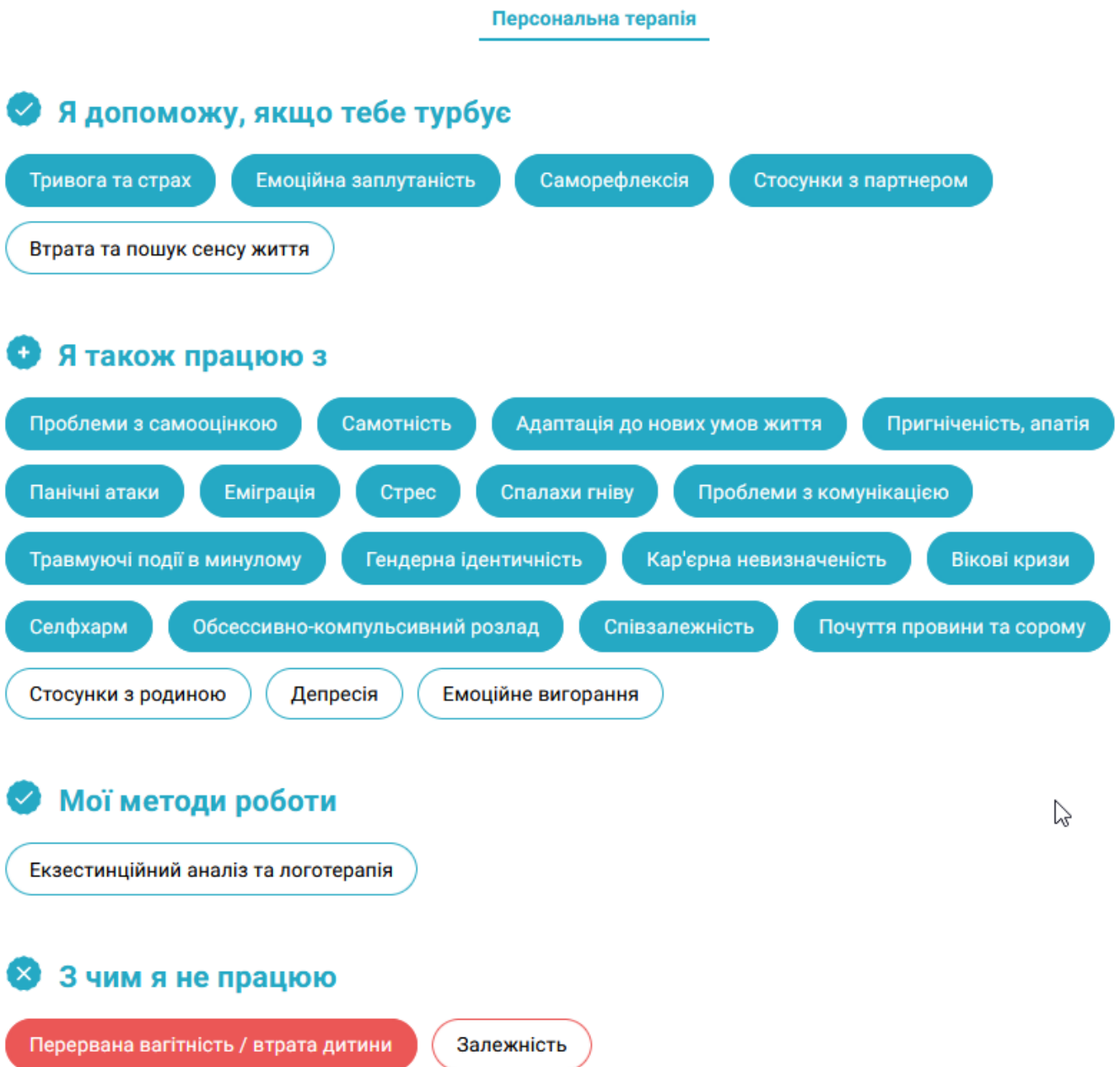


Рис 1.7 Знімок екрану рекомендованого лікаря і його навички та досягнення

Отже, описуючі позитивні та негативні рішення в цьому додатку, я виділив наступні, позитивні:

- Зручна взаємодія з користувачем;
- Автоматизація зайвої взаємодії між людьми;
- Великий перелік інформації про лікаря;
- Можливість запису;
- Має підтверджених та кваліфікованих лікарів.

До негативних рішень я виділи наступне:

- Не надає змоги зареєструватися як сторонній лікар, не від компанії;
- Не великий вибір персоналу;
- Комерційний проект, не надає знижок для постраждалих.

### **Проект «ХАБ СТІЙКОСТІ»**

Це інноваційна платформа підтримки, ініційована благодійним фондом «Брейнкульт» та громадською організацією «Фундація Суспільність». Ця спеціально розроблена ініціатива створена з метою надання всебічної допомоги та підтримки.

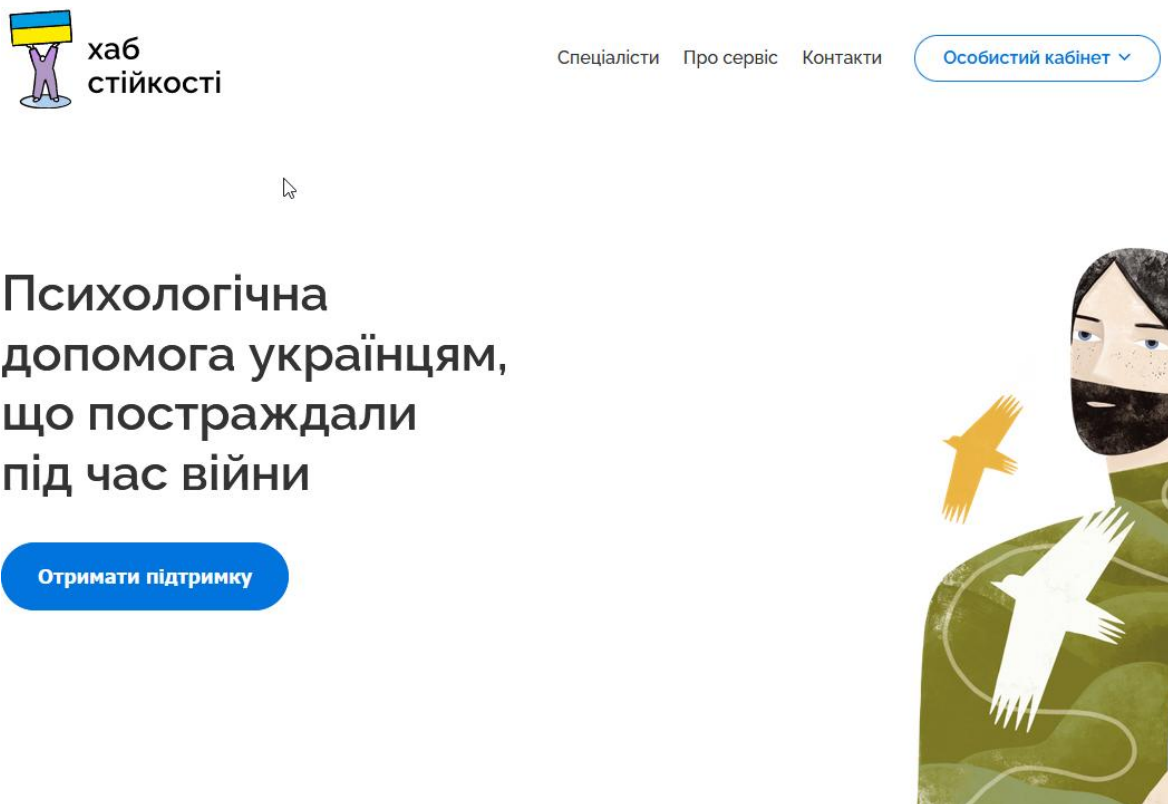


Рис 1.8 Знімок екрану головного веб-застосунку «ХАБ СТІЙКОСТІ»

При вході в систему маємо гарний оформлений дизайн і приємний не навантажений інтерфейс. Користувач має змогу швидко зареєструватись, що не створює проблем і хоча система не пропонує ніякого опитування, вона має свої якості. [4]

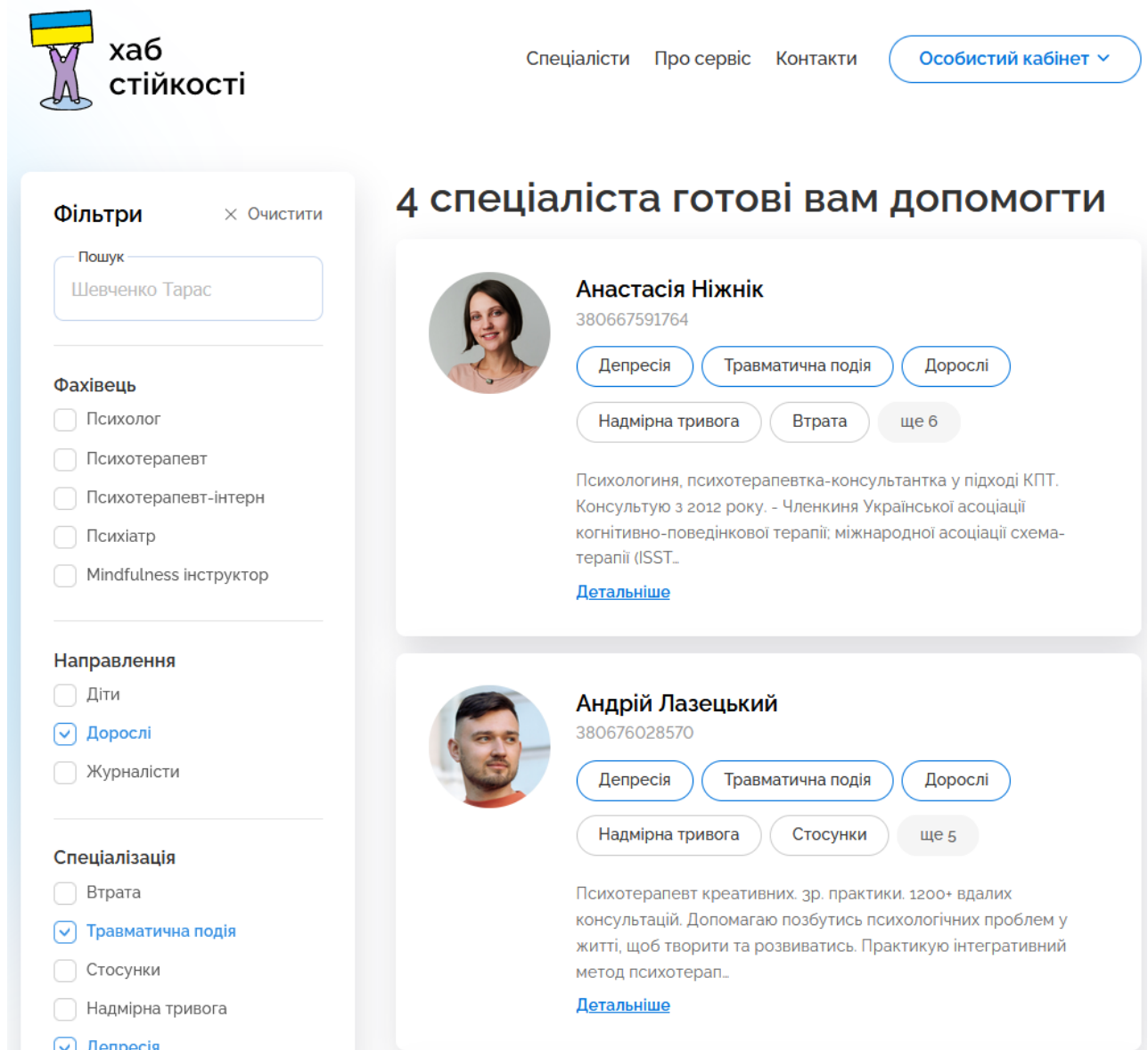


Рис 1.9 Знімок екрану головного веб-застосунку «ХАБ СТІЙКОСТІ»

Після реєстрації користувач переходить до «Особистий кабінет», де йому демонструються його пройдені сеанси та є окрема вкладка «Тести», хоча вона буде доступна лише після назначена спеціального тесту конкретним лікарем.

Отже, як новому користувачу, мені рекомендується записатися до одного з

лікарів, хоча система не передбачає рекомендацій, але при пошуку є достатня кількість налаштувань щоб обрати потрібного для конкретного випадку користувача.

Наприклад, як продемонстровано вище (Див. Рис. 1.9), користувачу пропонується ряд фільтрів за темами: «Фахівець», «Направлення» та «Спеціалізація»:

- *Фахівець.* В цьому розділі обирається спеціальність лікаря аби зменшити час на пошук потрібного фахівця;
- *Направлення.* Цей розділ спрямований на вибір типу запиту самого сеансу;
- *Спеціалізація.* Цей розділ допомагає вибрати за конкретною тематикою цікавого лікаря для вас.

Отже, перерахуємо основні позитивні та негативні рішення цього застосунку.

До позитивних можна віднести наступні:

- Зручний дизайн та інтерфейс;
- Зручний фільтр для вибору лікаря;
- Легка реєстрація;
- Не можна зареєструватися як лікар.

Звертаючись до негативних рішень я звернув увагу на:

- Не має рекомендацій;
- Про користувача не збирається достатньо інформації;
- Система не вирішує проблему зменшення зайвої комунікації;
- Використання не практичних рішень.

### **Проект «РОЗКАЖИ МЕНІ»**

Це інноваційна інтернет-платформа, призначена для надання безкоштовної психологічної підтримки особам, які стикаються з різними проявами емоційного



стресу та психічного нездужання. Головна ідея цього проекту це допомагати тим, хто відчуває тривогу, апатію, панічні атаки чи інші симптоми депресії, знайти шлях до покращення свого психічного стану і якості життя. [2]

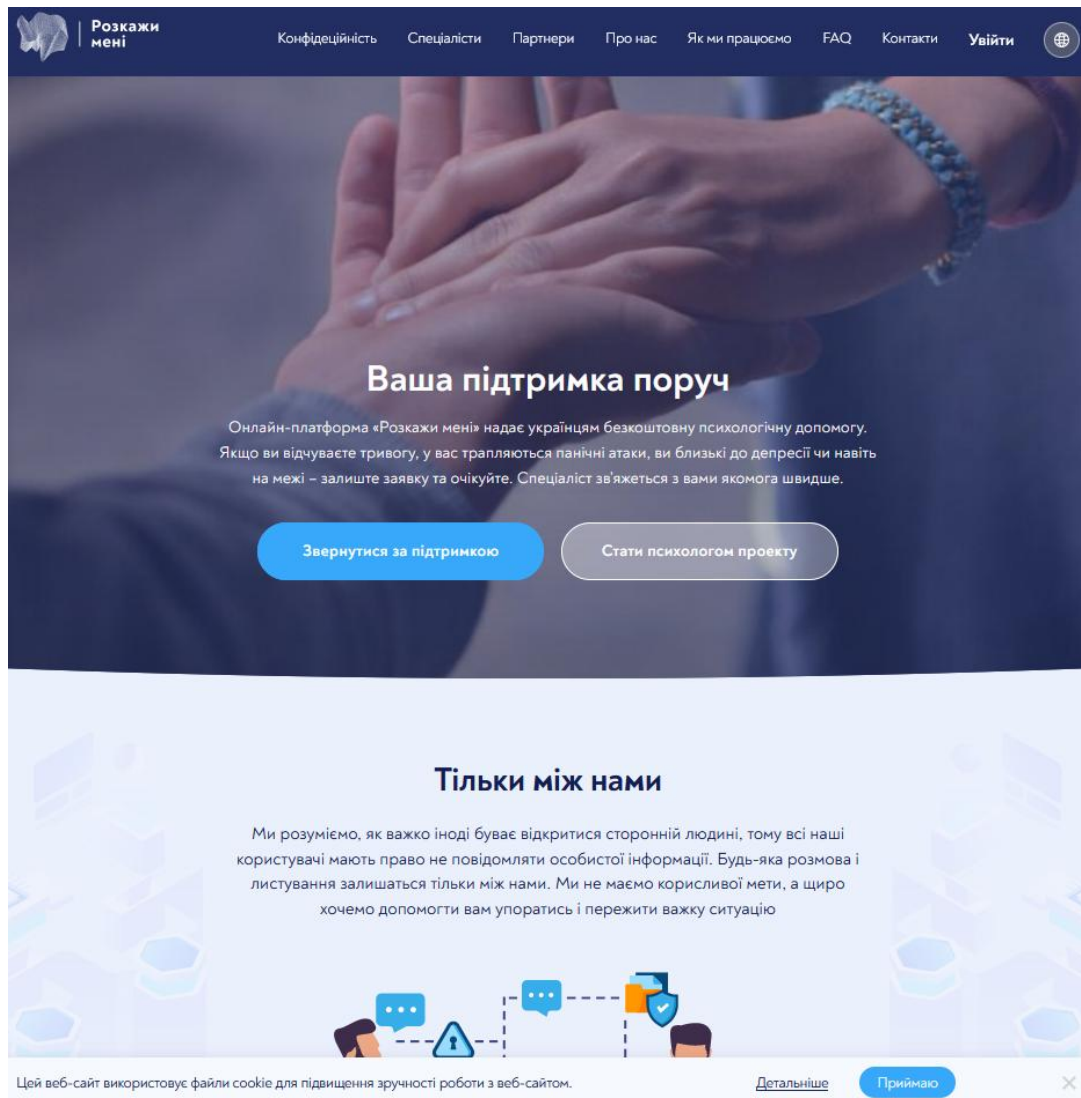


Рис 1.10 Знімок екрану головної сторінки застосунку «Розкажи мені»

При перегляді головної сторінки одразу виділяється меню для реєстрації як користувач чи лікар, отже система має гучність та необмежену кількість лікарів.

Якщо розглядати систему як для користувача, то при натисканні кнопки «Звернутися за підтримкою», викликається меню для реєстрації в системі та створені заявки, яка потребує розгляду. Такий підхід не є практичним, оскільки він не автоматизує, а лише ускладнює комунікацію, до того ж немає практично ніякого тесту для створення рекомендацій, користувач може лише обрати одну з причин,



також він не бачить чи розглянулась вже його заявка. Після заявки користувачу пропонується звернутися до телеграм боту «Друг», але цей функціонал на разі не працює. Щодо функціоналу лікаря, то він дещо схожий, хоча має декілька переваг, по-перше він заповнює форму як і звичайний користувач, але після доповнює це своїм фото та резюме в doc форматі, після пропонується пройти тестування (Див. Рис. 1.11), але інтерфейс не є зручним, та опитування не підв'язується до системи.

ажі

Конфідесійність Спеціалісти Партнери Про нас Як ми працюємо FAQ Контакти

Заздалегідь дякуємо, що змогли знайти час для приєднання до нашої команди. Нас вже майже 1000. Працюємо задля психічного здоров'я України!

[Увійдіть в обліковий запис Google](#), щоб зберегти надані відповіді.  
[Докладніше](#)

Зірочка (\*) указує, що запитання обов'язкове

Електронна адреса \*

Ваша електронна адреса

Ваше Прізвище та Ім'я \*

Ваша відповідь

Ваш профіль в Instagram/Facebook \*

Ваша відповідь

Чи є у вас вища психологічна освіта? \*

Так

Ні

Рис 1.11 Знімок екрану тесту для лікаря під час реєстрації

Серед позитивних рішень цієї системи я виділив наступне:

- Є можливість використовувати систему як сторонній лікар;
- Можна переглянути персонал не зайшовши до системи;

А негативними є наступні:

- Не зручний дизайн та інтерфейс;
- Про користувача не збирається достатньо інформації;
- Система не вирішує проблему зменшення зайвої комунікації;
- Не працюючий функціонал;
- Використання не практичних рішень.

### **Проект «Поруч»**

Проект «Поруч» - це один з найвідоміших проектів на нашому ринку. Суть цього проекту полягає в тому щоб покращити психоемоційний стан людини за підтримкою Міністерства освіти і науки України.

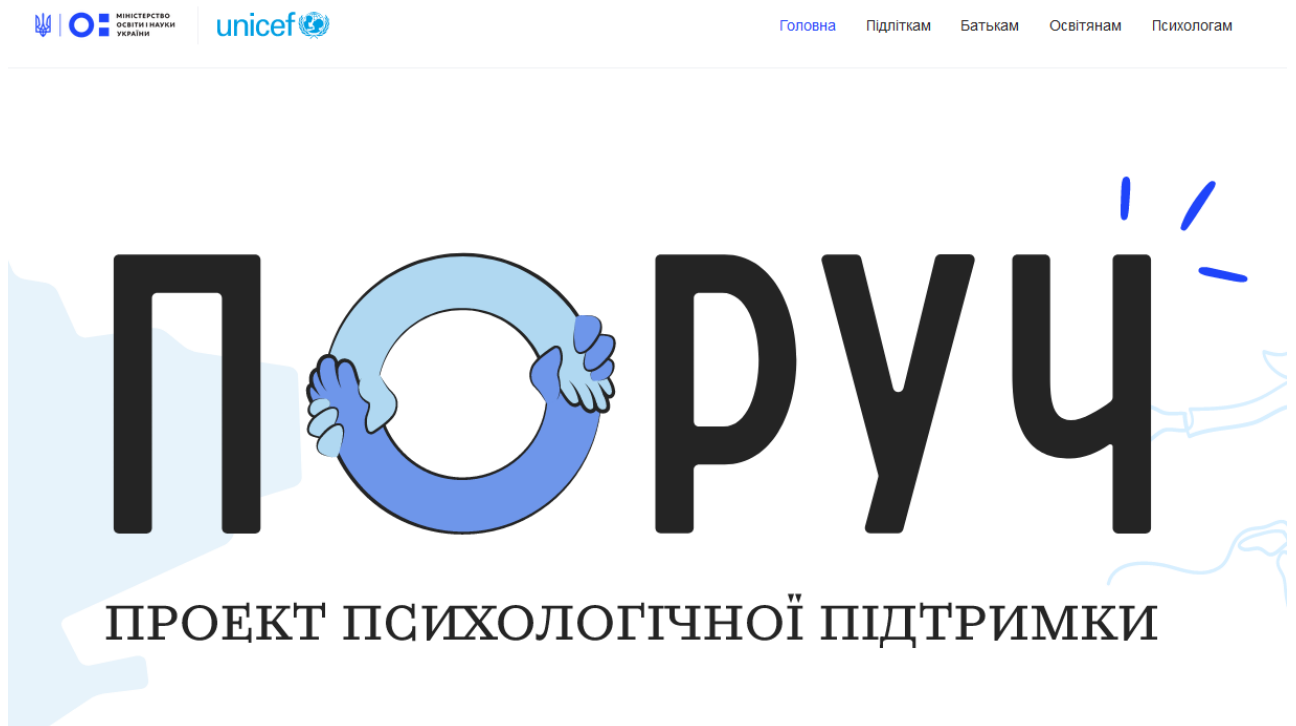


Рис. 1.12 Знімок екрану головної сторінки проекту «Поруч»

При першому перегляді головної сторінки цього проекту можна помітити

одразу зручний поділ на декілька так званих «типів підтримки», серед яких наступні пункти:

- 1) **Вид підтримки «Підліткам»** - цей розділ поділений для підлітків, яким важко пережити сучасні реалії війни та інші розлади. На самій сторінці зазначено наступне: «Нам усім важко переживати війну, але тобі може бути найважче. Тому ЮНІСЕФ запускає підтримку для тебе». Заняття відбуваються по 60 хвилин та поділена на групи, що має зручність у порівнянні з використаним часом на кількість людей, але може бути не достатньо практичною для цієї верстви населення. [9]

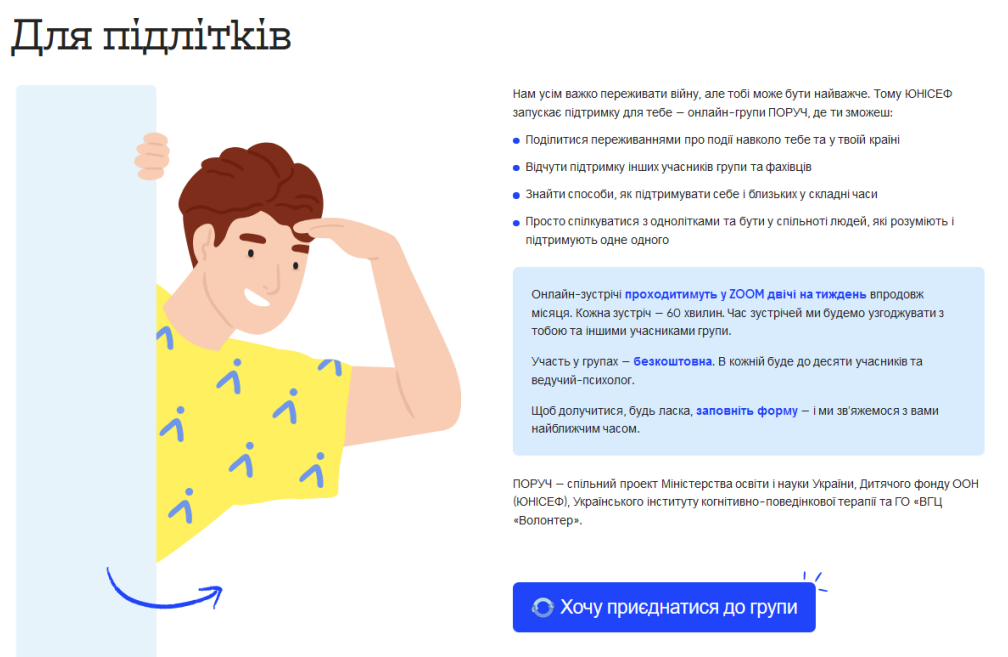


Рис. 1.13 Знімок екрану для підтримки підлітків

- 2) **Вид підтримки «Батькам»** - цей вид підтримки направлений на більшу частину верств населення. Ця група людей є вже повністю сформованими, але вона також схилена до розладів у психіці. Постійні переживання через роботу, колектив, проблеми з сім'єю та інші проблеми були присутні усе життя, але до цього додалися розлади через бойові дії, війну, тощо. Саме про вирішення цих проблем і було створено даний проект. [10]

## Для батьків

### Шановні батьки!

ЮНІСЕФ запускає групи підтримки для вас – якщо ви опинилися у складних обставинах, потребуєте психологічної підтримки або хотіли би спілкуватися з іншими батьками й фахівцями про те, що відбувається сьогодні у нашій країні, з нами, нашими дітьми та нашими родинами.

Ці групи – можливість обмінятися інформацією, дізнатися щось нове про себе та свої емоції, отримати фахові поради, як підтримати себе та своїх дітей і рідних.

Зустрічі проходять онлайн у Zoom двічі на тиждень впродовж місяця. Кожна зустріч – 60 хвилин. Час зустрічей ми будемо узгоджувати з вами та іншими учасниками групи.

Участь у групах – **безкоштовна**. В кожній буде до десяти учасників та ведучий-психолог.

Щоб долучитися, будь ласка, **заповніть форму** – і ми зв'яжемося з вами найближчим часом.

### Тематика зустрічей:

- Як підтримати себе та своїх близьких
- Як керувати емоціями у складні часи
- Як втамувати біль і знайти себе в нових життєвих умовах
- Як планувати у часи невизначеності
- Де знайти ресурси для цього

ПОРУЧ – спільний проект Міністерства освіти і науки України, Дитячого фонду ООН (ЮНІСЕФ), Українського інституту когнітивно-поведінкової терапії та ГО «ВГЦ «Волонтер».



Рис. 1.14 Знімок екрану для підтримки батьків

3) Вид підтримки «Освітнянам» - цей розділ співвідноситься до наступного, але основною відмінністю цього напряму є покращення себе у сфері психології;

## Для освітян

### Шановні освітяни!

ЮНІСЕФ запускає групи підтримки для вас – якщо ви потребуєте психологічної підтримки або хотіли би спілкуватися з іншими колегами та фахівцями-психологами про те, що відбувається сьогодні у нашій країні, з усіма нами та з нашими учнями і як допомогти впоратись собі та учням.

Ці групи – можливість обмінятися інформацією, дізнатися щось нове як про емоції дітей, які живуть у ситуації війни, так і про себе та свої емоції, отримати фахові поради, як підтримати себе та своїх учнів.

Зустрічі проходять онлайн у Zoom двічі на тиждень впродовж місяця. Кожна зустріч – 60 хвилин. Час зустрічей ми будемо узгоджувати з вами та іншими учасниками групи.

Участь у групах – **безкоштовна**. В кожній буде до десяти учасників та ведучий-психолог.

Щоб долучитися, будь ласка, **заповніть форму** – і ми зв'яжемося з вами найближчим часом.

### Тематика зустрічей:

- Як підтримати себе та підтримати дітей
- Як керувати емоціями у складні часи
- Як реагувати на поведінку дітей екологічно
- Де знайти ресурси для цього

ПОРУЧ – спільний проект Міністерства освіти і науки України, Дитячого фонду ООН (ЮНІСЕФ), Українського інституту когнітивно-поведінкової терапії та ГО «ВГЦ «Волонтер».



Рис. 1.15 Знімок екрану для освітян

4) Вид підтримки «Психологам» - цей вид скоріше про надання допомоги, а ніж навпаки, але з деякими нюансами. Якщо зважати на деякі інші проекти, які зазначені вище, де психологом може виступати будь-хто що має інтерес та освіт, то цей проект ставить інші рамки. У цій системі це психолог який буде скоріш нанятий та отримувати зарплату від проекту, тому місця є обмежені, але підтримка має бути краще, оскільки працюють спеціалісти. [12]

## Для психологів

### Шановні колеги!

ЮНІСЕФ відкриває набір психологів, які готові долучитися до проведення онлайн-груп підтримки батьків та підлітків, котрі пережили бойові дії, втрату, травму, у проекті ПОРУЧ.

### Вимоги до психологів:

- Вища психологічна освіта
- Досвід роботи з дітьми та батьками
- Досвід роботи з дітьми та дорослими, які пережили бойові дії, втрату, травму, є перевагою

Відібрані психологи будуть проходити навчальні вебінари, супервізії та інтерв'ї щодо роботи з батьками та підлітками в означеній тематиці.

Базова програма для проведення груп підтримки буде надана організаторами, проте може корегуватися психологами з урахуванням їхнього досвіду. Організатори також забезпечуватимуть робочі ZOOM-майданчики. Робота психологів **оплачуватиметься**.

Щоб долучитися, будь ласка, **заповніть форму** – і ми зв'яжемося з вами найближчим часом.

ПОРУЧ – спільний проект Міністерства освіти і науки України, Дитячого фонду ООН (ЮНІСЕФ), Українського інституту когнітивно-поведінкової терапії та ГО «ВГЦ «Волонтер».

 Хочу приєднатися до команди

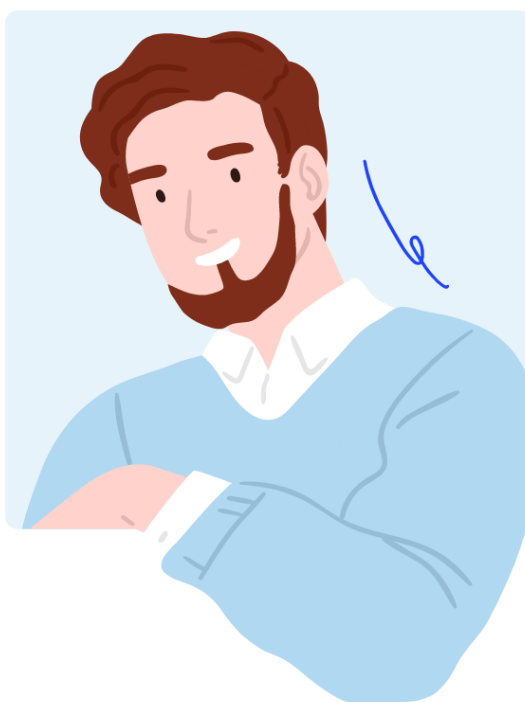


Рис. 1.16 Знімок екрану для психологів

При реєстрації для одного з цих видів підтримки користувача надсилає до його нового вікна у браузері з гугл-формою для заповнення. При заповненні форми з будь-якого типу психологічної підтримки, кожен заповнює свою форму, яка має специфічні запитання, приємно те що окрім звичайних питань по типу віку чи статі, можна відповісти на питання, які зменшать в подальшому затрати часу обох сторін. Хоча деякі з них мають нюанси, наприклад оскільки підвид «психолог», як описувалось вище, має обмежену кількість людей, тому часто можна отримати

наступне повідомлення при заповненні форми (Див. Рис. 1.17)

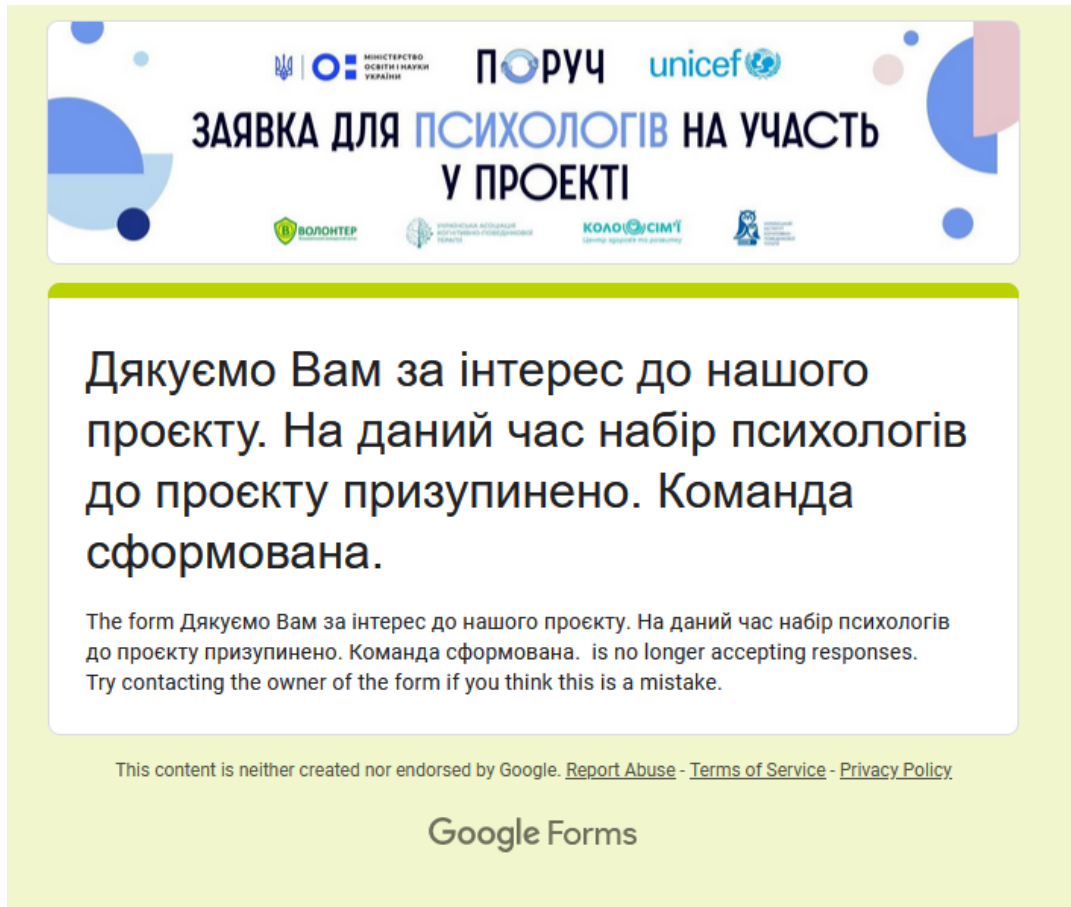


Рис. 1.17 Змінок екрану при заповненні форми як психолог

До того ж треба зазначити що, використання стороннього ресурсу для інтеграції у свій проект має практичність на зменшенні затрат часу на розробки своєї системи, але є мінусом що даний сайт не пропонує користувачу переглянути потрібну інформацію у власному кабінеті.

Серед позитивних рішень цієї системи я виділив наступне:

- Проект є безкоштовним для користувачів;
- Проект наймає професійних психологів;
- Зручні та обширні форми для заповнення;
- Розподіл на категорії та гнучкий графік.
- Проект є за підтримкою держави.

А негативними є наступні:

- Не можна переглянути персонал не зайшовши до системи;
- Обмежена кількість психологів;
- Немає кабінету для об'єднання потрібної інформації в одне місце;
- Використання сторонніх сервісів.

## **Висновки**

Головна сутність людини до покращення свого життя та тенденції до автоматизації всього навколо неї призвела до швидкої цифровізації всіх продуктів, які так чи інакше використовує людина.

Оскільки ми живемо в час цифровізації ця тематика вважається актуальною на декілька десятків років, а то і більше. І зважаючи на те що кожна людина так чи інакше пов'язана з психологією, це може бути як допомога близьким чи психологічна допомога собі, цифровізація цієї системи як ніколи є актуальною.

Створення такого програмного засобу зміг би допомогти людям витримати ці складні часи для більшості людей. Актуальність реалізації цього проекту полягає в тому що не зважаючи на ситуацію в світі люди в будь-якому разі знайдуть причину для використання цього продукту. Тобто головною ціллю створення такого програмного засобу є розробка сервісу для покращення психологічного стану, що буде актуальним завжди.

До того ж мною було проведено великий аналіз спектру ринку та виділено основні проблеми аналогів для їх подальшого виправлення.

## РОЗДІЛ 2

### МЕТОДИКА ВЗАЄМОДІЇ ЛІКАРЯ-ПСИХОЛОГА З КЛІЄНТАМИ

#### 2.1 Сутність та основні етапи розвитку психологічної підтримки

Психологічна підтримка є життєво важливим аспектом забезпечення психічного та емоційного благополуччя, що охоплює різні етапи його розвитку. Суть психологічної підтримки полягає у забезпеченні людей необхідними інструментами для подолання викликів, підвищення життєстійкості та зміцнення психічного здоров'я. У цьому есеї досліджуються основні етапи її розвитку, простежується еволюція психологічної підтримки в часі.

Витоки психологічної підтримки можна простежити в ранніх терапевтичних практиках, коли люди шукали розради через релігійні чи філософські настанови. Стародавні цивілізації часто покладалися на духовних лідерів або мудрих людей, які давали поради у важкі часи. Ці ранні форми підтримки заклали основу для розуміння взаємозв'язку психічного та емоційного здоров'я. [6]

Поява психології як окремої галузі наприкінці 19 століття стала значною віхою в розвитку психологічної підтримки. Першопрохідці, такі як Зигмунд Фрейд і Карл Юнг, заглиблювалися в тонкощі людського розуму, впроваджуючи терапевтичні техніки і теорії, які лягли в основу сучасного консультування і психотерапії. Цей період підкреслив важливість дослідження несвідомого та вирішення невирішених конфліктів для сприяння психологічному благополуччю.

У середині 20-го століття відбулася зміна парадигми з появою гуманістичної психології, яку очолили такі діячі, як Абрахам Маслоу та Карл Роджерс. Цей підхід наголошував на закладеному в людині потенціалі для зростання та самоактуалізації. Гуманістична психологія заклала основу для клієнтоцентрованої терапії, надаючи пріоритет емпатії, активному слуханню та безумовному позитивному ставленню як важливим компонентам психологічної підтримки. [7]

Друга половина 20-го століття принесла сплеск досліджень різних терапевтичних модальностей і науково обґрунтованих практик. Когнітивно-поведінкова терапія (КПТ) набула популярності завдяки своїй спрямованості на



зміну дисфункціональних моделей мислення та поведінки, надаючи людям практичні інструменти для управління своїм психічним здоров'ям. Ця епоха також стала свідком інтеграції різних терапевтичних підходів, що сприяло більш цілісному розумінню психологічної підтримки.

У наш час цифрова епоха трансформувала надання психологічної підтримки. Платформи онлайн-консультування, мобільні додатки та телетерапія стають дедалі популярнішими, розширюючи доступ до ресурсів у сфері психічного здоров'я. Технології сприяють негайній підтримці та втручанню, долаючи географічні бар'єри та охоплюючи людей, які інакше могли б зіткнутися з труднощами при зверненні за допомогою.

Крім того, зростає визнання важливості профілактичних заходів і дестигматизації проблем психічного здоров'я. Кампанії з підвищення обізнаності громадськості, оздоровчі програми на робочому місці та освітні ініціативи спрямовані на підвищення грамотності з питань психічного здоров'я та заохочення раннього втручання. Такий проактивний підхід є важливим етапом в еволюції психологічної підтримки, підкреслюючи важливість створення сприятливого середовища та виховання життєстійкості з раннього віку.

Отже, суть психологічної підтримки полягає в її спрямованості на покращення психічного та емоційного благополуччя. Еволюція психологічної підтримки проходила через історичні, теоретичні та технологічні ландшафти, адаптуючись до мінливих потреб людей і суспільства. Від стародавньої мудрості до сучасних терапевтичних методів, розвиток психологічної підтримки відображає постійне прагнення до розуміння і розвитку людської психіки.

### **2.1.1 Важливість дослідження та аспекти тематики психологічної підтримки**

Вивчення психології має величезне значення для розуміння людської поведінки, пізнання та емоцій. Як багатогранна галузь, вона охоплює різні аспекти психічного здоров'я, розвитку та благополуччя. У цьому есе ми заглибимося у

важливість вивчення психології та розглянемо ключові аспекти тем психологічної підтримки.

Психологія забезпечує основу для розуміння складних процесів мислення та поведінки людини. Досліджуючи когнітивні функції, соціальні взаємодії та емоційні реакції, психологи сприяють глибшому розумінню того, що керує людськими вчинками. Ці знання мають вирішальне значення в різних сферах, від освіти до бізнесу, оскільки вони дозволяють краще приймати рішення, ефективніше спілкуватися та покращувати міжособистісні стосунки. [8]

Одним із ключових аспектів психологічних досліджень є психічне здоров'я. Поширеність розладів психічного здоров'я підкреслює важливість дослідження та вирішення проблем психологічного благополуччя. За допомогою емпіричних досліджень психологи визначають фактори ризику, розробляють стратегії профілактики та вдосконалюють втручання для полегшення проблем психічного здоров'я. Ці знання допомагають формувати суспільство, яке надає пріоритет психічному здоров'ю, зменшувати стигму та сприяти ранньому втручанню.

Крім того, вивчення психології необхідне для розуміння розвитку людини протягом усього життя. Від дитинства до старості люди зазнають психологічних змін, які впливають на їхню поведінку та сприйняття. Розуміння цих траєкторій розвитку дозволяє педагогам, батькам і політикам створювати сприятливе середовище, яке відповідає потребам людей на різних етапах життя.

У сфері психологічної підтримки на увагу заслуговують різні теми. Консультування і терапія відіграють ключову роль у наданні допомоги людям, які стикаються з такими проблемами, як стрес, тривога і травма. Вивчення терапевтичних технік, методологій консультування та ефективності втручань сприяє вдосконаленню практик охорони психічного здоров'я.

Іншим важливим аспектом є вивчення соціальної психології, яка заглиблюється у вплив соціальних впливів на індивідуальну поведінку. Такі теми, як упередження, конформізм і групова динаміка, є невід'ємними для розуміння суспільної динаміки та сприяння інклюзивності. Дослідження в галузі соціальної

психології лежать в основі інтервенцій, спрямованих на зменшення дискримінації та формування більш справедливого і співчутливого суспільства.

Отже, важливість вивчення психології полягає в її здатності розгадати складнощі людської поведінки та сприяти індивідуальному і суспільному благополуччю. Теми психологічної підтримки - від втручань у сферу психічного здоров'я до соціальної динаміки - стосуються нагальних питань, які впливають на якість життя окремих людей і спільнот. Розбираючись у хитросплетіннях людського розуму, психологія залишається безцінним інструментом у сприянні розумінню, співпереживанню та позитивним змінам.

### **2.1.2 Ціль та мета психологічної допомоги людині**

Психологічна допомога відіграє важливу роль у підтримці людей, які стикаються з різними життєвими проблемами. Вона передбачає застосування терапевтичних методів, консультування та підтримку осіб, які зазнають психологічних труднощів. Кінцевим результатом є зміцнення психічного здоров'я та сприяння особистісному зростанню, що дозволяє людям жити повноцінним життям.

Основна мета психологічної допомоги полягає у вирішенні та полегшенні емоційного дистресу. Незалежно від того, чи є це наслідком травми, стресу або інших життєвих подій, психологи працюють над тим, щоб допомогти людям впоратися зі своїми емоціями, сприяючи підвищенню життєстійкості та емоційному благополуччю.

Допомога психолога часто відіграє вирішальну роль у кризових ситуаціях. Психологи надають негайну підтримку особам, які стикаються з гострими психічними кризами, допомагаючи їм стабілізуватися та орієнтуватися в складних обставинах.

Окрім кризового втручання, психологічна допомога є життєво важливою для постійної підтримки психічного здоров'я. Регулярні сесії з психологом можуть сприяти запобіганню ескалації проблем психічного здоров'я та сприяти проактивному підходу до благополуччя.

Психологи співпрацюють з людьми для розробки ефективних стратегій подолання стресу. Ці стратегії допомагають людям справлятися зі стресом, тривогою та іншими проблемами, підвищуючи стійкість і адаптивність перед обличчям життєвої невизначеності.

Ще одна важлива мета — спрямувати людину на шлях самопізнання та особистісного зростання. Завдяки самоаналізу та терапевтичному дослідженню люди можуть отримати уявлення про свої думки, поведінку та емоції, що призведе до позитивного особистісного розвитку.

Спектр дії психологічних сеансів також поширюється на покращення міжособистісних стосунків. Звертаючись до моделей спілкування, вирішення конфліктів і міжособистісної динаміки, психологи допомагають людям будувати здоровіші зв'язки з іншими, як в особистому, так і в професійному плані.

Психологічна допомога сприяє дестигматизації проблем психічного здоров'я. Забезпечуючи сприятливе і конфіденційне середовище, психологи відіграють ключову роль, допомагаючи людям орієнтуватися в складнощах їхнього внутрішнього світу, що в кінцевому підсумку сприяє більш здоровому і повноцінному життю.

### **2.1.3 Сутність проблематики у галузі психологічної допомоги та шляхи вирішення**

В сучасному світі свідомість про важливість психічного здоров'я наростає, але разом з цим збільшується коло викликів, які ускладнюють надання ефективної допомоги.

Перш за все, зберігається тенденція нерівного доступу до психологічних послуг, це є серйозною проблемою, а особливо в сільських районах та країнах із обмеженими ресурсами. Для вирішення такого виклику рішенням може бути розвиток телемедицини та програм психологічної допомоги для покращення доступності.

Існує проблема стійкої стигми, пов'язаної з сеансами психотерапії, яка заважає багатьом людям звертатися за психологічною допомогою. Можливий осуд,

розглядання похід до психолога як щось ганебне, заважає людям вчасно отримувати належну підтримку, що призводить до погіршення психоемоційного стану людини. В даному випадку, активна робота із зменшенням стигми через освітні проекти та взаємодію із ЗМІ може створити середовище, де кожен відчувається вільно звертатися за допомогою. [24, 25]

Також можна помітити, що з розвитком прогресу, зокрема, посилення впливу соціальних мереж на життя людей, різноманіття потреб в психологічній допомозі не завжди відповідає наявним ресурсам. Але постійне оновлення психологічних програм та розширення фахівців може полегшити справу. Використання технологій у психологічних послугах допомагає розширити їх обсяг, але вносить етичні виклики. Розробка стандартів безпеки даних та етичних принципів використання технологій є критичним завданням.

Попри ці виклики, постійна робота над вирішенням проблем галузі психологічної підтримки веде до створення більш доступних, ефективних та культурно чутливих послуг, що покликані полегшити тягар психічних проблем у суспільстві.

#### **2.1.4 Актуальні проблеми у галузі психологічної підтримки в Україні та їх вирішення**

Зараз активно пропагується самопізнання, різноманітні психологічні практики, курси для укріплення ментального здоров'я. Людям дійсно цікаво досліджувати себе, тому на зараз ми можемо спостерігати позитивну тенденцію на поширення цієї галузі, але на жаль так і залишається багато стереотипів.

Для ефективного рішення завжди потрібно знаходити корінь проблеми. У статистиці, яка була наведена на 2022 рік, а саме у цитаті з доповіді про психологічну допомогу говориться: «Переважна більшість опитаних (88%) не зверталися за психологічною допомогою впродовж останніх 6 місяців, з них тільки 27% зазначають про наявність наміру це зробити коли-небудь, тоді як 61% - не планують цього. Так, за допомогою зверталися 5% опитаних (частіше це жінки та

люди 18-24 років). Загалом намір звернутися за психологічною допомогою зменшується з віком» [8]

Варто відмітити, що перш за все в Україні немає чіткого законодавства, яке регулювало б психотерапевтичну практику. Це може призводити до відсутності стандартів, сертифікації та контролю за якістю послуг. Тому перш за все потрібно врегулювати це питання на законодавчому рівні.

Система психотерапії часто стикається з недостатнім фінансуванням. Це в свою чергу зменшує рівень доступності послуг для широкого населення та ускладнює професійний розвиток психотерапевтів. Для підвищення кваліфікації потрібно використовувати можливість співпраці, заохочувати спеціалістів приймати участь у міжнародних форумах, тренінгах та обмінюватися досвідом з іншими країнами, що дозволить запроваджувати використання передових практик та методик.

## **2.2 Сучасні психологічні виклики, з якими стикається населення**

У швидкоплинному та мінливому ландшафті сучасного світу люди стикаються з безліччю психологічних викликів, які суттєво впливають на їхнє психічне благополуччя. Ці виклики є різноманітними і зумовлені соціальними, економічними, технологічними та культурними змінами. Розуміння та вирішення цих сучасних психологічних проблем має вирішальне значення для формування більш здорового та життєздатного населення. [8]

Однією з найважливіших проблем є зростання рівня стресу і тривожності. Вимоги сучасного життя, що характеризується постійним зв'язком, тиском на роботі та суспільними очікуваннями, сприяють підвищенню рівня стресу. За даними Всесвітньої Організації Здоров'я, стрес на робочому місці став однією з ключових проблем сучасного життя. Близько 264 мільйони людей у всьому світі переживають депресію, пов'язану зі стресом на роботі. Постійна зайнятість часто призводить до вигорання, що впливає на психічний та емоційний стан людини.

Інциденти депресії та самогубства зростають у багатьох країнах. Світова Організація Здоров'я повідомляє, що депресія є найпоширенішим розладом, а кожен 40-й чоловік на планеті має ризик вчинити самогубство.

Повсюдне використання технологій є ще одним ключовим фактором, що формує сучасні психологічні виклики. Хоча технології принесли безпрецедентну зручність і зв'язок, вони також породили такі проблеми, як цифрова залежність, соціальна ізоляція та негативний вплив соціальних мереж на самооцінку. Досягнення балансу між перевагами технологій та їхніми потенційними негативними наслідками становить значну проблему.

Зміна характеру роботи та економічна невизначеність сприяють підвищенню стресу, пов'язаного з роботою, та незадоволеності роботою. Економіка гігантських проєктів, віддалена робота та постійна потреба у підвищенні кваліфікації створюють середовище, в якому людям важко знайти стабільність та реалізацію у своєму професійному житті, що впливає на їхнє психічне здоров'я.

Соціальна ізоляція та ерозія суспільних зв'язків є додатковими психологічними викликами. Сучасний спосіб життя часто надає перевагу індивідуалізму, що призводить до зменшення особистого спілкування та відчуття роз'єднаності. Самотність і відсутність систем соціальної підтримки можуть мати глибокі наслідки для психічного здоров'я. [26]

Екологічні проблеми та глобальні кризи, такі як зміна клімату та триваюча пандемія, сприяють виникненню почуття колективної тривоги. За даними світових організацій здоров'я, анксіетет та побоювання про здоров'я стали поширеними серед сучасного населення, особливо під час глобальних криз, таких як пандемія COVID-19. Невпевненість у майбутньому в поєднанні з усвідомленням глобальних викликів додає додатковий рівень психологічного стресу, з яким люди повинні справлятися.

Отже, сучасні психологічні проблеми, з якими стикається населення, є складними і взаємопов'язаними. Вирішення цих проблем потребує спільних зусиль з боку окремих осіб, громад, робочих місць і політиків. Розвиваючи культуру взаєморозуміння, надаючи доступні ресурси психічного здоров'я та усуваючи

першопричини стресу і тривоги, суспільство може працювати над створенням більш психологічно стійкого і процвітаючого населення.

### **2.2.1 Вплив та наслідки війни на психоемоційний фон людини**

Війна, з її руйнівною силою та людськими жертвами, залишає незгладимі сліди на психіці тих, хто переживає її тяготи. Наслідки війни виходять далеко за межі фізичної сфери, глибоко впливаючи на психоемоційний фон людей.

Досвід війни часто викликає низку сильних емоцій, включаючи страх, тривогу і глибоке горе. Постійна загроза життю, свідчення насильства та втрата близьких створюють психологічний тягар, який залишається ще довго після завершення конфлікту.

За інформацією від Всесвітньої організації охорони здоров'я (ВООЗ) заявляє, що в умовах будь-якого збройного конфлікту «серед людей які переживали бойові дії близько 10% людей, які пережили травматичні події, матимуть серйозні проблеми з психічним здоров'ям, а ще у 10% розвинеться поведінка, яка перешкоджатиме їхній здатності ефективно функціонувати". Депресія, тривога і психосоматичні проблеми, такі як безсоння, є найпоширенішими наслідками. Сагден зосереджується на трьох групах населення, схильних до негативних наслідків для психічного здоров'я:

- цивільне населення на батьківщині мішені;
- солдати по обидва боки конфлікту;
- ті, хто споживає зображення, відео- та аудіоматеріали про війну через соціальні мережі, телебачення, радіо та Інтернет.

"Дивно, але цивільне населення в зоні конфлікту, як правило, є групою з найменшою кількістю психологічних травм - але вони все одно можуть бути значними", - каже Сагден. Менша травматизація може бути результатом того, що цивільне населення має можливість негайно поговорити зі своєю соціальною мережею і обробити свої емоції, що допомагає зміцнити стійкість. [24]



Довгострокові наслідки травми для солдатів є значними. "Ми спостерігаємо зростання кількості бездомних серед ветеранів США, і серед цієї групи найвищий рівень самогубств порівняно з будь-якою іншою групою населення", - каже Сагден. Солдати в усьому світі схильні до травматичних подій, а з травматичним впливом пов'язано більше загальних медичних ускладнень, дисфункції в сім'ях, безробіття, вживання психоактивних речовин тощо.

Дослідження показали, що споживачі інформації про війну через телебачення, соціальні мережі або інші форми медіа можуть зазнавати такого ж впливу, як і безпосередні учасники конфлікту.

Посттравматичний стресовий розлад (ПТСР) є поширеним наслідком, що проявляється у вигляді нав'язливих спогадів, нічних кошмарів і підвищеної пильності.

Одним із довготривалих наслідків війни є розрив соціальних і сімейних зв'язків. Розлука з членами сім'ї, переміщення та втрата громадських структур сприяють виникненню почуття ізоляції. Руйнування мереж соціальної підтримки загострює проблеми психічного здоров'я, залишаючи людей без звичних буферів, які сприяють психологічній стійкості.

Діти, які є особливо вразливими під час конфлікту, часто страждають від довготривалих психологічних шрамів. Вплив насильства та порушення нормальної траєкторії їхнього розвитку може призвести до довготривалих емоційних труднощів. Перерви в навчанні, втрата почуття безпеки та відсутність стабільного середовища, де про них піклуються, сприяють виникненню психологічних проблем, які можуть залишитися і в дорослому віці.

Депресія є ще одним поширеним наслідком війни, що виникає внаслідок кумулятивного впливу травм, втрат і переміщення. Економічні наслідки конфлікту, включаючи бідність і безробіття, ще більше посилюють ризик розладів психічного здоров'я. Боротьба за відновлення життя на тлі зруйнованої інфраструктури часто породжує відчуття безнадійності, що сприяє збереженню депресивних симптомів.

Крім того, нормалізація насильства під час війни може змінити суспільні цінності та норми, впливаючи на морально-етичні рамки людей. Моральна травма,

що характеризується почуттям провини або сорому, пов'язаним з морально неоднозначними діями або звірствами, свідками яких стали, додає додатковий шар до психологічної травми, яку переживають ті, хто бере участь у конфлікті або постраждав від нього.

Отже, вплив і наслідки війни на психоемоційний фон людей є глибокими і довготривалими. Визнання складності постконфліктних проблем психічного здоров'я має вирішальне значення для розробки ефективних інтервенцій, спрямованих на задоволення різноманітних потреб постраждалих осіб. Визначаючи пріоритетність підтримки психічного здоров'я в постконфліктних ситуаціях, суспільство може працювати над загоєнням психологічних ран, завданих війною, і розвитком стійкості перед лицем несприятливих обставин.

### **2.2.2 Найкращі практики у виявленні та вирішенні психологічних проблем населення**

У складній картині суспільного благополуччя виявлення та вирішення психологічних проблем серед населення є першочерговим завданням. Впровадження найкращих практик у цій сфері має вирішальне значення для зміцнення психічного здоров'я, життєстійкості та загального суспільного процвітання. У цьому есе досліджуються ефективні стратегії та підходи до розпізнавання та вирішення психологічних проблем, підкреслюючи важливість цілісного підходу, орієнтованого на громаду.

Для ідентифікації психологічних проблем активно використовується впровадження програм раннього втручання, особливо в освітніх закладах, дозволяє своєчасно виявляти нові психологічні проблеми в окремих осіб, особливо серед дітей та підлітків. Шкільні психологи та консультанти відіграють ключову роль у розпізнаванні ранніх ознак та наданні необхідної підтримки. [27]

Важливим є створення інформаційно-просвітницьких програм у громаді, які сприяють підвищенню обізнаності про психічне здоров'я та дестигматизації звернень за допомогою. Це передбачає проведення освітніх кампаній, семінарів та ініціатив, спрямованих на створення середовища, в якому люди почуваються

комфортно, обговорюючи та вирішуючи питання свого психологічного благополуччя.

Використання стандартизованих інструментів скринінгу та оцінки допомагає фахівцям у різних сферах, таких як охорона здоров'я та робочі місця, виявляти осіб, які мають ризик виникнення психологічних проблем. Ці інструменти допомагають оцінити симптоми, оцінити фактори ризику та спрямувати відповідні втручання.

Вирішення цих сучасних психологічних проблем вимагає багатогранного підходу. Освітні та інформаційні кампанії можуть відігравати ключову роль у подоланні стигматизації психічного здоров'я та сприянні відкритим розмовам. Ініціативи на робочому місці, спрямовані на зміцнення психічного здоров'я, такі як програми управління стресом і консультаційні послуги, можуть допомогти працівникам впоратися з проблемами, пов'язаними з роботою.

Заохочення здорових стосунків з технологіями передбачає розвиток цифрової грамотності та усвідомленості. Балансування переваг технологій зі стратегіями пом'якшення їхніх негативних наслідків може сприяти більш гармонійній інтеграції технологій у повсякденне життя.

Крім того, ініціативи з розбудови громад та зусилля зі зміцнення соціальних зв'язків можуть пом'якшити наслідки соціальної ізоляції. Заохочення почуття приналежності та взаємозв'язку сприяє емоційній стійкості та створенню мереж підтримки.

Реінтеграція в постконфліктне суспільство ставить свої власні виклики. Шрами війни можуть бути не одразу помітними, що призводить до нерозуміння та співчуття з боку оточуючих. Стигматизація та дискримінація людей з проблемами психічного здоров'я ще більше ускладнюють їхню здатність відновлювати своє життя.

Зусилля, спрямовані на подолання психоемоційних наслідків війни, мають бути комплексними та багатовимірними. Заходи з охорони психічного здоров'я мають бути інтегровані в зусилля з постконфліктної відбудови, включаючи консультування, травмофокусовану терапію та системи підтримки на рівні громад.

Освітні програми, спрямовані на підвищення обізнаності та зменшення стигми, є важливими компонентами створення сприятливого середовища.

Впровадження інтегрованих моделей охорони здоров'я, які безперешкодно включають послуги з охорони психічного здоров'я в заклади первинної медичної допомоги, забезпечує комплексний підхід до благополуччя. Це дозволяє виявляти та вирішувати психологічні проблеми поряд із проблемами фізичного здоров'я.

Забезпечення доступності послуг з охорони психічного здоров'я є життєво важливим. Це включає в себе наявність доступних і культурно компетентних фахівців у сфері психічного здоров'я, а також інтеграцію можливостей телемедицини для охоплення людей у віддалених або недостатньо охоплених послугами районах.

Розробка ефективних стратегій кризового втручання, включаючи гарячі лінії, групи реагування на кризові ситуації та тренінги з надання першої допомоги у сфері психічного здоров'я, має вирішальне значення. Ці ініціативи забезпечують негайну підтримку під час гострих психологічних криз і допомагають запобігти подальшій ескалації проблем психічного здоров'я.

Насамкінець, впровадження найкращих практик у виявленні та вирішенні психологічних проблем вимагає багатогранного та спільного підходу. Інтегруючи раннє втручання, доступні послуги та стратегії на рівні громади, суспільство може створити середовище, яке сприятиме психічному здоров'ю, життєстійкості та загальному благополуччю населення.

### **2.3 Вимоги до застосунку з психологічної допомоги населенню**

При побудові будь якого застосунку важливо розуміти його сутність та реальні потреби в ньому, ідею, реалізацію та проектування часу. Саме це було проаналізовано. Оскільки, реалізація будь-якого проекту це в першу чергу бізнес, навіть тоді коли він є некомерційний, це в першу чергу про раціональне розподілення ресурсів. Хороший продукт, а за тим і бізнес – це в першу чергу той продукт на який витрачалосся достатньо часу та ресурсів перед його реалізацію, тому

для побудування найбільш якісного продукту на ринку було винесено проаналізувати наступні вимоги:

- Бізнес вимоги;
- Функціональні вимоги;
- Користувацькі вимоги;
- Системні вимоги;
- Нефункціональні вимоги.

Далі про ці вимоги та їх аналіз у даній галузі було проаналізовано про них детальніше.

### **2.3.1 Бізнес вимоги до програмного застосунку**

Для проектування будь-якого продукту важливо проаналізувати та визначити його бізнес-вимоги, саме про це буде йтись у цьому розділі.

Бізнес-вимоги – це процес аналізу та створені на їх основі комплексного рішення, яке буде представлено у виді документу, де визначені основні потреби специфічного проекту.

Таке проектування направлене на оптимізацію планування та реалізацію проекту. Строге визначення потреб та плану реалізації мінімізує та оптимізує затрачений час в процесі реалізації. Також варто зазначити, що при аналізі вимог визначається суть проекту та його зацікавленість та вже на цьому етапі можна заощадити велику кількість ресурсів, серед яких грошові ризики та людський час.

Після дослідження основних методики галузі психоемоційної підтримки, було сформовано наступні бізнес-вимоги до продукту:

- Якісний комфортний інтерфейс;
- Інтеграція потрібних методик до програмного продукту;
- Автоматизація рутинної роботи між пацієнтом та лікарем;
- Інтеграція та гнучкість між системами;
- Заохочення до проекту як пацієнтів, так і лікарів.

Сформовані бізнес-вимоги допоможуть у розумінні суті проекту, надобності та формуванні наступних кроків для проектування якісного продукту на ринку.

### **2.3.2 Функціональні вимоги до програмного застосунку**

Аналіз та проектування функціональні вимоги – це процес при якому досліджуються деталі і інструкції, що визначають основні функції програмного продукту, які будуть надані користувачам.

Зазвичай подібні вимоги проектуються спеціальним фахівцем або навіть групою людей, які проектуються функціональні вимоги до програмного продукту починаючи з початкових етапів розробки проект, так і протягом усього існування проекту. Щоб гарантувати надобність програмного продукту вимоги проектуються за спеціальними принципами: забезпечення має бути реалізоване якомога простіше у використанні та працездатним.

Функціональні вимоги можуть бути поділені за декількома напрямками, але головна ідея є у представленні найбільш відомих рішень для подальшого імплементування. Окрім того, вони можуть відрізнятися за поведінкою, можливостями та протоколами, залежно від галузі, в якій працює користувач. Наприклад, розробник відеоігор може використовувати різні функціональні вимоги до програмного забезпечення, які зосереджені на дизайні ігор, тоді як вчитель може використовувати функціональні вимоги, які зосереджені на зручності користування для учнів.

Основними функціональними вимогами до продукту з підтримки психоемоційного стану населення були виведені наступні:

- Функціонування створення власного кабінету;
- Функціональність власного кабінету;
- Можливість використання системи з обох сторін, як лікар, так і пацієнт;
- Функціональність з пошуку лікаря за фільтрами;
- Перегляд найчастіших запитань;
- Функціональна стрічка з цікавими порадами;

- Імплементация опитування та їх інтеграція.

Саме наведені вище функціональні вимоги до програмного застосунку допоможуть у строгій реалізації потрібного функціоналу для заохочення більшої кількості користувачів за рахунок комбінації основних задач в одному додатку. Оскільки, виходячи з проведеного аналізу ринку, було спроектовано найголовніші функціональні вимоги до застосунку для звичайних користувачів.

### **2.3.3 Користувацькі вимоги до програмного застосунку**

Опис специфікації вимог користувача містить у собі ціль – вирішення проблем потенційних клієнтів даного додатку та план реалізацій їх функціонування, саме таку ціль, першу чергу для себе ставить бізнес як продукт.

При написанні користувацьких специфікацій буде сформований один з головних документів для бізнесу, що зазначає у собі усі можливі вимоги користувача.

Специфікації користувацьких вимог – це один з найголовніших документів для планування, що визначає, як користувач хоче бачити систему як від початку проекту та і до останніх реалізацій проекту.

Бізнес звертається за цими специфікаціями від імені менеджера, СЕО проекту, інші головні посади, до команди з інженерів-розробника за профільною освітою. В основному з боку бізнесу надається професіонал з команди забезпечення якості, які разом з командою інженерів-розробників проектують систему, а саме - специфікації користувацьких вимог. Ця команда аналізує та впроваджує їх упродовж усього проекту.

Вихідним результатом взаємодії цієї команди є повністю сформований документ за наступними пунктами:

- Документ що був сформований це в першу чергу взаємодія двох команд;
- Документ не містить занадто багато технічних термінів та інформації;

- Результатом цього документу, має бути повністю сформований та зрозумілий список специфікацій користувацьких вимог;
- Документ має містити кінцеву точку зору на продукт з боку користувача;

Звертаючись до тематики проекту, продукт має на мені інтегрувати основні потреби двох типів користувачів: лікар та пацієнт. Важливо зазначити, один користувач може бути одночасно, як лікарем, для надання послуг, так і пацієнтом для отримання послуг від першого.

Однією з головних цілей проекту є автоматизація взаємодії та її інтеграція у систему між двома особами (лікар і пацієнт).

У цьому розділі було проаналізовано вимоги на основі проектування системи для лікарів та пацієнтів, тому вимоги були умовно поділені на два етапи:

Основні користувацькі вимоги до системи від лікаря:

- Можливість переключитись до системи «як пацієнт»;
- Функціонал перегляду поточних пацієнтів у системі;
- Зручний пошук по контактній інформації пацієнта;
- Функціонал взаємодії з опитуваннями у системі;
- Функціонал надання документів засвідчення;
- Оновлення профілю користувача системи;

Основні користувацькі вимоги до системи для пацієнта:

- Функціонал з пошуку лікарів у системі;
- Можливість переключитись до системи «як пацієнт»;
- Можливість мати декілька лікарів;
- Функціонал зручного перегляду контактів лікаря;
- Функціонал взаємодії з опитуваннями;
- Оновлення профілю користувача системи;



Також однією з найголовніших функцій є реалізація простого та інтуїтивно зрозумілого інтерфейсу для будь-якого користувача.

Після проведення обширного аналізу було спроектовано вище зазначені вимоги користувачам цього застосунка. Як зазначається вище, цей проект допоможе як лікарям, так і користувачам планувати та заощадити свій час, що є однією з головних цілей проекту.

### **2.3.4 Системні вимоги до програмного застосунку**

Системні вимоги — це опис конфігурації та вимог, які має мати система, щоб апаратне або програмне забезпечення працювало безперебійно та ефективно. Системні вимоги також відомі як мінімальні системні вимоги.

У раз невиконання системних вимог це може призвести до проблем із встановленням, що допоможе запобігти встановленню пристрою чи програми, чи продуктивністю системи., що може спричинити несправність продукту або роботу, нижчу від очікуваної, або навіть зависання чи збій.

Для реалізованих проектів, що вже запуснені на ринку, систем вимоги часто можна побачити в зазначених пунктах при покупці, якщо виконувати це онлайн, або вони можуть бути надруковані у спеціальному посібнику для користувача, який йде разом з продуктом.

В сучасності використовується перший варіант. Одним з прикладів системних вимог можуть бути наступні системні вимоги до Windows (Див. Рис. 2.1). Подібний підхід практикується часто, навіть можна сказати що це шаблон при проектуванні будь-якої системи, бо знаючи системні вимоги, розробник розуміє потреби клієнта та що він собою представляє. При цьому, команда розробників зможе краще побудувати систему і така практика часто використовується при розробці важкого програмного забезпечення

Як зазначається на цьому рисунку, кожен користувач повинен ознайомитися з цими параметрами, аби система працювала доречно, або користувач отримає негативний досвід взаємодії з системою що буде проявлятися в некоректній роботі системи: баги (або «bug»), глюки (або «lags»), інші труднощі з системою.

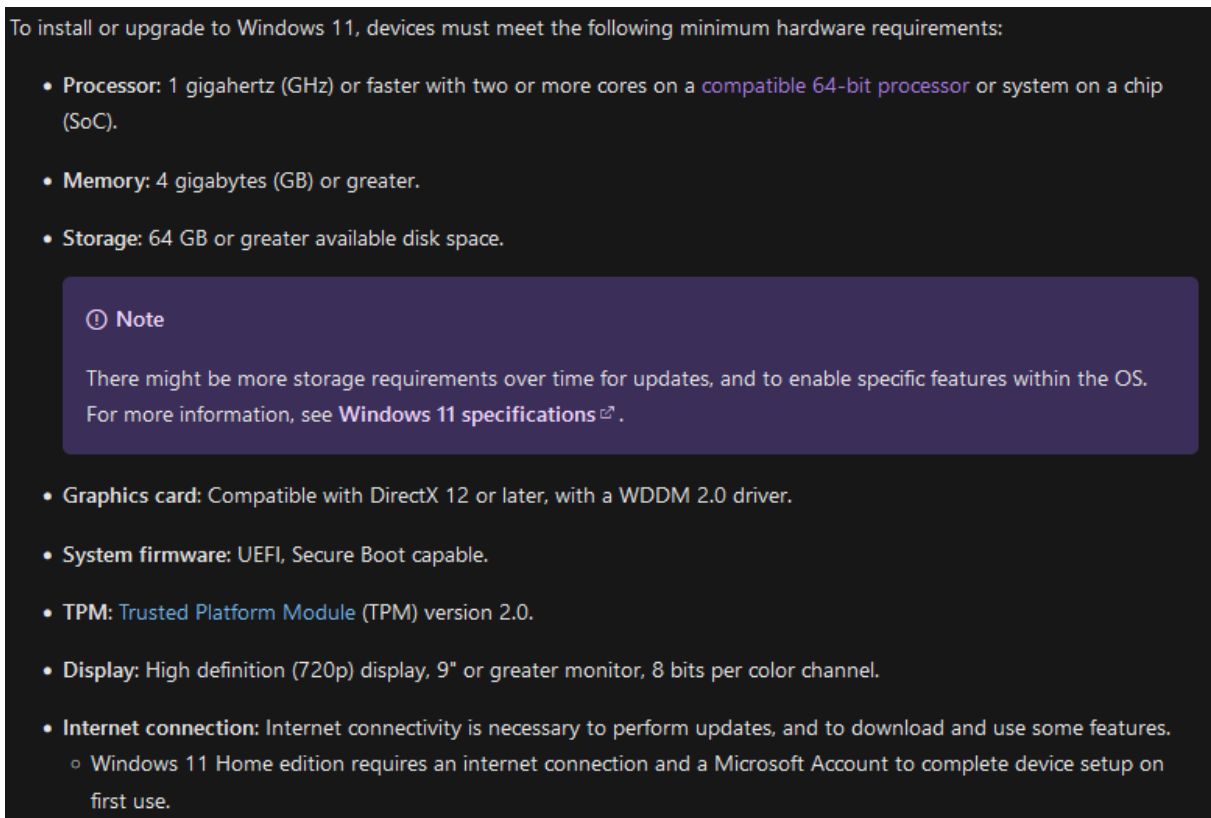


Рис. 2.1 Знімок екрану з прикладом системних вимог для системи «Windows 11»

Отже, як зазначалось вище, для продуктів що були вже реалізовані системні вимоги часто друкуються на упаковках, а для продуктів, які можна завантажити, системні вимоги часто вказуються на сторінці завантаження. Системні вимоги можна в цілому класифікувати як функціональні вимоги, вимоги до даних, вимоги до якості та обмеження. Вони часто надаються споживачам у повних деталях. Системні вимоги часто вказують на мінімальну та рекомендовану конфігурацію. Перше є найосновнішою вимогою, достатньою для встановлення або запуску продукту, але продуктивність не гарантовано буде оптимальною. Останній забезпечує безперебійну роботу. Системні вимоги до апаратного забезпечення часто визначають версію операційної системи, тип процесора, розмір пам'яті, доступний дисковий простір і додаткові периферійні пристрої, якщо такі є. Системні вимоги програмного забезпечення, на додаток до вищезазначених вимог, також можуть визначати додаткові залежності програмного забезпечення (наприклад, бібліотеки, версія драйвера, версія інфраструктури). Деякі виробники обладнання та

програмного забезпечення надають програму помічника з оновлення, яку користувачі можуть завантажити та запустити, щоб визначити, чи відповідає їхня система вимогам продукту.

Далі, будуть перераховані основні системні вимоги до системи:

- Гарне підключення до мережі Інтернет;
- Браузер що підтримує сучасні технології вебу (Google Chrome, MS Edge, Firefox, інші);

Оскільки програмний продукт не передбачає імплементацію на інші платформи окрім веб-застосунку, тому були наведені тільки основні системні вимоги. У разі, якщо б проектувалась на інші платформи типу «Windows», «iOS/Android» були б зазначені як мінімальні так і рекомендовані, з зазначенням операційної системи та конфігурації пристроїв.

### **2.3.5 Нефункціональні вимоги до програмного застосунку**

Нефункціональні вимоги є життєво важливими для успіху програмних систем. Якщо їх не враховувати належним чином, це може призвести до небажаних результатів, таких як незадоволені користувачі, розробники та клієнти, а також до перевищення графіку та бюджету на виправлення програмного забезпечення, яке було розроблене без урахування нефункціональних вимог.

Неузгодженість термінології, плутанина у визначеннях та відсутність загальноприйнятої схеми класифікації роблять розуміння нефункціональних вимог складним завданням. Отже, звернемось до визначення нефункціональних вимог:

Нефункціональні вимоги – це опис, наскільки добре може та функціонує програмна система. Вимоги спрямовані на створення відчуття системи. Більшість сучасних компаній на ринку, що використовують такі інструменти керування життєвим циклом програм, як Jira, Azure DevOps, інші, використовують нефункціональні вимоги. Найсучасніші компанії на ринку також звертаються до нових методологій, таких як Agile та BABOK, щоб адаптуватися до змін ринку.

Важливість нефункціональних вимог зумовлена тим, що вони можуть сприяти або перешкоджати успіху програмної системи або продукту у бізнесі.

Користувачі без вагань відмовляться від системи, навіть якщо вона відповідає всім очікуваним функціональним вимогам, але не може забезпечити необхідну якість результатів. Для бізнесу це не припустимо і може бути фатальним для конкретного проекту.

Нефункціональні вимоги слугують джерелом багатьох функціональних вимог, які мають величезне значення і повинні бути визначені з належним рівнем деталізації в документах вимог.

Крім того, вони є важливими вхідними даними для деяких ключових архітектурних рішень і допомагають гарантувати, що деякі ключові аспекти проектування будуть враховані з самого початку, а не на пізніших етапах процесу, коли вартість і ризик змін стають вищими.

Можна очікувати, що різні програмні системи будуть відповідати різним атрибутам якості залежно від їхньої природи, типу та розміру. Нижче наведено деякі загальні категорії нефункціональних вимог з їхніми описами:

- 1) **Доступність системи** - це ступінь працездатності рішення та доступності, коли це необхідно. Це міра часу, протягом якого система повністю працює, тобто доступна для використання та іноді включається як угода про рівень обслуговування, враховуючи її критичність для бізнесу.
- 2) **Сумісність з системою** - це ступінь сумісності рішення з іншими компонентами. Це показник того, наскільки ефективно система взаємодіє з іншими програмними системами та наскільки легко вона інтегрується із зовнішніми апаратними пристроями.
- 3) **Продуктивність системи** - це ступінь, до якого рішення або його компонент виконує свої функції з мінімальним споживанням ресурсів. Це надзвичайно важливий атрибут якості, оскільки він має серйозні наслідки для дизайну, а також впливає на вибір апаратного пристрою.

- 4) **Надійність системи** - це здатність рішення або його компонента виконувати свої необхідні функції без збоїв у заздалегідь визначених умовах протягом визначеного періоду часу.
- 5) **Безпека у системі** – це забезпечення системи у безпеці, головною метою є забезпечення рішення та компонентів захищеними від несанкціонованого доступу або атак зловмисного програмного забезпечення.
- 6) **Зручність використання** - це легкість, з якою користувач може навчитися використовувати рішення, і легкість, з якою він/вона може використовувати систему.
- 7) **Модернізація системи** - означає легкість, з якою рішення або його компонент можна виправити, покращити для задоволення потреб бізнесу чи підвищення ефективності, або адаптувати до мінливого середовища.
- 8) **Портативність системи** - це легкість, з якою рішення або його компонент можна перенести з одного операційного середовища в інше. З роками його важливість зросла, оскільки додатки повинні працювати в різних середовищах, таких як Windows, Mac і Linux, а також мобільних операційних системах, таких як Android і iOS.
- 9) **Масштабованість системи** - означає ступінь, до якого рішення здатне розвиватися, щоб виконувати збільшений обсяг роботи. Збільшений обсяг роботи може стосуватися бази користувачів, транзакцій, даних, мережевого трафіку чи інших факторів.

Звертаючись до реалій сьогоденного ринку та цього проекту було визначено основні нефункціональні вимоги до продукту. Оскільки вище було зазначено перелік загальних перелік нефункціональних вимог за ними було проаналізовано даний продукт.

Отже, сформований список нефункціональних вимог виглядає наступним чином:

- Система повинна мати можливість зручної авторизації та реєстрації у системі;
- Система має бути спроектована таким чином, щоб модифікація системи заощаджувала якомога менше часу;
- Система маж бути представлена у вигляді веб-застосунку для охоплення як можна більшої кількості користувачів;
- Користувач має отримати зручний та зрозумілий інтерфейс у використанні;
- Система має реалізувати основні ідеї цього програмного засобу;
- Система має надавати користувачам потрібний функціонал, що був зазначений вище.

На головній сторінці користувачу буде запропоновано увійти у систему, якщо користувач не має аккаунту йому буде запропоновано зареєструватися.

Після реєстрації чи авторизації користувача буде надіслано до системи, користувач матиме змогу обрати з двох видів діяльності: бути у системі «як лікар» чи «як пацієнт». В залежності від обраної події користувач зможе виконувати різний функціонал, так наприклад:

- Якщо користувач вибере «бути у системі як лікар», йому буде запропоновано переглянути пацієнтів, їх контакти, взаємодію з опитуваннями, оновити дані лікаря та інше;
- Як користувач вибере «бути у системі як пацієнт», йому буде запропоновано переглянути лікарів, їх контакти, переглянути конкретного лікаря, взаємодіяти з ним та його опитуваннями та інше.

Також система запропонує кожному користувачу зручний інтерфейс, переглянути цікаві посилання чи статті та переглянути найчастіші запитання для покращення стану.

## **Висновки**

Головна суть людини – це прогрес та еволюція людини. Отже протягом усього часу ця тематика буде актуальною на будь-якому етапі людства.

Ціль розділу полягала в тому щоб надати розуміння конкретної галузі та її проблематики, аби зазначити це в подальшому.

Оскільки, люди не є ідеальними то виникають безліч проблем, що були вже описані в цьому розділі. До них входить: людські вади, переживання, вплив війни та обмежень під час COVID-19 на людину.

Тому у цьому розділі були проаналізовані важливі для людства тематики, окрім того було досліджено найкращі методики та практику у галузі психоемоційного стану людини, вплив та вирішення проблем людини.

До того ж було описано основні вимоги до побудови продукту, на основі яких були побудовані наступні розділи.

## РОЗДІЛ 3

### СТРУКТУРА ІНФОРМАЦІЙНОЇ СИСТЕМИ

#### 3.1 Моделі сучасного проектування

Для проектування у сучасному світі існує безліч варіацій в архітектурних рішеннях. Оскільки проектування і побудова будь-якої системи в першу чергу заснована на зацікавленості бізнесу. Бізнес, як проект зацікавлений в декількох наступних тезисах:

- 1) Кожен бізнес, як проект, зацікавлений у покритті якомога більше користувачів;
- 2) Кожен бізнес, як проект, зацікавлений у побудові якісного та швидкого рішення;
- 3) Будь-який бізнес проект, має бути супроводжений командою від початку та до його кінця;
- 4) Будь-який бізнес зацікавлений реалізувати якомога кращу архітектуру проекту з затраченим якомога меншою кількістю ресурсів;
- 5) Бізнес хоче бути влучений та контролювати всі етапи розробки;
- 6) Великі бізнес проекти хочуть мати гнучку систему наскільки це тільки можливо;
- 7) Більш-менші проекти хочуть бути реалізовані у більш короткі терміни з меншою гнучкістю та затратами на ресурси.

Отже, проаналізуємо кожен тезис:

Звертаючись до першого тезису, мається на увазі, що будь який бізнес це проект, що має в планах охопити як можна більшу кількість користувачів. Говорячи більш детально, система має бути спроектована таким чином, щоб бути більш гнучкою для реалізацій майбутніх потреб бізнесу та користувачів. Окрім того, система має бути спроектована таким чином щоб містити у собі вже описані та



сформовані вимоги, як бізнесу так і користувачів. Саме слідуючи цим принципам, система має бути гарно спроектованою для усіх користувачів.

Опираючись на другий тезис, в першу чергу йдеться про побудову якомога якіснішого проекту з використанням меншої кількості ресурсів. Так, наприклад, бізнес-аналітик, команда з інженерів-програмістів, архітектор, тощо, це все і є одним з визначення слова «ресурс» у галузі бізнесу, але окрім цього це ще гроші, фінансування, тощо.

Наступним тезисом є зацікавленість будь-якого проекту від початку та до кінця. Авжеж будь-який бізнес буде зацікавлений у реалізації від самих перших початків бізнесу до реалізації більш пізніх ідей та аспектів. Так, наприклад, спочатку проект може бути реалізований на охоплення лише однієї групи населення, але з розвитком поширюватись і на інші, це все відбувається через вже описані основні тези бізнесу при проектуванні та буде розглянуто детальніше пізніше.

Подібна реалізація у сфері ІТ має дещо інший погляд. У цій сфері є свої рішення для проектування систем, прикладом може слугувати підтримка та розробка, далі більш детальніше:

- Говорячи про розробку, то зазвичай, бізнес наймає команду розробників, які разом з бізнесом, через свої організаційні питання, реалізують систему, в цьому випадку нічого нового;
- Говорячи про підтримку, то в цьому випадку з'являються нюанси, які описані вище. Оскільки система, як програмне забезпечення, має життєвий цикл разом з розвитком бізнесу, то визначити цей самий кінець є важким питанням. Тому при проектуванні систем, бізнес часто може звертатись до різних компаній для розробки та підтримки. Окрім того, після розробки, компанії зазвичай підписуються на підтримку системи ще якийсь час.

Наступний тезис є використання якомога меншу кількість ресурсів. Як було зазначено вище, бізнес зацікавлений у реалізації як можна більше функцій за як можна меншу кількість ресурсів, де ресурсами виступають команди з розробки,

бізнес-аналітиків та інші, окрім того до них відноситься також фінансування, проекту.

В наступному тезисі описується про влучення та контролювання всіх етапів розробки. Говорячи про проектування систему у бізнесі, було засновано безліч ідей та підходів, якими є Agile, SCRUM, інші. Описуючи коротко ці підходи, варто зазначити що однією з ідей є активним залучення бізнесу у проектування систем, планування та їх реалізацію. Системи відрізняються деякими підходами та нюансами, такими як представниками комунікації між бізнесом та командою розробки, але принципово системи виконують однакову головну суть, це залучення бізнесу у проектування системи.

Кожен бізнес зацікавлений у розростанні системи. Якщо проект був запущений успішно, то тільки питання часу, коли йому знадобиться розширення, саме цим буде займатися команда з підтримки проекту, яка описувалась в одному з тезисів. В даному випадку буде розглянутий тезис щодо гнучкості системи.

Оскільки будь-яка система, будь-якого масштабу проектується людьми та не може передбачити свої потреби досконало, питання проектування системи з якомога більш гнучкою системою завжди буде актуальною. Більшість бізнес-проектів проектується починаючи з малих вкладень, після чого розростаються. Так, наприклад, системи що є зараз найбільш відомими на початку свого реалізовували достатньо простий функціонал, після чого розширювались. Проте, варто зазначити що реалізація гарної та гучної системи є достатньо затратною ідеєю. Окрім того, будь-яка найрозумніша людина зможе лише спроектувати лише більш гнучкішу систему а ніж інші, але ніколи не зможе спроектувати ідеально, що варто враховувати при проектуванні бізнесу.

Останнім тезисом є висновок, який виходить з декількох інших тезисів та є доведений вище.

Отже, в цьому підрозділі було проаналізовано всю інформацію відносно проектування системи з точки зору бізнесу, оскільки будь-який проект це в першу чергу про бізнес, навіть тоді коли проект не має комерційних задумів.

### **3.2 Проектування архітектури систем програмного забезпечення**

У цьому підрозділі було описано найпопулярніші підходи до написання архітектури для програмних забезпечень. Окрім того, проаналізовано та виділені найкращі та найгірші наслідки кожного підходу.

#### **3.2.1 Специфіка архітектури при побудуванні web-застосунків**

Побудова архітектури – це один з основних та головних етапів при побудові будь-якого проекту, що поділяється за видами, серед яких присутні наступні види:

- Побудова застосунків для мобільних платформ (Android/iOS);
- Побудова веб-застосунків, яка є більш популярною через охоплення більшої кількості людей;
- Побудова застосунків на основі інтернет пристроїв, прикладом можуть слугувати такі датчики що мають можливість передачі інформації на відстані та сканувати, наприклад, температуру в приміщенні.
- До того ж слугують ще багато інших прикладів, але це одні з основних, що варто би виділити.

При проектуванні систем психологічної допомоги населенню було обрано проектувати саме веб-застосунок, оскільки саме цей вид охоплює найбільшу кількість користувачів. Авжеж, у подальшому, після збільшення попиту та для його розширення, система матиме зручну та швидке проектування на мобільні платформи, що зазначені в першому пункті.

При проектуванні архітектури проекту важливо досягти дві цілі: описати всі можливі та потрібні компоненти на даний час системі; та описати взаємодію цих двох компонентів між собою. В рамках проектування архітектури проекту визначається майбутній дизайн на початковому етапі, інфраструктуру проекту, користувацький досвід, набір технологій що потрібен для реалізації, програмні модулі (дизайн веб-архітектури), а також просування і монетизацію вашого майбутнього веб-додатку.

Окрім того побудова архітектура складається з визначення її абстракції, що дає зрозуміти кордони при проектуванні системи. Абстракція – важлива частина проектування, адже неможливо одразу побудувати велику систему. Після реалізація цих пунктів, проект починає створюватись.

Отже, наступним вирішальним моментом є прийняття рішення на основі вище описаних пунктів, що буде пов'язане з вибором самого типу архітектури програмного застосунку. Так, наприклад, це може бути програмний продукт моноліт, або заснований на мікро-сервісній моделі. Також, на цьому етапі команда з розробників, разом з представником від бізнесу, формує конкретні технічні вимоги до майбутнього програмного продукту та його рівні і план подальшої розробки системи.

Нижче наведені декілька параметрів, за якими можна проаналізувати якість архітектури програмного застосунку:

- Забезпечення рівня безпеки та стабільності системи;
- Швидкість у обробки запитів в системі;
- Гнучкість (можливість повторного використання компонентів програмного продукту);
- Простота програмного коду;
- Забезпечення статистикою з результатів продукту (аналітика та тести різних компонентів продукту незалежно від інших);
- Масштабність та поширення продукту і компонентів;
- Забезпечений рівень автоматизації інформаційної системи.

При роботі над архітектурою програмного застосунку великого бізнесу, гарною практикою також є інтеграція з бізнесом, де клієнт (бізнес) надає бізнес-завдання та маркетингові фактори, що мають бути враховані у проектуванні системи. Наприклад, представник бізнесу зосереджується на слабких сторонах своїх конкурентів, вираховує їх слабкі сторони під час проектування застосунку з командою розробки.

### 3.2.2 Монолітна архітектура при побудуванні застосунку

Монолітна архітектура – це традиційний підхід у розробці програмного забезпечення, який заснований на самому початку проектування системи. В такій системі, зазвичай в такій системі проектується цілісний продукт як одне ціле. До того ж, часто у такій системі намагаються проектувати з логічним розподіленням між папками, аби покращити читабельність, що насправді не сильно покращує ситуацію.

В основному, містить у собі базу даних, серверний та клієнтський інтерфейс. Всі функції поширюються між собою, оскільки представляють собою цілісну систему в одному місці та є уніфікованою.

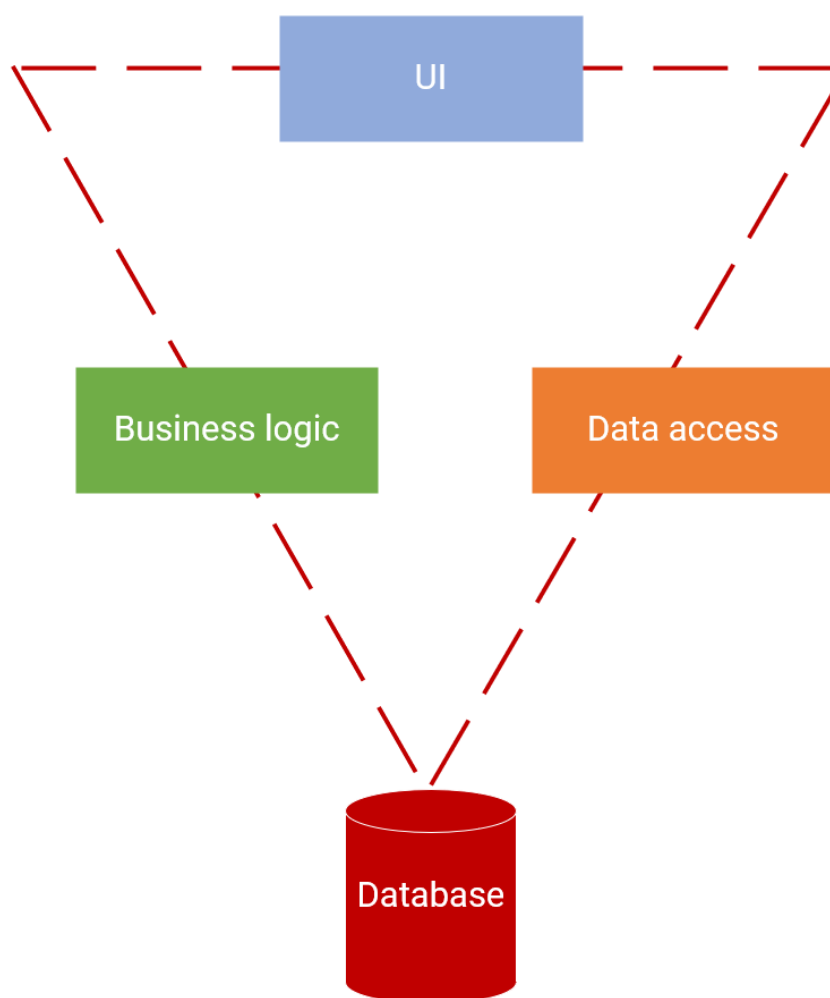


Рис. 3.1 Діаграма монолітної архітектури

Одним з головних проблем такого підходу є погіршення темпів розробки на етапі підтримки додатку. Оскільки з часом, зі сторони бізнесу буде потреба заміні одних сервісів на інші, та імплементація нового функціоналу, це може бути:

- проблема з доступом до бази даних;
- зміна доступу в додатку між собою;
- оновлення всього стеку.

Саме перераховані вище пункти призводять до обмеження оновлення та погіршення гнучкості додатку, а саме робить його оновлення більш трудомістким.

На початкових етапах проектування програмного засобу за даним підходом позитивними аспектами є:

- спрощене управління проектом та його кодом;
- зменшеному когнітивному навантаженню;
- спрощеному процесу розгортання;

Хоча така система має купу негативних проблем при побудуванні проекту, але вона також охоплює своє місце на ринку. Також одним з плюсів цього підходу є швидка розробка у порівнянні з наступними підходами, тому це гарний привід використовувати цей підхід на початку розробки бізнесу або при умові що замовник не має планів на подальшу підтримку і хоче отримати результат якомога швидше та дешевше. [32]

Прикладом такого підходу можуть слугувати відомі компанії, що мали схожий розвиток:

- 1) На початку «Amazon» проектувався саме з підходом монолітної архітектури. З розвитком компанія перейшла до сервіс-орієнтованої та мікро-сервісної архітектури для управління зростаючою складністю;
- 2) Наступним прикладом може слугувати сервіс «eBay». Він також починався з підходу монолітної архітектури під час побудови своєї платформи. З часом, компанія перейшла на гібридну систему з поєднанням монолітних компонентів з мікро-сервісами;
- 3) Ще одним прикладом може слугувати «Netflix», який відомий своєю архітектурою мікро-сервісів, хоча на самому початку працював з

монолітною архітектурою. Однак, пізніше, компанія поступово перейшла на мікро-сервіси, щоб підвищити гнучкість, масштабованість і ремонтпридатність;

- 4) Також компанія «Meta» на ранніх етапах використовував монолітну архітектуру. З розширенням компанія прийшла до висновку, що компанія потребує оновлень і перейшла до більш сервісно-орієнтованої архітектури, щоб пристосувати її до різноманітних функціональних можливостей;
- 5) Компанія «X». Витоки, як і інших компаній починались з монолітного додатку, а далі переходить до більш нового методу. Зі збільшенням кількості користувачів «X» зіткнувся з проблемами масштабування, що спонукало до переходу на більш розподілену архітектуру, яка інтегрувала мікро-сервіси.

### **3.2.3 N-tier архітектура при побудові застосунку**

N-tier архітектура – це підхід до архітектури програмного забезпечення при якому система розділяється на  $n$  шарів, де  $n$  – це кількість шарів. Найпростішим представником може бути одно-рівнева архітектура. При цьому підході вся система розділяється на окремі логічні модулі.

З розвитком часу, розробники зрозуміли, що для підтримки великих проєктів потрібно краще організувати програмний код, тому одним із вирішенням почали розділяти компоненти програми, таким чином прийшли до розробки двох-рівневої, а потім і більше архітектури. Найпростішим прикладом двох-рівневої архітектури є відгалуження бази даних до окремого компоненту, що окрім організації програмного коду також спрощує заміну цього компоненту. [21, 22]

При розробці системи, проєктується кількість рівнів які залежать від самої системи. Так при проєктуванні команда розробки може спроектувати безліч рівнів, наприклад: додати до одно-рівневої архітектури і ще один рівень

Зазвичай поділяються на:

- рівень доступу до даних;

- рівень бізнес-логіки;
- рівень представлення.

Тому класичним представником зазвичай є 3-рівнева архітектура. Окрім цих рівнів, також додаються ще, за потреби самої системи, тому і містить назву «n-tier», що залежить від конкретної системи. [33]

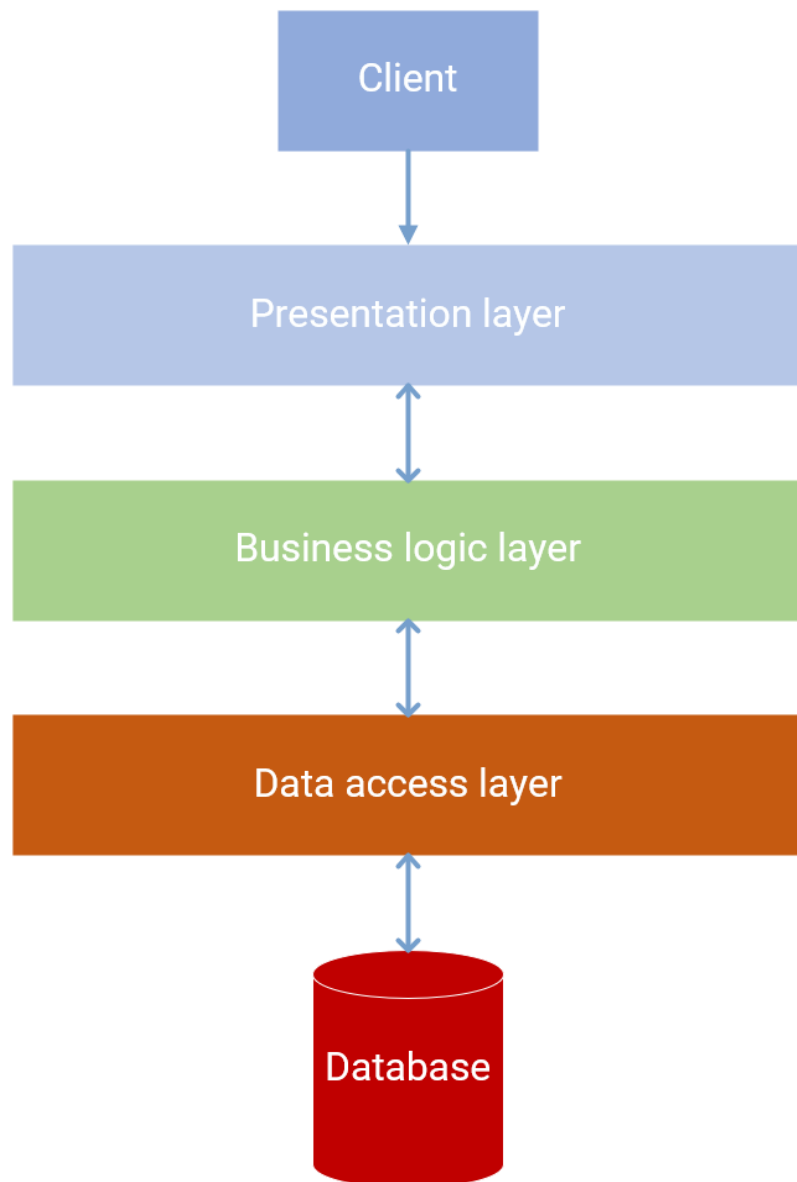


Рис. 3.2 Діаграма трьох-рівневої архітектури

Багаторівнева архітектура є добре перевіреною програмною архітектурою - вона використовується в багатьох великих додатках корпоративного рівня і дозволяє



нам створювати дуже масштабовані системи. Прикладом компаній що використовують цей підхід можуть слугувати наступні компанії: Salesforce, NetSuite, Oracle, Microsoft та багато інших, використовують її для багатьох великих систем.

### 3.2.4 Сервісно-орієнтована архітектура при побудові застосунку

Сервісно-орієнтована архітектура – це такий стиль архітектури при якому основною ціллю є організації програмних систем як набору слабо пов'язаних і сумісних сервісів.

При цьому підході, система поділяється на модулі або бізнес-функції, які представляють собою реалізацію окремих систем, що мають слабкі зв'язки між собою та розгортається і підтримується незалежно. [31]

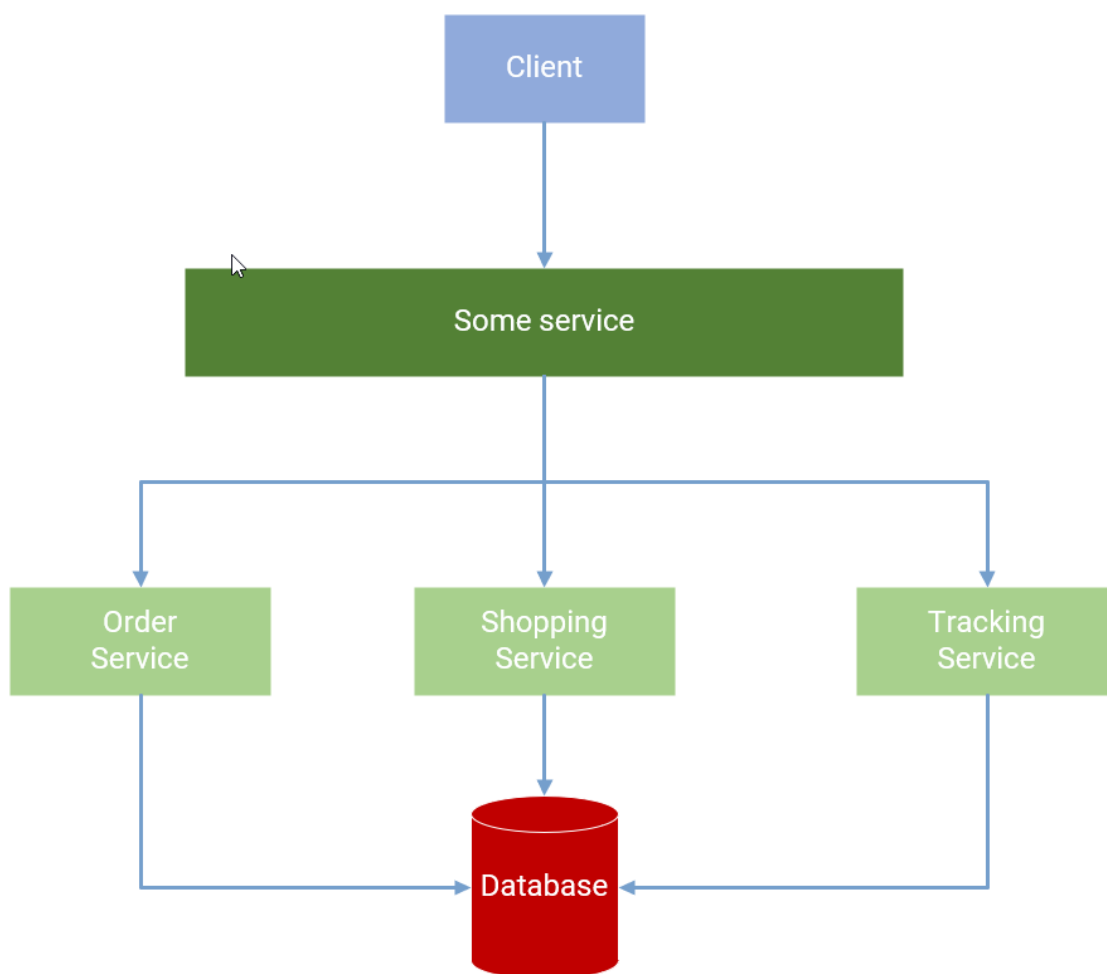


Рис. 3.3 Діаграма сервісно-орієнтованої архітектури

Ці сервіси взаємодіють один з одним через чітко визначені інтерфейси, як правило, використовуючи стандартизовані протоколи, такі як HTTP або SOAP.

Далі наводяться відомі компанії що використовують цей підхід:

- 1) Компанія «IBM». Ця компанія є одна з прихильників сервісно-орієнтованої архітектури, «IBM» пропонує спеціалізовані інструменти та рішення для впровадження сервіс-орієнтованих архітектур в корпоративних середовищах;
- 2) Компанія «Microsoft». Принципи сервісно-орієнтованої архітектури глибоко інтегровані в продукти та сервіси компанії «Microsoft», що охоплюють хмарні сервіси Azure та різноманітні корпоративні програмні рішення;
- 3) Компанія «Salesforce». Це CRM-платформа «Salesforce», побудована на сервісно-орієнтованій моделі, яка дозволяє компаніям персоналізувати та інтегрувати послуги відповідно до їхніх конкретних потреб;
- 4) Компанія «Walmart». Сервісно-орієнтована архітектура слугує основою для управління масштабними операціями електронної комерції та роздрібною торгівлі «Walmart», що забезпечує безперебійну взаємодію між різноманітними системами та сервісами;
- 5) Компанія «Bank of America». Фінансові установи, включаючи «Bank of America», використовують сервісно-орієнтовану архітектуру для безперешкодної інтеграції різноманітних банківських послуг, покращення клієнтського досвіду та забезпечення відповідності нормативним вимогам.

### **3.2.5 Мікро-сервісна архітектура при побудові застосунку**

Мікро-сервісна архітектура – це підхід, в якому використовується стратегія для створення додатків. Цей підхід має важливу відмінність від всіх інших та є нащадком починаючи від монолітної архітектури. Суть архітектури дещо схожа на сервісно-орієнтовану, але з нюансом, кожен сервіс виступає окремим додатком,

тобто суть системи полягає в наступному тезисі «система є незалежною частиною від окремих сервісів». [23]

Кожен мікро-сервіс є самодостатнім та фокусується на виконанні однієї цілі (Див. Рис. 3.4), яка була призначена конкретно для цього сервісу окремо, прикладом може слугувати сервіс для нотаток, де операціями буде додавання, видалення та редагування існуючих нотаток.

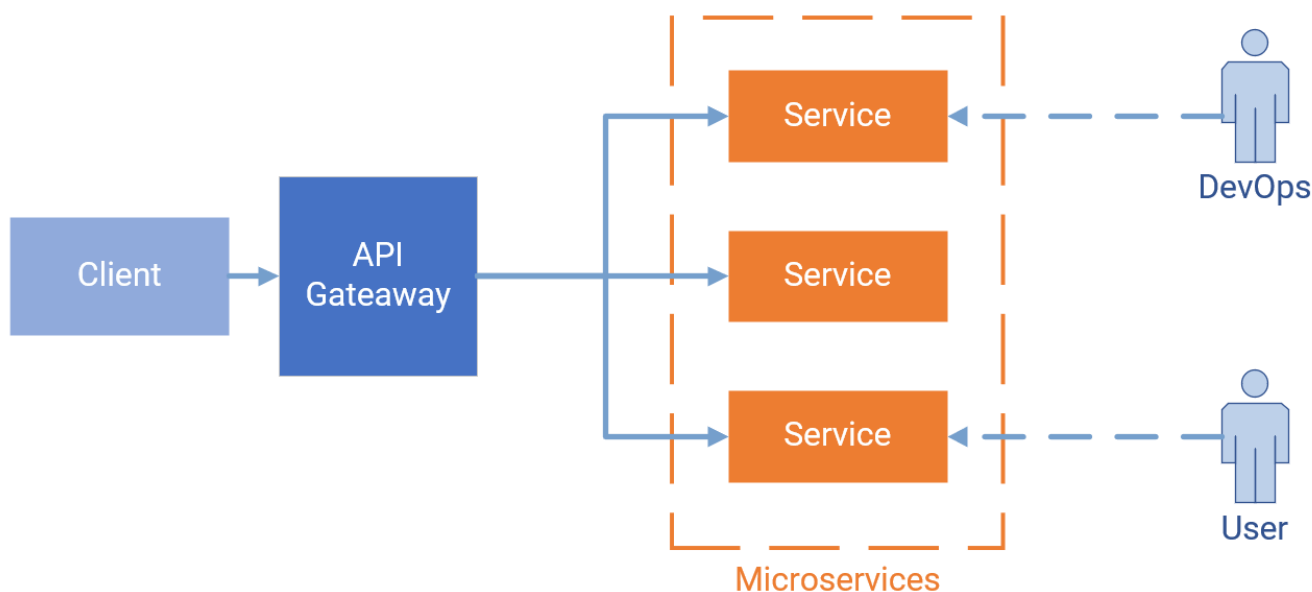


Рис. 3.4 Діаграма мікро-сервісної архітектури

Отже, мета використання мікро-сервісів – підвищити гнучкість, масштабованість, автономність та простоту обслуговування шляхом розбиття додатків на менші, більш керовані частини.

Окрім вище згаданих компаній що перейшли на мікро-сервісну архітектуру можна додати наступні сучасні компанії:

- 1) Компанія «Uber». Мікро-сервісна архітектура – це одна з найпопулярніших та важливих підходів для великих проектів, однією з яких є «Uber», де цей підхід відіграє ключову роль у безперебійному управлінні поїздками та платежами;

- 2) Компанія «Spotify». У цій компанії мікро-сервіси використовуються у музичному додатку «Spotify», що представляє своїм користувачам персоналізовані рекомендації;
- 3) Компанія «Airbnb». За допомогою мікро-сервісів компанія спрощує процеси бронювання житла на платформі «Airbnb»;
- 4) Компанія «Nike». За допомогою мікро-сервісів покращують роботу інтернет-магазину «Nike», де надають клієнтам персоналізований досвід покупок своїх речей.

### **3.3 Архітектура у проектуванні API застосунків**

У сучасності проектування архітектури засновується на розділенні системи на логічні блоки і чим більше така система тим більше опираються на проектування гнучких систем.

Тому в сучасних системах окрім розділення на окремі компоненти різних шарів таких як бізнес логіки, бази даних та інші, окрім розділення на різні модулі та сервіси, як окремі системи, розробники прийшли до розподілення рівня представлення.

Таким чином, при проектуванні будь-якої системи за будь-якою з вище описаних архітектур, виділяється окремий рівень, що може використовувати різні назви, але головна суть цього шару саме рівень представлення, або так зване «API».

API (або «Application Programming Interface») - це інтерфейс прикладного програмування, якщо перекладати дослівно. Говорячи більш зрозуміліше, це набір визначень та протоколів для інтеграції між застосунками.

При виділенні шару представлення розробники прийшли до висновку розділення якомога на більше незалежних частин, тому при проектуванні такої системи, винайшли шаблони за якими будується система, виділяють три системи:

- MVP
- MVVM
- MVC

При побудові систем є нюанси по кожній, хоча, на перший погляд, в цілому вони містять схожу будову і принципи.

### 3.3.1 Побудова на базі MVC

MVC - це найперший архітектурний патерн представлений з MV\* подібних патернів. При побудові застосунків на шаблоні MVC додаток розділяється на відповідні шари відокремлені один від одного. Що мінімізує всі зусилля, необхідні для розширення, тестування та підтримки програми. Процес максимізує розмір класу представлення між інтерфейсом користувача, логікою зв'язування даних і бізнес-операціями. [13]

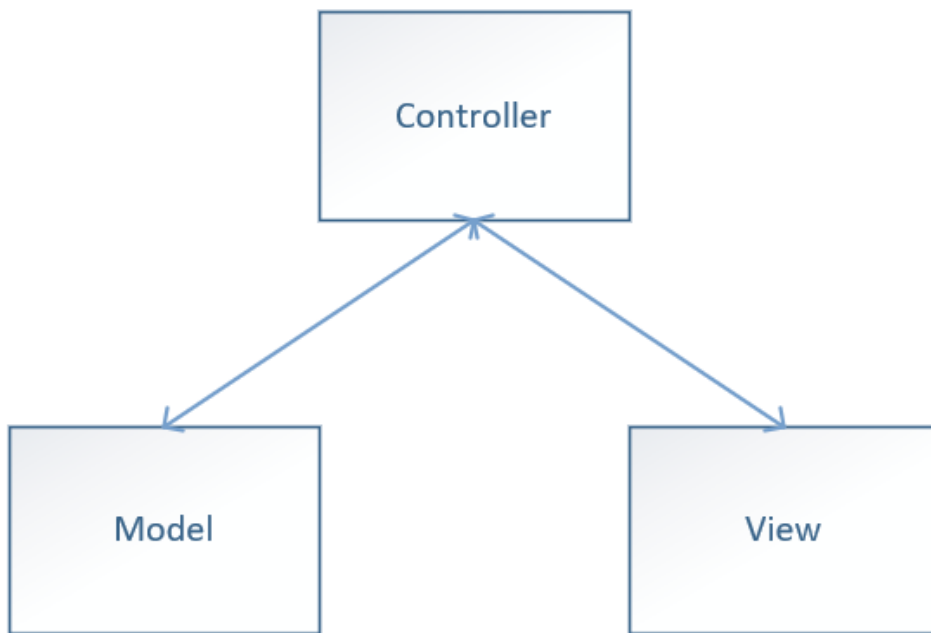


Рис. 3.5 Побудова шаблону MVC

Підсумовуючи, MVC пришвидшує розробку завдяки паралельній роботі, мінімізації дублювання коду та підтримці різноманітних додатків. Гнучкість при використанні цього патерну дозволяє легко змінювати дизайн, не впливаючи на основну модель, що є однією з головних переваг шаблону. [14]

### 3.3.2 Побудова на базі MVP

Описуючи шаблон MVP варто зазначити що він має багато спільного з патерном MVC, авжеж оскільки він є похідним від цього патерну. В цілому головна різниця між ними це заміна рівня контролера на представлення. MVP - це архітектурний патерн, який можна використовувати для усунення деяких недоліків першого. Будь які елементи на сторінці відображаються за допомогою рівня вигляду. Рівень представлення звертається до подій інтерфейсу користувача і збирає всі вхідні дані, що надходять від користувачів, а потім переміщує їх на сторону моделі, передаючи результат у рівень вигляду.

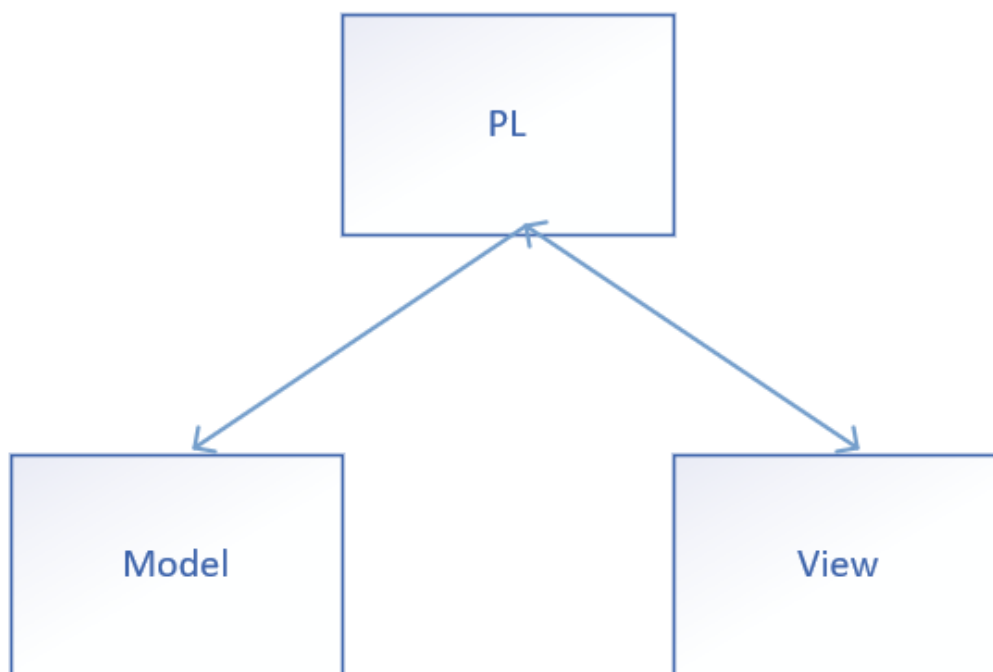


Рис. 3.6 Побудова шаблону MVP

Говорячи в цілому то MVP спрощує налагодження додатків завдяки трьом рівням абстракцій. Архітектура дозволяє проводити модульне тестування під час розробки, ізолюючи бізнес-логіку від представлення. Кілька рівнів представлення в MVP покращують повторне використання програмного коду та гнучке управління представленням.

### 3.3.3 Побудова на базі MVVM

Шаблон Model-View-ViewModel (MVVM) допомагає відокремити бізнес-логіку та логіку представлення будь-якого додатку від інтерфейсу користувача. Цей шаблон допомагає вирішити велику кількість проблем які виникають при етапі розробки. Притримування цьому підходу полегшує тестування, підтримку та розвиток додатку. Окрім того значно покращує можливості повторного використання коду і дозволяє розробникам і дизайнерам інтерфейсу легко співпрацювати при розробці відповідних частин програми. У патерні MVVM є три основні компоненти: модель, представлення та модель представлення. [15]

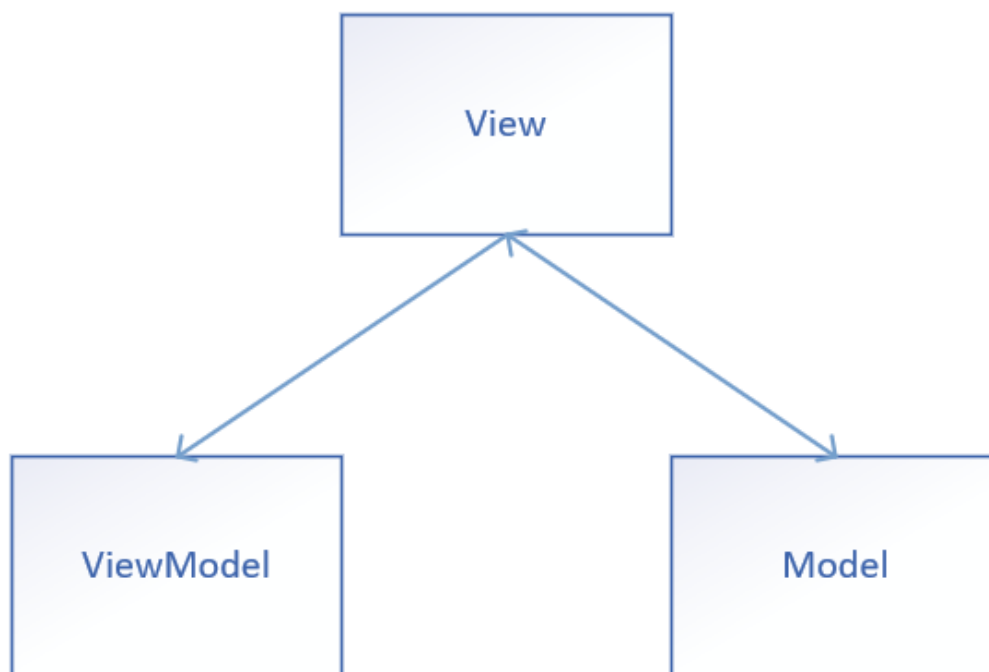


Рис. 3.7 Побудова шаблону MVVM

Підсумовуючи, MVVM спрощує модифікацію коду завдяки чіткому розділенню компонентів, прискорюючи роботу та полегшуючи випуск нових версій. Написання модульних тестів для основної логіки є простим завдяки ізольованим залежностям. Архітектура забезпечує незалежну роботу дизайнерів і розробників, дозволяючи редизайн інтерфейсу без зміни коду, за умови, що представлення реалізовано на XAML.

### **3.4 Побудова ПЗ та інтеграція з системою контролю версій Git**

Система контролю версій дозволяє відстежувати зміни на великому відрізку часу. Це є важливо частиною розробники, оскільки постійно, у великих чи навіть не дуже великих компаніях, доводиться працювати з однією системою та її файлами. Суть цієї системи полягає в поєднуванні використанні одних і тих же файлів та забезпечення безпеки даних. Наприклад, коли проектується система можуть видалятися ті чи інші рішення, для того щоб зменшити витрати на це використовується система контролю версій. Під час історії розвитку ІТ сфери, розвивалося безліч систем контролю версій, але саме Git є найпопулярнішою та найпоширенішою, краще навіть сказати що всі інші системи використовуються тільки на 10% від всіх проектів, настільки малий відсоток. Звертаючись до точних даних, у приклад можна привести опитування за 2018 рік, в якому прийняло участь понад 100 000 розробників та серед яких 78,0% респондентів, що користуються системою контролю версій Git. Git – це розподілена система контролю версій, що означає використання локального серверу, це робить Git більш гнучким поміж інших систем VCS. VCS дозволяє розробникам отримати доступ до історії проекту, що допомагає знайти відповіді на питання «які зміни були внесені? де? коли?» та «хто їх використав?» З якою метою були внесені зміни? Саме ці прості рішення допомагають в розробці великих проектів. [29]

### **3.5 Розділення моделі користувача**

Оскільки система була побудована таким чином щоб її користувачі мали змогу доступу до функціоналу обох сторін, то обидва користувачі було умовно поділено на два типи:

- Лікар – це модель заснована на ідеї реалізації щось схожого до маркетплейсу, де система не заточується на перевірці доктора, а лише дає поширювати свою діяльність;
- Пацієнт – ця модель користувача має доступу до звичайного функціоналу, наприклад, перегляду своїх опитувань, тощо.



Отже, ця система, містить у собі розподілення користувача на два модулі, але він може використовувати одночасно весь функціонал системи.

### 3.6 Проектування програмного засобу з психологічної допомоги

На цьому етапі описуються і будуються UML діаграми для побудови застосунку.

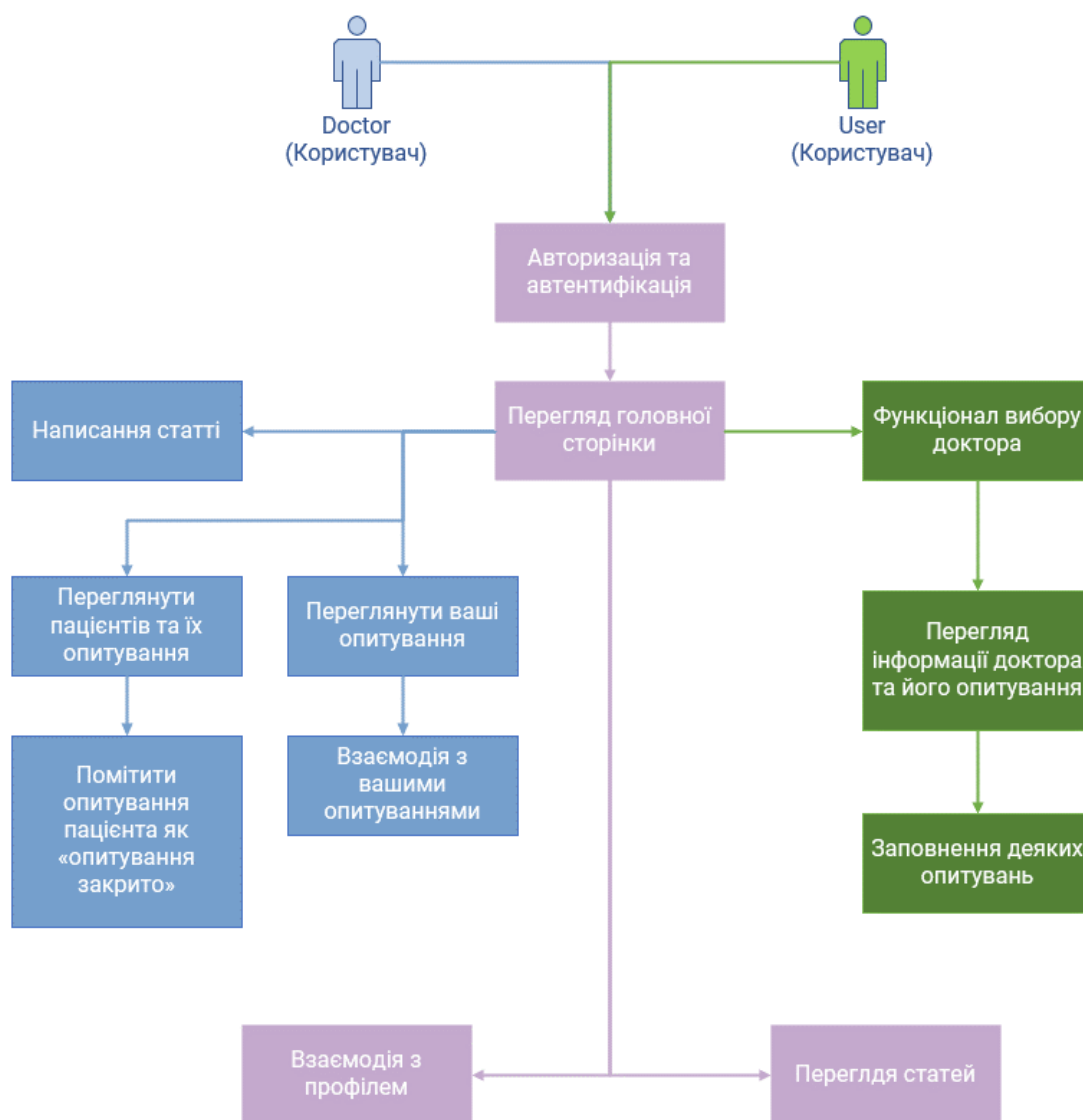


Рис. 3.8. Діаграма взаємодії

Перша діаграма, що є діаграмою взаємодії (Див. Рис. 3.8), містить опис взаємодії користувачів з системою. Оскільки кожен користувач може бути як

лікарем, так і клієнтом, тому було прийнято рішення розділити діаграму на дві моделі, а саме лікар та клієнт.

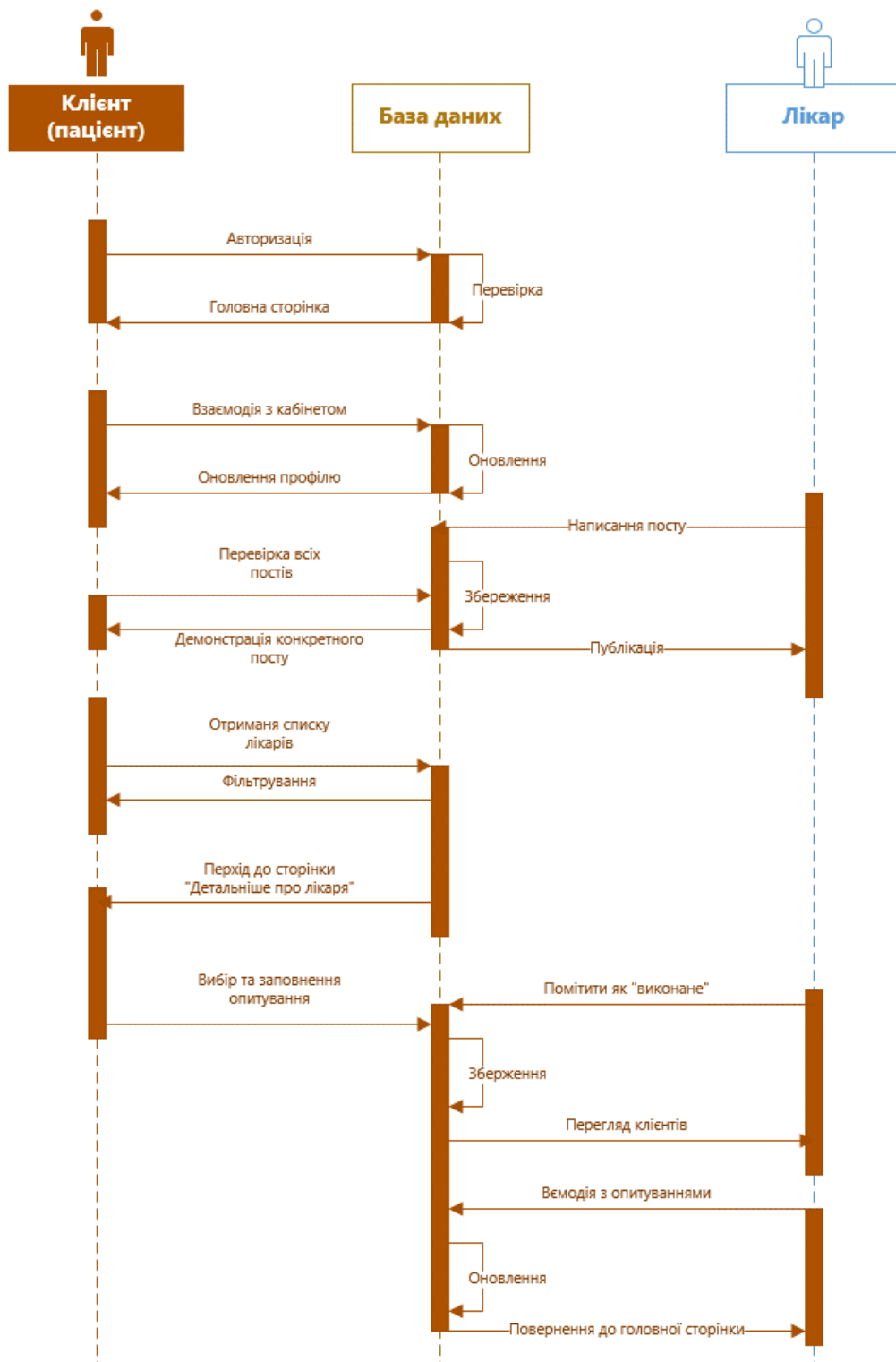


Рис. 3.9 Діаграма послідовності

Наступним представником є діаграма послідовності (Див. Рис. 3.9). На цій діаграмі демонструється послідовність з виконання деякого функціоналу в системі, між клієнтом та лікарями.

### **3.7 Специфікація автентифікації**

У сучасному світі ми спостерігаємо прогрес цифровізації, який починає впливати на життя людини прямо сьогодні, а отже тематика безпеки зараз є на часі, та практикуватиметься разом, рука об руку з цифровізацією. Отже, автентифікація є необхідним аспектом повсякденного життя.

Автентифікація – це процес підтвердження особи чи пристрою. Головна функція автентифікації – це захист зв'язку між клієнтом та сервером. Користувач – це особа, яка виконує вхідний запит, що супроводжується набором ідентифікаційних даних. [28]

З поширення інформаційних систем, а особливо інтеграцію з життєво важливими для людини системами, питання стабільності механізму автентифікації стало в край важливим, наприклад: захист інформації в онлайн-застосунку лікарні чи сервіс з оплатою податків, покупок, тощо. Тому людство прийшло до пошуку нових вирішень безпеки для систем.

#### **3.7.1 Історія типів автентифікації їх розвиток**

На початку розвитку, системи автентифікації мали простий та зрозумій функціонал, який опирався на паролі та логіни, які зберігалися на сервері.

В таких системах використовували наступну логіку автентифікації, якщо користувач під час входу вводить логін та пароль, що підходять та збережені в базі даних, такий користувач вважався коректним, а автентифікація мала задоволений результат («Success»). Звісно подібний підхід не був надійний на сьогодні.

Розглянемо паролі ті їх суть більше детальніше, зазвичай вони містять у собі наступне:

- Користувач вигадує набір символів, цифр та букв;
- Людина запам'ятовує цю комбінацію;

- Користувач вводить її повторно для доступу до системи.

В той же час з'являється і крадіжка паролів, одним з перших зареєстрованих прикладів крадіжки паролів був зареєстрований у 1962 році. До того ж, дослідження доводять, що люди не можуть запам'ятати паролі, тому зловживають наступними пунктами що шкодить системі.

- Запис паролів на клаптику паперу;
- Часто користувачі використовують вже існуючі в інших системах паролі;
- При запиті на оновлення, користувачі, часто до вже існуючого паролю додають декілька символів.

Звісно, всі ці методи не є раціональними та більш шкодять системі. До того ж, такий метод автентифікації негативно впливає на систему, адже при кожній такій операції створюється нова транзакція, що навантажує систему. В результаті подібних операцій це негативно впливає на продуктивність системи.

До того ж, зараз, впроваджується декілька відносно нових методів для покращення системи автентифікації та її безпеки. Таким чином, розробники винайшли наступні декілька типи автентифікації, які будуть описані далі детальніше:

- Cookie-файли;
- Web Tokens та JWT.

### **3.7.2 Детальніше про cookie-файли**

Cookie-файли - це малі за розміром файли, що використовуються веб-сервісами. Вони містять у собі інформацію про відвідування, що в свою чергу дозволяє уникнути постійної авторизації у систему. [16]

Cookie-файли поділяються на декілька типів:

- Постійні cookie-файли зберігають у собі дані протягом довгого періоду, саме вони допомагають зберігати інформацію про ім'я користувача та пароль;
- Сторонні cookie-файли містять у собі дані про активність в Інтернеті, надсилають її власникам веб-сайтів для покращення системи реклами;
- Сесійні cookie-файли зберігають у собі усіяку інформацію про сеанс користувача, наприклад: додавання речей до кошику. Ці cookie-файли виділяються тим що видаляються одразу після закриття браузера.

Cookie-файли зазвичай не містять у собі віруси чи шкідливе програмне забезпечення, причиною тому є незмінність даних при пересилці. [17]

Але, деяке шкідливе забезпечення може маскуватися під них. Сторонні cookie-файли можуть також містити у собі шкідливе забезпечення, оскільки вони полегшують стороннім особам відслужувати вашу діяльність в мережі Інтернет.

### **3.7.3 Детальніше про Web Tokens та JWT**

Токен (або «Token») – це безпечний формат даних для спілкування та передачі конфіденційної інформації між клієнтом та сервером. Він представляє собою дані розміщені у компактному вигляді. Зазвичай вони використовуються для посилення процесу автентифікації у системі.

При автентифікації на основі токенів, сервіс перевіряє запит до сервера. Після завершення перевірки, сервер повертає токет на токкер-перезагрузки і відповідає на запит.

Користувач може використовувати один пароль, але токен надає більше складніший тип доступу, що ускладнює процес крадіжки або злому. До того ж, система не навантажується в процесі, оскільки запис сесії не займає місця на сервері. [19]

JWT (або «JSON Web Token») – це відкритий стандарт, що будує взаємодію між клієнтом та сервером обмінюватися інформаційною безпекою. Кожен JWT, закодований на JSON-об'єкти що містить набір тверджень.

JSON (або «JavaScript Object Notation») - це стандартний текстовий формат для кодування структурованих даних на основі об'єктного синтаксису JavaScript. Цей формат є найпоширенішим та використовується для передачі даних у веб-додатках. Наприклад: при обміні інформацією між клієнтом та сервером для відображення даних.

JWT підписуються за допомогою криптографічного методу, щоб гарантувати, що твердження не можуть бути змінені після випуску токена. Він стислий, читабельний і підписаний цифровим підписом провайдера ідентифікації (IdP , або «Identity Provider») за допомогою приватного та відкритого ключа. Таким чином, інші сторони можуть перевірити цілісність і легітимність токена. Метою використання JWT є підтвердження достовірності даних.

Кодування використовується для того, аби зберегти дані придатними для використання і може бути скасоване за допомогою того ж алгоритму для декодування інформації. Шифрування використовується для збереження конфіденційності даних і потребує приватного ключа, що зберігається на сервері в таємниці, для їх розшифрування.

Дивлячись на те що JWT – це метод автентифікації заснований на токенах. Оскільки токен зберігається одразу після першої автентифікації у системі, взаємодія між сервером та клієнтом скорочується, що зменшує завантаженість на систему.

Головню різницею між JWT та Web Tokens є те що перший, зберігає інформацію про себе на стороні клієнта, в той час як інший створює транзакцію у базі даних, аби витягти потрібну інформацію про користувача, пов'язану з токеном, а потім взаємодіяти з цими даними.

В свою чергу JWT містить у собі усю потрібну інформацію для взаємодії з системою. По суті, основна частина токена завжди містить у собі наступні параметри:

- дату та час створення і закінчення терміну дії токена;
- інформацію про користувача (ролі, пошти, вік, тощо).



1) *Заголовок* (або «header»):

Це початкова частина, яка містить у собі зашифровану інформацію що складається з двох частин: тип токена, та метод його підписання (HMAC, SHA256 або RSA).

Сам заголовок це JSON-об'єкт, що закодований під Base64, де Base64 - це набір методів кодування двійково-десятькового коду в текст, які перетворюють двійкові дані в представлення radix-64 у форматі рядка ASCII.

Отже, на описаному вище токени, прикладом заголовку є наступний малюнок (Див. Рис. 3.12), де розшифровується його вміст.

HEADER: ALGORITHM & TOKEN TYPE

---

```
{  
  "alg": "HS256",  
  "typ": "JWT"  
}
```

Рис. 3.12 Розшифровка першої частини токена

2) *Вміст* (або «payload»):

Другим елементом токена є корисне навантаження, яке містить твердження. Претензії - це твердження про суб'єкт (зазвичай, користувача) та додаткова інформація. Існує три типи тверджень: зареєстровані твердження, публічні твердження і приватні твердження.

- Зареєстровані твердження:- Це набір попередньо налаштованих тверджень, які не є обов'язковими, але рекомендуються для надання набору релевантних, сумісних між собою тверджень. Наприклад, iss (емітент), exp (термін дії), sub (тема), aud (аудиторія) та інші.



- Загальнодоступні запити:- Це користувацькі запити, які користувач може визначити за допомогою алфавітно-цифрових символів. Однак вони повинні бути зареєстровані в реєстрі веб-токенів IANA JSON або використовувати стійке до колізій ім'я, щоб уникнути зіткнень.
- Приватні вимоги:- Приватні вимоги часто включають інформацію, унікальну для вашої компанії, наприклад, внутрішній ідентифікатор користувача. Приватні вимоги, на відміну від публічних, вразливі до колізій, оскільки вони не записуються, і їх слід використовувати з обережністю.

Отже, при описанні першої частини токена вище було наведено приклад заголовку. В той час як друга частина містить у собі корисну інформацію для системи. На малюнку демонструється приклад розшифрування реального токена (Див. Рис. 3.13).

PAYLOAD: DATA

---

```
{  
  "Id": "8",  
  "sub": "spiceAndWolf@holo.com",  
  "email": "spiceAndWolf@holo.com",  
  "jti": "235f8f24-ab2b-4ce8-89a0-bb7400908204",  
  "iat": 1677516360,  
  "role": "CustomUser",  
  "nbf": 1677516360,  
  "exp": 1677516450  
}
```

Рис. 3.13 Розшифровка другої частини токена

### 3) Сигнатура (або «signature»):

Підпис - це третя частина токена, хоча комусь може здатися що вона не є важливою, але це не так, оскільки без цієї структури шифрування не вдасться отримати інформацію.

Для створення секції підпису ви повинні підписати закодований заголовок, закодоване корисне навантаження, секрет і алгоритм, вказаний в заголовку.

Наприклад, якщо ви хочете використовувати алгоритм HMAC SHA256, підпис буде згенеровано відносно їх структури. Аналізуючи приведений в приклад токен, можна отримати інформацію що тут використовується SHA256 з наведених даних на схемі (Див. Рис. 3.14).

```
VERIFY SIGNATURE

HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  secret_key
)  secret base64 encoded
```

Рис. 3.14 Розшифровка сигнатури частини токена

Підпис використовується для перевірки того, що відправник JWT є тим, за кого себе видає, і для того, щоб переконатися, що повідомлення не було змінено на шляху до нього. Підпис створюється за допомогою секретного ключа, відомого лише серверу, який потім може бути використаний для перевірки автентичності JWT на стороні клієнта.

Важливо відмітити основні **переваги** JWT над cookie-файлами, серед цих переваг використання токенів є:

1) **Масштабованість.**

На відміну від cookie-файли, автентифікація на основі токенів є безстатутною. Це означає, що ніяка інформація про користувача не зберігається в базі даних або на сервері. Оскільки сервер відповідає лише за створення та перевірку токенів, за допомогою механізму на основі токенів можна побудувати більш масштабовані рішення, ніж

на основі cookie-файлів. Веб-токен JSON зберігається на стороні клієнта (як правило, в локальному сховищі, але також можуть використовуватися cookie-файли).

2) ***Продуктивність.***

Бекенд виконує пошук при використанні автентифікації на основі cookie-файлів, незалежно від того, чи це типова база даних SQL, чи NoSQL. Зворотній шлях, безумовно, займе більше часу, ніж декодування токена. За допомогою токенів можна уникнути декількох пошукових запитів для отримання та обробки необхідних даних, оскільки деякі дані знаходяться всередині JWT, наприклад, рівень дозволу користувача.

3) ***Готовність до мобільних пристроїв.***

Сучасні API не просто взаємодіють з браузерами. Один API, якщо він правильно написаний, може обслуговувати як браузер, так і нативні мобільні платформи, такі як iOS та Android. Cookie-файли та нативні мобільні платформи не уживаються разом. Існує кілька обмежень, при використанні cookie-файли на мобільних платформах. З іншого боку, токени набагато простіше встановити на iOS та Android. Токени також простіше розгортати для додатків і сервісів Інтернету речей, які не використовують парадигму зберігання cookie-файлів.

4) ***Наступні сервіси.***

Ще одним поширеним шаблоном сучасних веб-додатків є їхня залежність від наступних сервісів. Наприклад, виклик до головного сервера програми може надіслати запит на наступний сервер ще до того, як буде оброблено початковий запит. Проблема полягає в тому, що cookie-файли не можуть легко передаватися на наступні сервери і не можуть інформувати про статус автентифікації користувача. Існує великий опір потоку, оскільки кожен сервер має свою схему cookie-файлів, і підключення до них є складним завданням.

## 5) *RESTful API-сервісу.*

Традиційний підхід використання сеансів і cookie-файлів для ідентифікації користувача не є ефективним, оскільки вони вносять стани в додаток. RESTful API повинен бути безстатусним, що означає, що відповідь в межах певних параметрів завжди може бути очікуваною без побічних ефектів, коли робиться запит. Стан автентифікації користувача вводить такий побічний ефект, що порушує цю концепцію. Стандартним є те, що API обслуговується з одного сервера, а споживається з іншого. Оскільки cookie-файли можуть бути використані тільки для домену, де вони були створені, вони мало корисні для API на інших доменах, окрім додатків. У цьому сценарії використання JWT для автентифікації означає, що RESTful API є безстатусним. У випадку JWT вам не потрібно турбуватися про те, де обслуговується API або додаток. [19, 20]

## **Висновки**

В цьому розділі було розглянуто сучасні прототипи деяких систем. Окрім того відзначається важливість проектування гарної системи з врахуванням потреб ринку та бізнесу.

Важливою складовою цього розділу є аналіз сучасних архітектур з використанням їх на ринку. Було розглянуто декілька найбільш важливих з них, а саме: монолітна, N-tier, сервісно-орієнтована, мікро-сервісна.

Окрім того, було продемонстровано найбільш впливові компанії що використовують ті чи інші архітектурні підходи, це допоможе системі одразу сфокусуватися на тенденціях ринку та бути конкурентоспроможними.

Також було проведено аналіз з побудови сучасних API. Було розглянуто такі популярні шаблони як MVP, MVC та MVVM

Під час проектування системи було розглянуте умовне розподілення моделей користувача на дві системи, представлено діаграми послідовності та взаємодії в системі. До того ж, було розглянуто важливу частину системи – захист даних на

основі токенів, де були наведене порівняння та історію їх винайдення.

Висновком з цього розділу є розуміння кращого використання тих чи інших архітектурних підходів з залученням шифрування даних на сучасних технологіях для специфічного проекту.

## **РОЗДІЛ 4**

### **ПРОТОТИП ПРОГРАМНОГО ЗАСТОСУНКУ ПСИХОЛОГІЧНОЇ ДОПОМОГИ НАСЕЛЕННЮ**

#### **4.1 Побудова прототипу для психологічної допомоги населенню**

У розділах вище було проведено аналіз потреб та специфіку у галузі психологічної підтримки.

В поточному розділі було побудовано та проаналізовано програмний застосунок психологічної допомоги населенню.

##### **4.1.1 Створення прототипу**

Першим кроком при побудові будь-якої системи є її планування, це довгий та кропіткий процес, але навіть під час подібних процесів можуть виникнути проблеми. Одна з найпоширеніших проблем при побудові прототипу це людський фактор, а саме не врахування деяких не дуже очевидних аспектів програми.

Прикладами таких помилок можуть бути:

- погано продуманий та побудований прототип;
- внесення великої кількості правок до проекту;
- порушення деяких норм при побудові прототипів;
- деякі інші проблеми, але які вже не є сильно значимими.

Для запобігання подібних помилок використовуються системи для побудування прототипів, однією з таких систем є програмний засіб «Figma» (Див. Рис. 4.1). Він допомагає організувати процес побудови прототипу, налагодити систему та зекономити купу часу під час розробки, адже система не матиме великих негативних виправлень. До того ж, такий підхід при побудові системи є набагато кращим за використання прямої реалізації, оскільки саме під час цього процесу затрати на час внесення правок є занадто складним. Окрім того це погіршує якість написання програмного продукту.

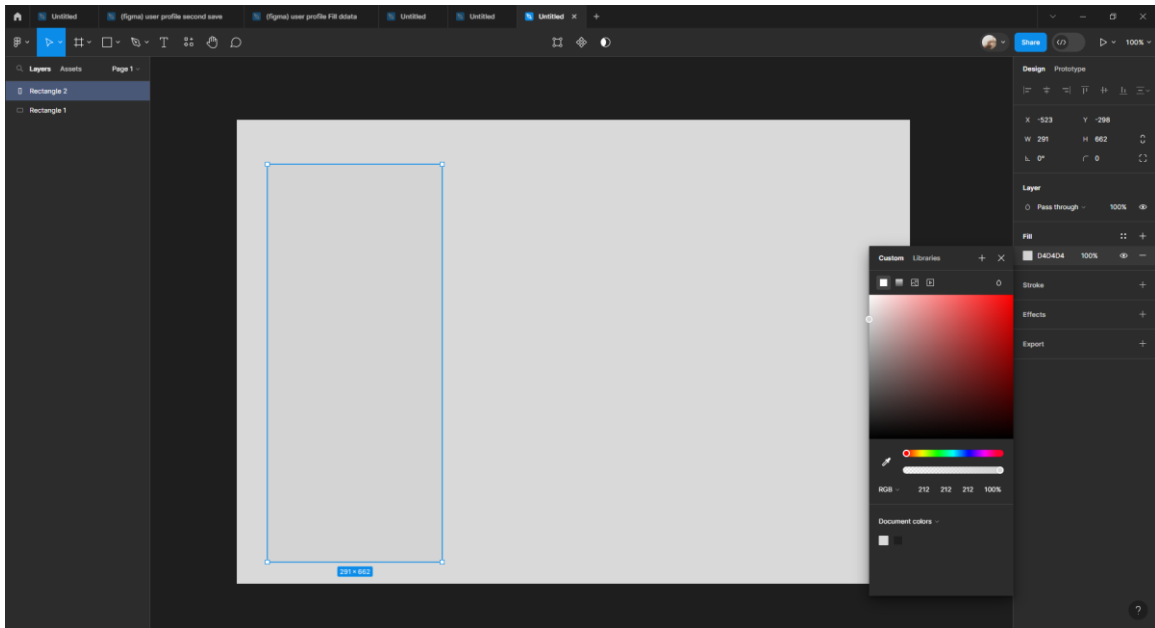


Рис. 4.1 Побудова прототипу у програмному продукті «Figma»

При побудові системи, важливо зазначити що у такому продукті є доволі гарні вирішення для цілої команди розробки, наприклад, команда може кооперуватися у створенні прототипу для більш продуманої реалізації, що є важливим аспектом на сьогодні.

#### 4.2 Аналіз прототипу

Для побудови системи було вирішено проектувати її з використанням N-tier архітектури, оскільки система представляє собою невеличку систему для ринку, з подальшим зростом, проекту може бути перебудований під більш сучасні архітектури як це робили відомі компанії світ.

При побудові системи було враховані основні тенденції ринку (Див. Рис. 4.2). У ця система поділяється на чотири модуля, які взаємодіють між собою, ось їх перелік:

- Domain. На цьому рівні виноситься спільні модулі для зменшення залежності і пропорційного зростання складності коду;
- Data Access. Рівень доступу до даних описує базу даних, взаємодію з нею та конфігурацію з використанням EF fluent api;

- Business Logic. Проміжний рівень між доступом до даних та представлення. Містить у собі набір сервісів;
- PL. Рівень представлення, що представляє інтереси клієнта.

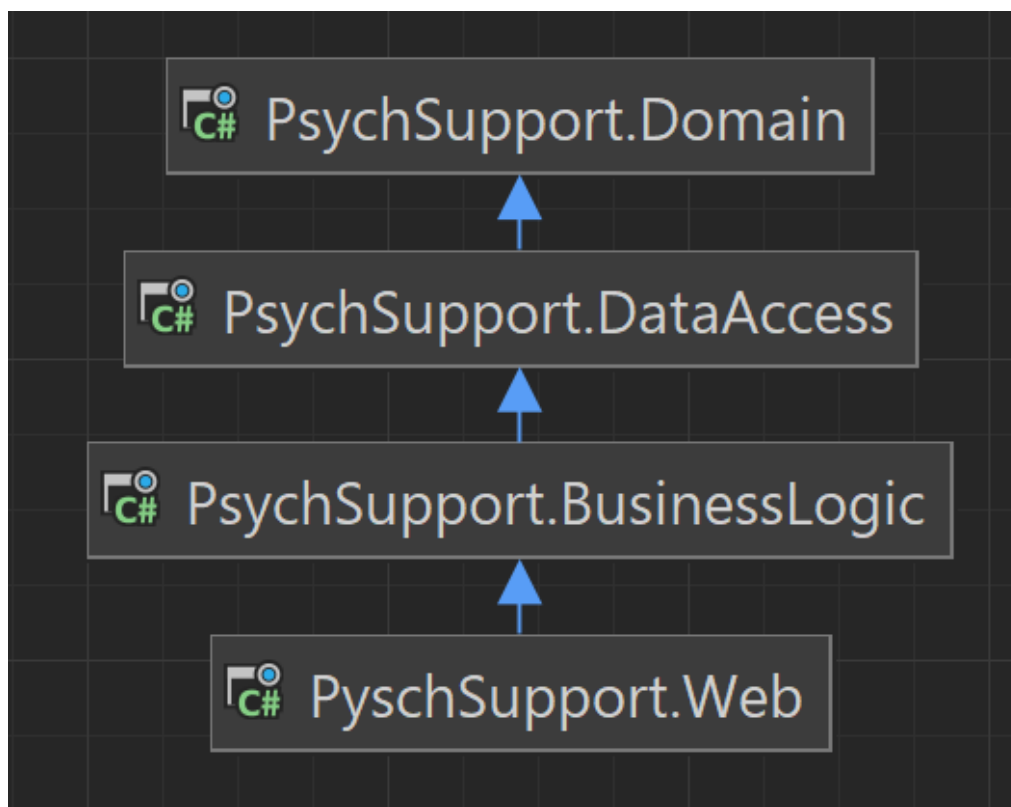


Рис. 4.2 Будова архітектури проекту

#### 4.2.1 Опис функціоналу

Кожна система має реалізовувати свій набір функціоналу, тому в цьому розділі буде розглянуто опис функціоналу:

- Користувач може використовувати систему як лікар і пацієнт;
- Користувач авторизується у системі та може переглядати наявних лікарів, розмістити себе як лікаря та пройти опитування;
- Можливість фільтрації лікарів;
- Зручний доступ до контактної інформації;
- Перегляд інформації про лікаря, розміщену самим лікарем;
- Лікар може створювати опитування та завершувати їх.



## 4.2.2 Представлення інтерфейсу користувача

Коли користувач потрапляє до системи перше що запропонує система це увійти до системи. Якщо користувач не має доступу або немає в системі, він має перейти до пункту «забув пароль» та отримати вказівки для оновлення паролю. У разі, якщо користувач не має облікового запису йому запропонують його створити.

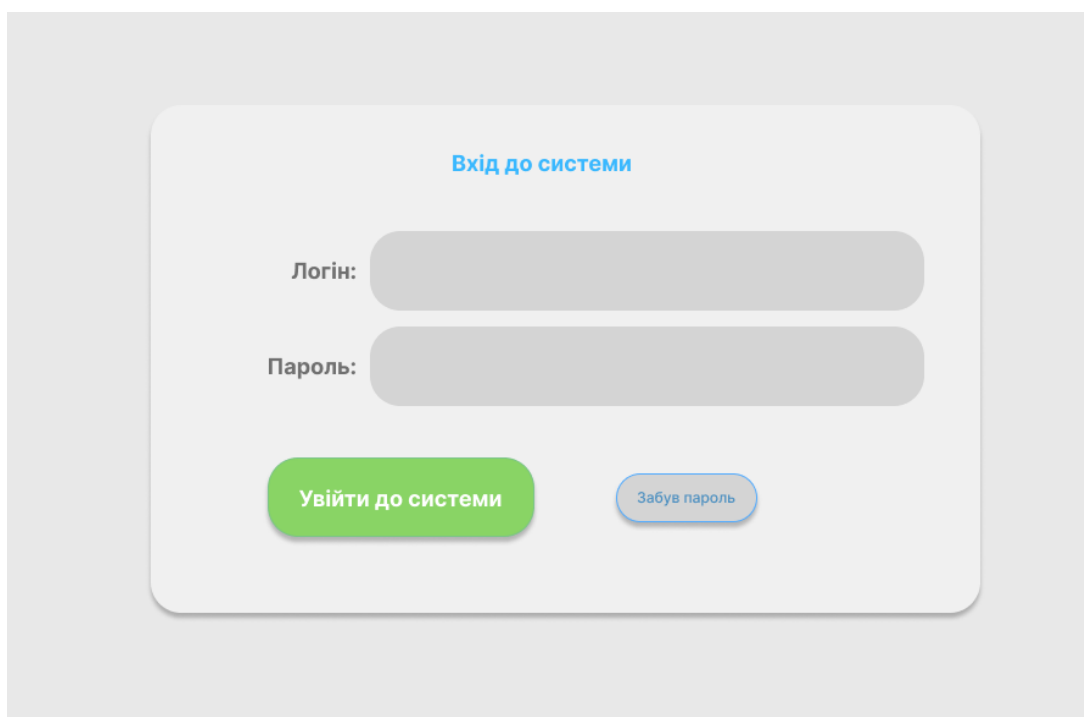


Рис. 4.3 Побудова систем авторизації

Після входу у систему у користувача є декілька варіантів для поведінки. Перш за все він може як лікар заповнити свої дані та розміститися у системі, проте варто зазначити що конкретний лікар може з'явитися у пошуку лише через деякий час, аби не навантажувати систему.

Якщо користувач захоче переглянути лікарів він переходить до посилання на список лікарів, після чого переглядає. Приклад подібної сторінки можна переглянути нижче (Див. Рис. 4.4). При цьому, користувач може переглянути інформацію про лікаря детальніше, хоча коротка інформація завжди доступна при пошуку. Окрім цього, користувач може фільтрувати лікарів за бажаними фільтрами, такими як «тип направлення», «загальні фільтри» та «спеціалізація».

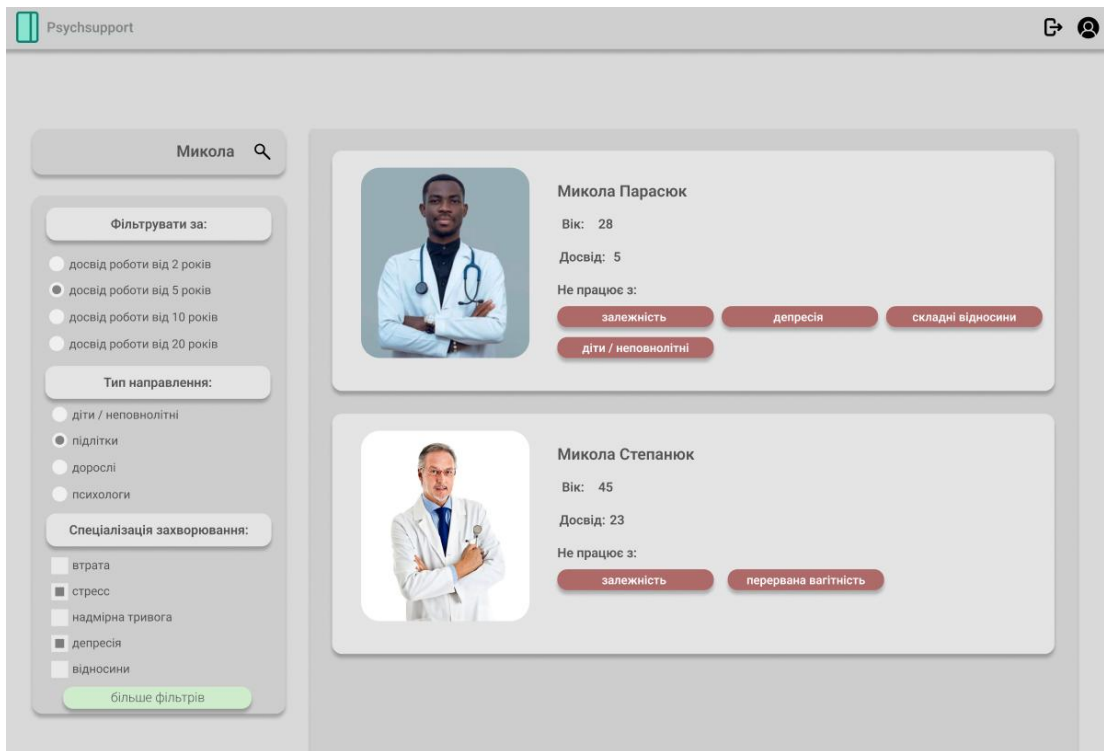


Рис. 4.4 Меню пошуку лікарів

Після перегляду наявних лікарів за фільтром, користувач має змогу обрати будь якого з них, після чого перейде на нову сторінку (Див. Рис. 4.5) з детальним описом конкретного лікаря та наявних в нього опитувань.

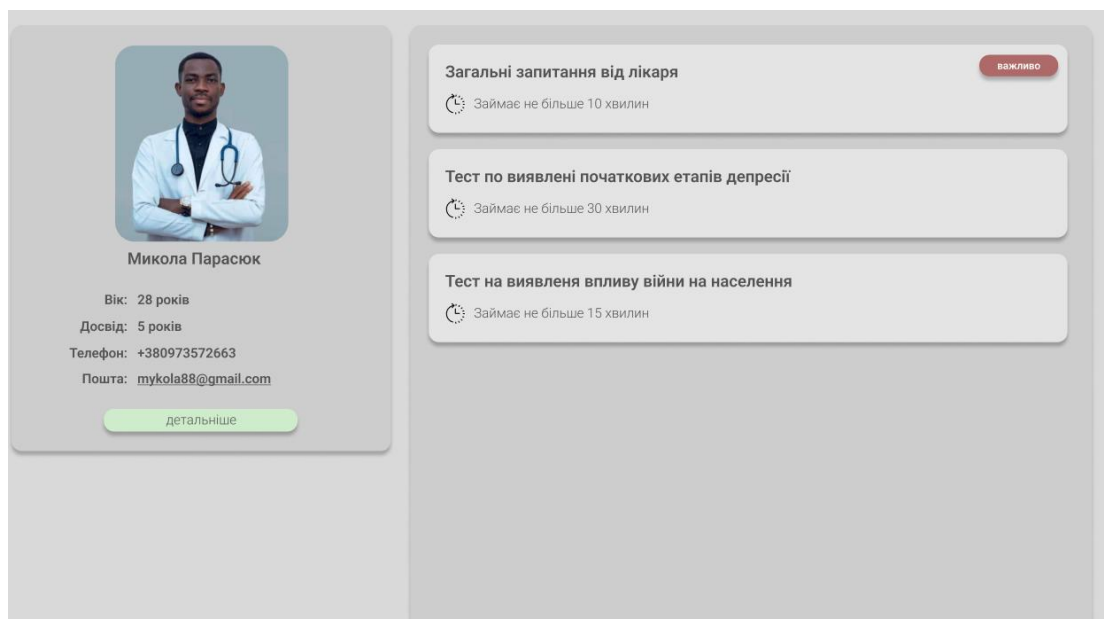


Рис. 4.5 Перегляд лікаря та його опитувань

Після перегляду користувач може обрати будь-яке опитування та пройти тест, після заповнення форми, через деякий час, користувача з'явиться у системі лікаря.

Важливо зазначити, кожен лікар завжди створює хоча б одне опитування, перш за все рекомендується зробити коротке опитування для зменшення витрат часу на комунікацію та постійного повторення однакових запитань та відповідей.

До того ж, лікар зможе переглянути подібні опитування перед сеансом, який вам назначить.

Psychsupport

Загальні запитання від лікаря

важливо

Займає не більше 10 хвилин

1. Опишіть ваш стан зараз:

2. Що вас зараз хвилює?

3. Які у вас були порушення з наведених нижче?

- втрата
- стресс
- надмірна тривога
- депресія
- відносини

Микола Парашчак

Вік: 28 років

Досвід: 5 років

Телефон: +380973572663

Пошта: mykola88@gmail.com

детальніше

важливо

Рис. 4.6 Проходження опитування

### 4.3 Покращення системи

Після представлення та аналізу прототипу було прийнято рішення розширити в подальшому систему, одним з головних факторів системи є зменшення затрат часу, тому наступним етапом буде реалізація нижче наведеного функціоналу:

- відображення вільного часу для відвідування;
- реалізація створення постів від імені лікаря;
- покращення інтерфейсу застосунку

## **Висновки**

Під час проведення побудови системи на вище розглянути розділах, автором кваліфікаційної роботи було прийнято рішення у побудові простих рішень для зменшення затрат часу на реалізацію для демонстрації прототипу та виявлення доповненого функціоналу для системи.

Окрім того, було спроектовано систему, що ділиться на декілька модулів та спрощує її розуміння. При виборі архітектури було відібрано N-tier архітектуру для реалізації. Архітектура проекту має 4 рівні та поділяється на: рівень домейну, де вносяться спільні модулі для зменшення залежності і пропорційного зростання складності коду; рівень доступу до даних описує базу даних, взаємодію з нею та конфігурацію з використанням EF; проміжний рівень між доступом до даних та представлення, що містить у собі набір сервісів; та рівень представлення, що представляє інтереси клієнта.

Отже, слідуючи архітектурі проекту та вимогам що були розглянуті раніше, було спроектовано прототип. Цей прототип було детально описано з розподіленням на два рівні користувача. Прототип продемонстрував що додаток потребує деяких нововведень, але не зважаючи на це він має гарний дизайн та достатньо функціональну систему.

## ВИСНОВКИ

Головна сутність людини до покращення свого життя та тенденції до автоматизації всього навколо неї призвела до швидкої цифровізації всіх продуктів, які так чи інакше використовує людина.

Оскільки ми живемо в час цифровізації ця тематика вважається актуальною на декілька десятків років, а то і більше. І зважаючи на те що кожна людина так чи інакше пов'язана з психологією, це може бути як допомога близьким чи психологічна допомога собі, цифровізація цієї системи як ніколи є актуальною.

В першому розділі було проведено великий аналіз спектру ринку та виділено основні проблеми аналогів для їх подальшого виправлення.

Ціль другого розділу полягала в дослідженні специфіки конкретної галузі та її проблематики. До того ж було описано основні вимоги до побудови продукту, на основі яких були побудовані наступні розділи.

В третьому розділі розглянуто сучасні прототипи деяких систем. Окрім того відзначається важливість проектування гарної системи з врахуванням потреб ринку та бізнесу. Окрім того, було продемонстровано найбільш впливові компанії що використовують ті чи інші архітектурні підходи, що допоможе системі одразу сфокусуватися на тенденціях ринку та бути конкурентоспроможними.

В останньому розділі було спроектовано прототип. Цей прототип було детально описано з розподіленням на два рівні користувача. Прототип продемонстрував що додаток потребує деяких нововведень, але не зважаючи на це він має гарний дизайн та достатньо функціональну систему.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Програмний засіб «Pleso» [Електронний ресурс] // Pleso Inc. – 2023. – Режим доступу до ресурсу: <https://pleso.me/>
2. Програмний засіб «Розкажи мені» [Електронний ресурс] // TellMe. – 2023. – Режим доступу до ресурсу: <https://tellme.com.ua/>
3. Програмний засіб «UA Mental Help» [Електронний ресурс] // UA Mental Help. – 2023. – Режим доступу до ресурсу: <https://uamentalhelp.org/>
4. Програмний засіб «Хаб стійкості» [Електронний ресурс] // Resiliencehub. – 2023. – Режим доступу до ресурсу: <https://resiliencehub.com.ua/catalog>
5. Роль психології в сучасному суспільстві [Електронний ресурс] // Alexis. – 2023. – Режим доступу до ресурсу: <https://alexus.com.ua/zavdannya-psixologi%D1%97-yak-nauki-rol-psixologi%D1%97-v-suchasnomu-suspilstvi/>
6. The Colorful History of Pills Can Fill Many a Tablet [Електронний ресурс] // Los Angeles Times – 2023. – Режим доступу до ресурсу <https://www.latimes.com/archives/la-xpm-2002-mar-25-he-booster25-story.html>
7. Детальніше про історію фемінізму [Електронний ресурс] // Wikipedia – 2023. – Режим доступу до ресурсу [https://uk.wikipedia.org/wiki/%D0%86%D1%81%D1%82%D0%BE%D1%80%D1%96%D1%8F\\_%D1%84%D0%B5%D0%BC%D1%96%D0%BD%D1%96%D0%B7%D0%BC%D1%83](https://uk.wikipedia.org/wiki/%D0%86%D1%81%D1%82%D0%BE%D1%80%D1%96%D1%8F_%D1%84%D0%B5%D0%BC%D1%96%D0%BD%D1%96%D0%B7%D0%BC%D1%83)
8. Психологічне здоров'я та ставлення до нього [Електронний ресурс] // Gradus – 2022. – Режим доступу до ресурсу [https://gradus.app/documents/307/Gradus\\_Research\\_\\_\\_Mental\\_Health\\_Report\\_full\\_version.pdf](https://gradus.app/documents/307/Gradus_Research___Mental_Health_Report_full_version.pdf)
9. Програмний засіб «PORUCH» у розділі «підлітки» [Електронний ресурс] // Poruch. – 2023. – Режим доступу до ресурсу: <https://poruch.me/teenagers>
10. Програмний засіб «PORUCH» у розділі «батьки» [Електронний ресурс] // Poruch. – 2023. – Режим доступу до ресурсу: <https://poruch.me/parents>

11. Multitier architecture [Електронний ресурс] // Wikipedia. – 2023. – Режим доступу до ресурсу: [https://en.wikipedia.org/wiki/Multitier\\_architecture](https://en.wikipedia.org/wiki/Multitier_architecture)
12. Програмний засіб «PORUCH» у розділі «психологи» [Електронний ресурс] // Poruch. – 2023. – Режим доступу до ресурсу: <https://poruch.me/psychologist>
13. MVC, MVP, MVVM: Which One to Choose? [Електронний ресурс] // Makeuseof Inc. – 2023. – Режим доступу до ресурсу: <https://www.makeuseof.com/mvc-mvp-mvvm-which-choose/>
14. Difference Between MVC, MVP and MVVM Architecture Pattern in Android [Електронний ресурс] // Geeksforgeeks. – 2023. – Режим доступу до ресурсу: <https://www.geeksforgeeks.org/difference-between-mvc-mvp-and-mvvm-architecture-pattern-in-android/>
15. MVP vs MVVM - Why? [Електронний ресурс] // Stackoverflow. – 2023. – Режим доступу до ресурсу: <https://stackoverflow.com/questions/1685921/mvp-vs-mvvm-why>
16. About HTTP cookie [Електронний ресурс] // Wikipedia. – 2023. – Режим доступу до ресурсу: [https://en.wikipedia.org/wiki/HTTP\\_cookie](https://en.wikipedia.org/wiki/HTTP_cookie)
17. About Cookie [Електронний ресурс] // Wikipedia. – 2023. – Режим доступу до ресурсу: <https://ru.wikipedia.org/wiki/Cookie>
18. How do you use HTTP cookies, sessions, and tokens for web authentication and authorization? [Електронний ресурс] // LinkedIn. – 2023. – Режим доступу до ресурсу: <https://www.linkedin.com/advice/0/how-do-you-use-http-cookies-sessions-tokens-web>
19. Token Authentication vs. Cookies [Електронний ресурс] // Stackoverflow. – 2023. – Режим доступу до ресурсу: <https://stackoverflow.com/questions/17000835/token-authentication-vs-cookies>
20. Cookie vs. Token Authentication [Електронний ресурс] // Algodaily. – 2023. – Режим доступу до ресурсу: <https://algodaily.com/lessons/cookie-vs-token-authentication>
21. How to decide which architecture fits better the application requirements [Електронний ресурс] // Stackoverflow. – 2023. – Режим доступу до ресурсу:

<https://stackoverflow.com/questions/51224607/android-how-to-decide-which-architecture-fits-better-the-application-requireme>

22. Stack Overflow Architecture [Електронний ресурс] // Medium. – 2023. – Режим доступу до ресурсу: <https://medium.com/@saddy.devs/stack-overflow-architecture-52cfdec5e018>
23. What are microservices? [Електронний ресурс] // Chris Richardson. – 2023. – Режим доступу до ресурсу: <https://microservices.io/>
24. Mental health in the EU [Електронний ресурс] // Europarl Inc. – 2023. – Режим доступу до ресурсу: [https://www.europarl.europa.eu/RegData/etudes/BRIE/2023/751416/EPRS\\_BRI\(2023\)751416\\_EN.pdf](https://www.europarl.europa.eu/RegData/etudes/BRIE/2023/751416/EPRS_BRI(2023)751416_EN.pdf)
25. The Origins of Psychology [Електронний ресурс] // Dotdash Media, Inc . – 2023. – Режим доступу до ресурсу: <https://www.verywellmind.com/a-brief-history-of-psychology-through-the-years-2795245>
26. Психологія. Основні поняття [Електронний ресурс] // Osvita. – 2023. – Режим доступу до ресурсу: <https://osvita.ua/vnz/reports/psychology/9705/>
27. ІСТОРИЯ ПСИХОЛОГІЇ: від античності до початку ХХ століття [Електронний ресурс] // Київ Інтерсервіс – 2016. – Режим доступу до ресурсу: [https://elibrary.kubg.edu.ua/id/eprint/16139/1/O\\_Kokhanova\\_IP\\_IL.pdf](https://elibrary.kubg.edu.ua/id/eprint/16139/1/O_Kokhanova_IP_IL.pdf)
28. Pro Git Second Edition. United States: Apress. // Scott, Chacon; Straub, Ben - 2014. p. 14.
29. Rochkind, Marc J. "The Source Code Control System". // IEEE Transactions on Software Engineering - 1975. SE-1 (4): 364–370.
30. What is Git? [Електронний ресурс] // Atlassian Inc. – 2023. – Режим доступу до ресурсу: <https://www.atlassian.com/git/tutorials/what-is-git>
31. What is service-oriented architecture? [Електронний ресурс] // Amazon. – 2023. – Режим доступу до ресурсу: <https://aws.amazon.com/what-is/service-oriented-architecture/>
32. Microservices vs. monolithic architecture [Електронний ресурс] // Atlassian Inc. – 2023. – Режим доступу до ресурсу:



<https://www.atlassian.com/microservices/microservices-architecture/microservices-vs-monolith>