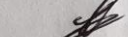


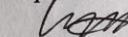
Вінницький національний технічний університет  
Факультет інформаційних технологій та комп'ютерної інженерії  
Кафедра обчислювальної техніки

**МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА**  
на тему:  
**ВПОРЯДКОВАНЕ МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ДОСТУПУ  
ДО ЛОКАЛЬНИХ СЛУЖБ ПОРЯТUNKУ**

Виконав студент 2 курсу, групи 1КІ-22м  
спеціальності 123 — Комп'ютерна інженерія

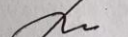
 Ярослав І.Р.

Керівник к.т.н., доц. каф. ОТ

 Обертюх М.Р.

" " 2023 р.

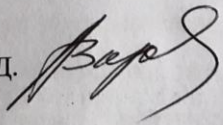
Опонент к.т.н., доц. каф. ПЗ

 Ракитянська Г. Б.

" " 2023 р.

Допущено до захисту

Зав. каф. ОТ

д.т.н., проф. Азаров О.Д. 

" 21 " 12 2023 р.

ВНТУ 2023

## ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

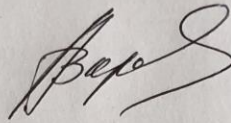
Факультет інформаційних технологій та комп'ютерної інженерії  
Кафедра обчислювальної техніки  
Галузь знань — Інформаційні технології  
Освітній рівень — магістр  
Спеціальність — 123 Комп'ютерна інженерія  
Освітня програма — Комп'ютерна інженерія

ЗАТВЕРДЖУЮ

Завідувач кафедри ОТ

д.т.н., проф., О.Д. Азаров

«26» вересня 2023 р.



### ЗАВДАННЯ

#### НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ

студенту Ярославичу Ігорю Руслановичу

1 Тема роботи «Впорядковане методичне забезпечення для доступу до локальних служб порятунку», керівник роботи Обертюх Максим Романович д.ф., ст. викл., затверджено наказом вищого навчального закладу від 18.09.2023 року № 247.

2 Строк подання студентом роботи 19.12.2023.

3 Вихідні дані до роботи: дані про екстрені служби м. Вінниці, дані про обставини та ситуації, які потребують реагування.

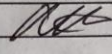
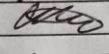
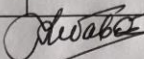
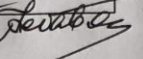
4 Зміст розрахунково-пояснювальної записки: вступ, аналітичний огляд сучасного стану локальних служб порятунку, вибір теоретичних методів розробки програмного засобу, розробка програмного засобу, тестування системи, економічний розділ, висновки.

5 Перелік графічного матеріалу: технічне завдання, схеми баз даних, лістинг програмного коду.



6 Консультанти розділів роботи приведені в таблиці 1.

Таблиця 1 — Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-4	Обертюх М.Р., к.т.н., доцент кафедри ОТ		
5	Небава М.І. к.е.н., проф. кафедри ЕПВМ		

7 Дата видачі завдання 19.09.2023.

8 Календарний план виконання МКР приведений в таблиці 2.

Таблиця 2 — Календарний план

№ з/п	Назва етапів МКР	Термін виконання	Примітка
1	Створення плану розробки	16.10.23	<i>вс</i>
2	Пошук необхідної інформації	18.10-27.10.23	<i>вс</i>
3	Аналіз існуючих методів розробки	28.10-5.11.23	<i>вс</i>
4	Розробка практичного завдання	6.11-9.11.23	<i>вс</i>
5	Проектування візуальної частини сайту	10.11-15.11.23	<i>вс</i>
6	Дослідження та тестування системи	16.11-26.11.23	<i>вс</i>
7	Розрахунок економічної частини роботи	27.11-8.12.23	<i>вс</i>
8	Аналіз виконання проекту. Висновки. Додатки	9.12-18.12.23	<i>вс</i>
9	Перевірка якості магістерської кваліфікаційної роботи та усунення недоліків	19.12.23	<i>вс</i>

Студент

Яросевич І.Р.

Керівник роботи

Обертюх М.Р.

## АНОТАЦІЯ

УДК 685.1+004.7

Яросевич І.Р. Впорядковане методичне забезпечення для доступу до локальних служб порятунку. Магістерська кваліфікаційна робота зі спеціальності 123 — Комп'ютерна інженерія. Вінниця: ВНТУ, 2023. 95 с. На укр. мові. Бібліогр.: 28 назв; рис.: 16; табл.: 10.

Магістерська кваліфікаційна робота присвячена розробці та створенню впорядкованого методичного забезпечення для доступу до локальних служб порятунку. Для визначення напрямків дослідження було розглянуто існуючі сервіси для пошуку такої інформації. Проаналізовано та оглянули сучасний стан служб порятунку, їх різновид, класифікації та можливості. Розглянуто теоретичні методами та засобами розробки баз даних та веб-застосувань. Було створено базу даних та інтерфейсу користувача, для зручного користування сайтом та для легкої взаємодії при інтуїтивно-орієнтованій обстановці проекту. Проведено тестування веб-застосунку та бази даних, а також огляд рішень що до практичної складової даного проекту. Вказано інструкції по користуванню даним сайтом та як орієнтуватись у ньому.

Ключові слова: локальні служби порятунку, бази даних, веб-застосунок.

## **ABSTRACT**

УДК 685.1+004.7

Yarosevych I.R. Streamlined methodological support for access to local rescue services. Master's thesis in the specialty 123 — Computer engineering. Vinnytsia: VNTU, 2023.95 p. In Ukrainian language. Bibliographer: 21 titles; fig.: 16; tabl.: 10.

The master's thesis is devoted to the development and creation of an orderly methodical support for access to local rescue services. To determine the directions of the research, existing services for searching for such information were considered. The current state of rescue services, their types, classifications and capabilities were analyzed and reviewed. Theoretical methods and means of developing databases and web applications are considered. A database and user interface were created for convenient use of the site and for easy interaction in an intuitively oriented project environment. Testing of the web application and database was carried out, as well as a review of solutions related to the practical component of this project. Instructions for using this site and how to navigate it are indicated..

**Keywords:** local rescue services, databases, web applications

## Зміст

ВСТУП.....	8
1 АНАЛІТИЧНИЙ ОГЛЯД СУЧАСНОГО СТАНУ ПИТАННЯ.....	11
1.1 Сучасний стан служб порятунку .....	11
1.2 Організація телефонного виклику служб екстренної допомоги .....	20
1.3 Служби екстренної допомоги міста Вінниці.....	23
2 ВИБІР ТЕОРЕТИЧНИХ МЕТОДІВ РОЗРОБКИ ПРОГРАМНОГО ЗАСОБУ	29
2.1 Математична модель швидкого пошуку інформації в структурах даних..	29
2.2 Вибір бази даних .....	32
2.3 Вибір технологій розробки .....	41
3 РОЗРОБКА ПРОГРАМНОГО ЗАСОБУ .....	49
3.1 Створення бази даних .....	49
3.2 Створення інтерфейсу користувача .....	51
4 ТЕСТУВАННЯ СИСТЕМИ .....	56
4.1 Огляд рішень для тестування веб-додатків.....	56
4.2 Інструкція по роботі з додатком.....	58
5 ЕКОНОМІЧНИЙ РОЗДІЛ.....	60
5.1 Оцінювання комерційного потенціалу розробки .....	60
5.2 Розрахунок витрат на здійснення науково-дослідницької роботи .....	65
5.2.1 Витрати на оплату праці.....	65
5.3 Розрахунок ефективності вкладених інвестицій .....	68

					08-54.МКР.022.00.000ПЗ			
Змн	Лист	№ докум.	Підпис	Дата				
Розроб.		Яросевич І.Р.			ВПОРЯДКОВАНЕ МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ДОСТУПУ ДО ЛОКАЛЬНИХ СЛУЖБ ПОРЯТУНКУ ПОЯСНЮВАЛЬНА ЗАПИСКА	Літ.		Аркушів
Перевір.		Обертох М.Р.						99
Реценз.		Ракитянська Г.Б.				ВНТУ, гр. 1КІ-22М		
Н. Контр.		Швець С.І.						
Затверд.		Азаров О.Д.						



## ВСТУП

У наш час вкрай необхідно мати змогу звернутись до тих чи інших спеціалістів за допомогою у різних ситуаціях, які за різних обставин можуть потребувати швидких і правильно прийнятих рішень, які можливо зможуть комусь допомогти або навіть врятувати. Загалом ці фахівці можуть бути у різних сферах діяльності, але швидке реагування і кваліфікаційна допомога мають бути на найвищому рівні. Тому головною складовою є створення такої візуально-простої системи, в якій інтуїтивно зможе розібратись будь-хто за необхідності, з пошуком потрібного фахівця або установи для надання професійних послуг, що може зберегти доволі важливий час при критичних ситуаціях. Особливо така система буде необхідна, для людей які кожного дня зустрічаються з небезпекою, чи то від роду їх діяльності чи то в звичайному житті, тому що завжди потрібно бути готовим до непередбачених обставин у таких випадках. Система створена саме для пошуку місцевих для необхідності служб порятунку.

У наш час пошук будь-якої інформації проводиться за допомогою мережі інтернет, у якому можна знайти все що тобі необхідно за доволі невеликих проміжок часу, що є зручним і необхідним у вирішенні якихось питань. Тобто інтернет може нести в собі як корисну мету, так і розважальну і бути доволі універсальним і необхідним ресурсом для людей.

Складність завдання полягає в тому, що крім державних служб порятунку існують і приватні, у яких можливо є свої особливості, але вони в будь-якому випадку можуть бути корисними для того кому потрібна саме конкретна служба. Потрібно звернути увагу на служби, якими користуються найчастіше для ефективного доступу до них, для їх подальшого швидкості та реагування.

Враховуючи високий ризик травм на робочому місці в обробній промисловості, компаніям, які хочуть залучати та утримувати найкращих співробітників та уникати величезних штрафів та витрат, необхідно зробити більше для створення бізнес-планів, що враховують справжню цінність безпечної робочої сили. Однак новітні інноваційні рішення можуть допомогти



виробникам на їхньому шляху до створення кращих умов праці. Звичайно завжди є підприємці, які вважають, що інвестиції в здоров'я і безпеку на робочому місці є невиправданими витратами. Чим більше робоче місце, тим більше людей вважають, що інвестиції в охорону здоров'я і безпеку приносять відчутні економічні вигоди. Це підтверджує усвідомлення потенційних наслідків недбалості.

На думку представників робочих місць, що займаються питаннями охорони праці, найбільш важливими наслідками нещасних випадків співробітників є витрати на компенсацію жертвам (53%) і вартість втраченого часу (49%). Інші витрати – це втрати іміджу (35%) і втрата доходу (34%), про які згадує кожен третій респондент. Кожен четвертий звертає увагу на втрату основних і оборотних коштів (23%). Вартість втраченого часу частіше вказується представниками середніх і великих компаній. Проте, тільки кілька компаній оцінюють, скільки коштує відсутність працівника, пов'язане з порушенням процедур охорони здоров'я і техніки безпеки.

Роблячи висновок з даної інформації, створити систему з пошуком служб порятунку є доволі потрібною і актуальною задачею.

**Метою дослідження** магістерської роботи є підвищення зучності, оперативності й швидкодії доступу до служб порятунку.

**Задачі дослідження** магістерської роботи:

- провести аналіз існуючих методів для створення системи за допомогою html та css, для легкого користування сайтом;
- провести аналіз існуючих методів для створення бази даних для швидкого пошуку служб порятунку відповідно до потреби користувача;
- створити проект, де при необхідності можна знайти служби порятунку та професійну допомогу.

**Об'єкт дослідження** магістерської роботи — процес пошуку необхідної інформації в масивах даних.

**Предмет дослідження** магістерської роботи — процес пошуку необхідних служб порятунку.

**Методи дослідження** магістерської роботи: використовувались методи про створення сайтів та пошуку служб швидкого реагування для надання допомоги, методи завдяки яким можна створити простий інтерфейс для зручного використання сайту. У роботі використано такі мови для написання сайті як html та css для реалізації створення даної системи.

**Наукова новизна отриманих результатів** магістерської роботи полягає у тому, що:

— вдосконалено метод пошуку інформації про необхідні служби порятунку, так як уся інформація про них буде знаходитись в одній системі.

**Практичне значення одержаних результатів** магістерської роботи:

- створено проект для простого та комфортного користування сайтом;
- розроблено програмне забезпечення для швидкого пошуку служб порятунку.

# 1 АНАЛІТИЧНИЙ ОГЛЯД СУЧАСНОГО СТАНУ ПИТАННЯ

## 1.1 Сучасний стан служб порятунку

Поліція (франц. *police*, від давньогрец. *πόλιτεία* — держава, місто) — відділ охорони громадського порядку. Характер завдань відрізняється від країни до країни. Транспортний засіб поліцейських вказано на рис. 1.1.



Рисунок 1.1 — Автомобіль поліції

Найбільш типовим завданням є попередження (попередження), припинення, виявлення та розкриття злочинів та інших злочинів. На міліцію також можуть бути покладені охорона різних об'єктів, безпосереднє забезпечення порядку в громадських місцях, регулювання дорожнього руху, попередження аварій, адміністративний нагляд і контроль у різних галузях, виконання рішень інших державних органів тощо. У багатьох країнах поліція відповідає за пожежну безпеку та рятувальні операції. Для вирішення цих завдань органи та посадові особи міліції мають повноваження, у тому числі щодо застосування засобів примусу (що не виключає застосування методів переконання та інших способів забезпечення впливу правових норм у своїй діяльності) [1].

У більшості країн поліція підпорядкована або є безпосередньо частиною Міністерства внутрішніх справ. Також підпорядковується поліції Міністерству юстиції та безпеки або наявності спеціального міністерства чи відомства.

У багатьох країнах поліція є цивільною структурою, але в деяких країнах, наприклад у Франції, Туреччині та Чилі, окрім власне поліції, існує жандармерія, яка спочатку була частиною збройних сил, а іноді й досі є їх частиною (наприклад, карабінери в Італії). У багатьох країнах поліція має чіткі військові риси: поліцейські мають звання, подібні до військових (наприклад, генерал поліції), іноді живуть у казармах тощо.

Чітке розмежування між поліцією та спецслужбою не завжди можливо, але суттєва різниця все ж є — поліція охороняє громадський порядок, а спецслужба — державну безпеку.

Служба поліції включає патруль, де поліцейські можуть пересуватися пішки, верхи або на транспортному засобі, уважно стежачи за тим, що відбувається. Патрульні здебільшого несуть цілодобовий режим роботи. За деяких обставин поліція виконує різноманітні завдання, включно з тими, які безпосередньо не входять до її компетенції. Наприклад, в деяких випадках він має справу з психічно неповноцінними, соціально незахищеними людьми та іншими людьми, які потребують допомоги.

Служба зобов'язана приймати та реєструвати заяви та повідомлення про вчинені або загрозливі злочини та події; прибуття на місце злочину, місце події; припинення протиправних діянь, усунення загроз безпеці громадян та громадській безпеці; документування обставин злочину, обставин події; надання допомоги (насамперед першої) особам, які постраждали від злочинів та (або) нещасних випадків.

Служба проводить кримінальне розслідування, яке ведеться переважно органами карного розшуку, регулювання дорожнього руху. ДАІ контролює та регулює дорожній рух також шляхом перевірки вживання водіями алкоголю, користування пасками безпеки тощо. Крім того, зазвичай займається розслідуванням дорожньо-транспортних пригод.

Крім того, поліція також може виконувати адміністративні завдання (наприклад, видача паспортів, різноманітних дозволів), функції кримінальної юстиції (як правило, до певного рівня), охороняти високопосадовців тощо.

Поліція організована як регіонально, так і функціонально. Її організація може бути централізованою або децентралізованою. У децентралізованих структурах, таких як Сполучені Штати, місцевий шериф (особа, обрана місцевим населенням) має широкі повноваження вирішувати, які питання слід розглядати, а які заслуговують менше уваги.

Пожежний відділ також відомий як пожежна компанія, пожежна служба, пожежна бригада, пожежно-рятувальна служба або пожежна служба в деяких регіонах — це організація, яка надає пожежну службу профілактики та пожежної безпеки та інші аварійно-рятувальні служби (рис. 1.2) [2].



Рисунок 1.2 — Пожежна служба

Пожежні відділи — це найчастіше організації державного сектору, які діють у муніципалітеті, окрузі, штаті чи спеціальному районі. Існують також приватні та спеціалізовані протипожежні організації, такі як повітряні, рятувальні та протипожежні організації.

Пожежна частина включає одну або кілька пожежних станцій у межах своїх меж і може мати пожежників, які можуть бути професійними, волонтерськими, рядовими або за викликом. У об'єднаних пожежних підрозділах працюють як професійні, так і волонтерські пожежники. У деяких країнах пожежні служби також можуть мати службу швидкої допомоги, укомплектовану волонтерами або професійним персоналом екстреної допомоги.

Пожежні частини організовані в адміністративну, службову, навчальну та оперативну системи, наприклад:

- адміністрація відповідає за нагляд, бюджети, політику та людські ресурси;
- служба забезпечує захист, безпеку та навчання населення;
- освіта готує людей знаннями та навичками для виконання своїх обов'язків;
- операції виконують завдання щодо зменшення шкоди людям, майну та навколишньому середовищу.

Пожежна частина зазвичай створюється там, де вона може мати пожежні станції, пожежні машини та інше відповідне обладнання, стратегічно розміщене по всій території, яку вона обслуговує, щоб диспетчери могли відправляти пожежні машини, пожежні машини або машини швидкої допомоги з найближчої пожежної станції. Більші відділи мають офіси для підвищення ефективності, які складаються з волонтерів, підтримки та досліджень:

- додаткову підтримку відділу в екстрених випадках надають волонтери;
- підтримує організацію ресурсів у відділі та за його межами;
- мета дослідження – надати закладу переваги в нових технологіях.

Більшість місць охоплено державними пожежними службами, які створені та фінансуються місцевими або національними органами влади. Навіть добровільні пожежні підрозділи все ще можуть отримувати державне фінансування.



Типовий розмір пожежної частини значно відрізняється від країни до країни. У Сполучених Штатах попередження пожеж зазвичай організовується на муніципальному рівні. Деякі муніципалітети належать до «районів протипожежної охорони», які обслуговуються однією пожежною службою, як-от район протипожежної охорони долини Сан-Рамон. Австрія, Німеччина та Канада також організовують пожежну службу на муніципальному рівні. У Франції пожежна служба здебільшого охоплює одне управління. У Сполученому Королівстві більшість пожежних служб охоплює один або кілька округів, тоді як у Шотландії та Північній Ірландії є одна пожежна служба. В Австралії уряди штатів керують пожежною службою, хоча три штати мають окремі агентства для столичних і сільських районів. У Польщі, Чехії, Ізраїлі, Італії та Новій Зеландії діють національні пожежно-рятувальні служби.

Пожежні підрозділи також можуть запропонувати інші екстрені послуги, такі як повітряне рятування та пожежогасіння, контроль небезпечних речовин, технічне рятування та гасіння лісових пожеж.

У деяких країнах або регіонах (наприклад, США, Німеччина, Японія, Гонконг, Макао) пожежні служби можуть відповідати за надання послуг першої допомоги. Персонал швидкої медичної допомоги може пройти перехресну підготовку як пожежники або окремий відділ техніків екстреної допомоги та парамедиків. У той час як деякі служби виконують лише функції першої служби реагування в екстрених медичних ситуаціях і стабілізують постраждалих до прибуття швидкої допомоги, інші пожежні служби також пропонують послуги швидкої допомоги.

Екстрена медична допомога (ШМД) — це медична допомога громадянам при захворюваннях, нещасних випадках, травмах, отруєннях та інших станах, що потребують невідкладної медичної допомоги [3]. Транспортний засіб швидкої допомоги зазначено на рис. 1.2.



Рисунок 1.2 — Карета швидкої допомоги

Загальні риси характеру або характеристики, які принципово відрізняють первинну медичну допомогу від інших видів медичної допомоги:

- невідкладний характер її надання у випадках першої та відстроченої невідкладної допомоги (швидка медична допомога);
- безпроблемність його надання;
- безкоштовний порядок надання першої медичної допомоги;
- діагностична невизначеність під тиском часу;
- виявив соціальну значимість.

Умови надання першої допомоги:

- поза медичною організацією (у місці виклику бригади, а також у транспортному засобі під час медичної евакуації);
- амбулаторна допомога (у закладах, де не передбачено цілодобове медичне спостереження та догляд);
- лікарня (в умовах цілодобового спостереження та догляду).

Основні функції невідкладної медичної допомоги надаються громадянам за обставин, що потребують невідкладної медичної допомоги (нешасні випадки, травми, отруєння та інші стани і захворювання). Зокрема, фельдшерські пункти (відділення) здійснюють:

— надання цілодобової своєчасної та якісної медичної допомоги хворим та постраждалим відповідно до стандартів надання медичної допомоги поза медичними закладами, у тому числі під час катастроф та стихійного лиха;

— своєчасне транспортування хворих, у тому числі з інфекційними захворюваннями, поранених і породіль, які потребують невідкладної стаціонарної допомоги (а також на вимогу медичного персоналу);

— надання медичної допомоги хворим і постраждалим, які звернулися за допомогою безпосередньо зі станції невідкладної допомоги, в амбулаторному приймальному відділенні;

— повідомляти органи охорони здоров'я міста про всі надзвичайні ситуації та нещасні випадки на території обслуговування станції швидкої медичної допомоги;

— забезпечення рівномірного укомплектування мобільних бригад першої медичної допомоги медичним персоналом на всіх змінах та повного їх забезпечення згідно з примірним переліком екіпіровки мобільної бригади першої медичної допомоги.

При цьому травмпункт може транспортувати донорську кров та її компоненти та транспортувати спеціалістів до травмпункту.

Служба невідкладної допомоги проводить науково-практичну, методичну та санітарно-просвітницьку роботу:

— центральна міська (районна або окрема обласна станція з районними (міжмуніципальними) фельдшерськими пунктами);

— відділення швидкої медичної допомоги, що включає структурний підрозділ міської центральної станції швидкої медичної допомоги та обласної або центральної лікарні (районної лікарні);

— лікарня швидкої медичної допомоги є як самостійною структурою, так і структурною частиною Центральної міської (обласної) фельдшерської станції;

— відділення невідкладної медичної допомоги — в районних, обласних та міських лікарнях.

Порятунок складається з дій швидкого реагування, які зазвичай передбачають порятунок життів або негайне лікування травм після нещасного випадку чи надзвичайної ситуації.

Використовувані інструменти можуть включати пошуково-рятувальних собак, верхових пошуково-рятувальних коней, гелікоптери, «щелепи життя» та інші гідравлічні ріжучі та розповсюджувальні інструменти, які використовуються для вилучення людей із розбитих транспортних засобів. Рятувальні операції іноді супроводжуються рятувальними машинами аварійно-рятувальних підрозділів.

Порятунок — це потужна тема в людській психології як від смертельних, так і від моральних небезпек, і вона часто згадується в художній літературі, де порятунок дівчини в біді є визначним тропом. Психолог Зигмунд Фрейд представив концепцію «фантазій порятунку» чоловіків, які переслідують «занепалих жінок» у своїй праці 1910 року «Особливий тип вибору об'єкта чоловіків, зробленого чоловіками». Погляд Фрейда на цей аспект чоловічої психології може бути заслуженим, хоча запропонований ним комплекс Едіпа, який використовувався для формулювання цієї концепції, більше не в моді. У практиці психоаналізу цей термін набув додаткового значення через бажання терапевтів «врятувати» своїх клієнтів.

Історично порятунок може означати конфіскацію активів через несплачений борг. Наприклад, земляк, який проживає неподалік від теперішнього Вітеншоу, був притягнутий до відповідальності в місцевому суді за «порятунок» свині, яка була конфіскована за несплату боргів.

Для попередження та ліквідації надзвичайних ситуацій можливе створення трьох видів рятувальних служб і підрозділів: професійні служби, професійні підрозділи; нештатні утворення; громадські підрозділи. Для вирішення завдань цивільного захисту використовуються аварійно-рятувальні служби та аварійно-рятувальні підрозділи або нештатні формування для забезпечення виконання заходів цивільного захисту.

Професійні підрозділи аварійно-рятувальні служби та збори створюються в організаціях, якщо їх присутність є необхідною законом залежно від виду діяльності організації.

Необхідність аварійної готовності до ліквідації наслідків промислових аварій підтверджується міжнародною угодою «Конвенція про транскордонний вплив промислових аварій». Угодою передбачено, що заходи щодо забезпечення готовності промислової території будуть здійснюватися власником (оператором). Законодавство передбачає, що організація, яка експлуатує небезпечний виробничий об'єкт, повинна укласти договори на технічне обслуговування з професійною аварійно-рятувальною службою (групами), щоб бути готовою до заходів з локалізації та ліквідації наслідків аварії. оскільки відповідно до закону створюють власні професійні служби порятунку — аварійно-рятувальні служби (чи збори).

Органи місцевого самоврядування самостійно фінансують заходи щодо захисту від надзвичайних ситуацій в межах муніципального утворення і лише при недостатності муніципальних сил і засобів залучають сили і засоби регіону.

Регіональні органи виконавчої влади або підпорядковані їм організації можуть створювати відомчі рятувальні служби.

Нештатні підрозділи — основа формування з'єднань не дозволяє віднести їх до сил постійної готовності, оскільки учасники з'єднання в штатному режимі роботи (відсутність надзвичайних ситуацій, аварій та небезпечних ситуацій) зайняті виконанням своїх завдань, за основними місцями роботи і тимчасово припиняти цю роботу лише для ліквідації надзвичайних ситуацій (аварій, небезпечних ситуацій) у разі їх виникнення. Вимоги щодо професійних об'єднань на них не поширюються. Вони не повинні бути на виклику 24/7; не залучаються до ліквідації надзвичайних ситуацій поза зоною своєї відповідальності.

Нештатні утворення можуть створюватися для різних цілей:

— ліквідація надзвичайних ситуацій відповідно до планів запобігання та реагування;

— вирішувати завдання цивільного захисту відповідно до планів цивільного захисту та цивільного захисту;

— вирішувати інші проблеми залежно від місцевих умов.

Під час ліквідації надзвичайних ситуацій неособовий склад цивільного захисту може виконувати невідкладні роботи, не пов'язані із загрозою життю та здоров'ю людей.

## 1.2 Організація телефонного виклику служб екстреної допомоги

Номер екстреної допомоги — це номер, який дозволяє абоненту зв'язатися з місцевою екстреною службою для отримання допомоги. Номер екстреної служби залежить від країни, зазвичай це трьохзначний номер, щоб його можна було легко запам'ятати та швидко набрати. Деякі країни мають різні номери екстрених служб для кожної служби екстреної допомоги. Вони часто відрізняються лише останньою цифрою [1-3].

У багатьох країнах номер 1-1-2 (використовується в Європі та частинах Азії, Африки та Південної Америки) або 9-1-1 (використовується в деяких частинах Америки) з'єднує абонентів із екстреними службами. Перегляньте список екстрених номерів для окремих країн.

Номер телефону екстреної служби є особливим випадком у тарифному плані країни. Раніше дзвінки на екстрений номер часто направлялися через спеціальні спеціальні канали. Хоча ці дзвінки зараз часто плутають із звичайним телефонним трафіком через електронні комутатори, вони все одно можуть досягати каналів, яких інший трафік не може. Часто система налаштована так, що коли телефонують на екстрений номер, на нього потрібно відповідати. Якщо абонент кладе трубку, лінію можна утримувати, доки служба екстреної допомоги не відповість і не розблокує виклик.

За номером служби екстреної допомоги може відповісти як телефоніст, так і диспетчер екстреної служби. Потім визначається характер надзвичайної ситуації (поліція, пожежна служба, медична служба, берегова охорона). Якщо телефонний оператор відповів на дзвінок, він зв'яже дзвінок з відповідною



екстреною службою, яка надішле відповідну допомогу. Якщо під час дзвінка потрібні декілька послуг, необхідно визначити найбільш нагальну потребу, за потреби викликати інші служби.

Центри екстреної допомоги навчені керувати викликом, щоб вони могли надати допомогу належним чином, їм можуть допомогти системи обробки викликів з комп'ютером (САСН). Оператор екстреної служби може вважати за потрібне надати термінову консультацію в ситуаціях, що загрожують життю. Деякі працівники мають спеціальну підготовку, щоб розповісти людям, як надавати першу допомогу або серцево-легеневу реанімацію.

У багатьох частинах світу екстрені служби можуть визначити номер телефону, з якого було здійснено дзвінок. Зазвичай це робиться за допомогою системи виставлення рахунків за дзвінки телефонної компанії, тому номер також бачать користувачі, які не мають номерів у списку або блокують ідентифікатор абонента. Удосконалені номери екстрених служб і подібні системи, такі як E112, можуть визначати місцезнаходження абонентів зі стаціонарних телефонів, шукаючи фізичну адресу в базі даних, а абоненти з мобільних телефонів використовують триангуляцію, отриману з веж або GPS пристрою. Це часто спеціально обумовлюється в законі країни про телекомунікації.

Мобільні телефони можна використовувати в країнах з різними екстреними номерами. Це означає, що мандрівнику, який відвідує іншу країну, не потрібно знати місцеві номери екстрених служб. Мобільний телефон і SIM-карта мають попередньо запрограмований список екстрених номерів. Коли користувач намагається зателефонувати за номером екстреної служби, відомим телефону GSM або 3G, виконується спеціальне налаштування екстреного виклику. Фактичний номер навіть не пересилається в мережу, але мережа направляє екстрений виклик на місцевий номер екстреної служби. Більшість мобільних телефонів стандарту GSM можуть дзвонити на екстрені номери, навіть якщо клавіатуру телефону заблоковано, у телефоні немає SIM-карти,

номер екстреної служби введено замість PIN-коду або немає сигналу мережі (мережа зайнята).

Більшість мобільних телефонів GSM мають 112, 999 і 911 як попередньо запрограмовані номери екстрених служб, які завжди доступні. SIM-карта, видана оператором, може містити додаткові номери екстрених служб для певної країни, які також можна використовувати під час поїздок за кордон. Мережа GSM також може оновлювати список відомих екстрених номерів, коли телефон реєструється в ній.

Використання номера екстреної служби, який розпізнається телефоном GSM, наприклад 112, замість іншого номера екстреної служби може бути вигідним, оскільки телефони та мережі GSM надають пріоритет екстреним викликам. Телефон, який дзвонить на номер екстреної служби, який він не розпізнає, може відмовитися від роумінгу до іншої мережі, що може призвести до проблем, якщо немає доступу до домашньої мережі. Набір відомого екстреного номера, наприклад 112, змушує телефон спробувати зателефонувати з будь-якої доступної мережі.

У деяких мережах GSM-телефон без SIM-карти можна використовувати для здійснення екстрених дзвінків, а більшість телефонів GSM підтримує більший список номерів екстреної допомоги без SIM-карти, наприклад 112, 911, 118, 119, 000, 110, 08 і 999. Проте деякі мережі GSM не приймають екстрені дзвінки з телефонів без SIM-картки або навіть вимагають SIM-картку з кредитом. Наприклад, латиноамериканські мережі зазвичай не дозволяють екстрені виклики без SIM-карти, а також британські мережі через поширеність шахрайських викликів.

Телефони GSM можуть зберігати деякі одно- або двозначні телефонні номери як коди спеціальних послуг. Екстрений дзвінок на такі номери, як 03 або 92, може бути неможливим на мобільному телефоні. У цих випадках на номер екстреної служби потрібно телефонувати зі стаціонарного телефону або за допомогою додаткового першого/останнього номера (наприклад, 922 або 992 замість 92 і 003 або 033 замість 03).

У Сполучених Штатах FCC вимагає, щоб мережі направляли всі мобільні та громадські дзвінки до центру екстреного виклику, включаючи телефони, які ніколи не використовувалися або термін служби яких закінчився [25][26]. У результаті існують програми, які надають безкоштовні використані мобільні телефони жертвам домашнього насильства та іншим особам, які, ймовірно, потребуватимуть екстреної допомоги. Станом на 2020 рік оператори 911 змогли краще визначати місцезнаходження абонентів, які дзвонять у 911 зі своїх мобільних телефонів у приміщенні, оскільки індустрія бездротового зв'язку США покращила місцезнаходження абонентів для більшості таких дзвінків. «Покращена точність визначення місцезнаходження», доступна для мереж підтримки та телефонів, може визначати місцезнаходження абонентів через сусідні пристрої Wi-Fi або Bluetooth, зареєстровані в певному місці в спеціальній базі даних екстрених служб.

Мобільні телефони створюють додаткові проблеми для операторів служби екстреної допомоги, оскільки багато телефонів дозволяють викликати екстрені номери навіть із заблокованою клавіатурою. Оскільки мобільні телефони зазвичай носять у кишенях і маленьких сумках, клавіші можна легко випадково натиснути, що призведе до випадкових дзвінків. У Великій Британії була розроблена система об'єднання дзвінків, коли абонент направляється до автоматизованої системи, що дозволяє більшій кількості операторів справлятися зі справжніми екстреними викликами.

### 1.3 Служби екстреної допомоги міста Вінниці

Пожежне забезпечення — це створена в установленому порядку група органів управління, відомств і організацій, призначена для організації запобігання, гасіння та проведення покладених на них аварійно-рятувальних робіт. Першочерговими рятувальними заходами, пов'язаними з гасінням пожежі, є заходи пожежного підрозділу з реагування на порятунок людей і майна та надання першої медичної допомоги постраждалим від пожежі. (рис.1.3, табл.1.1). Щоб звернутись до пожежних потрібно набрати номер — 101.

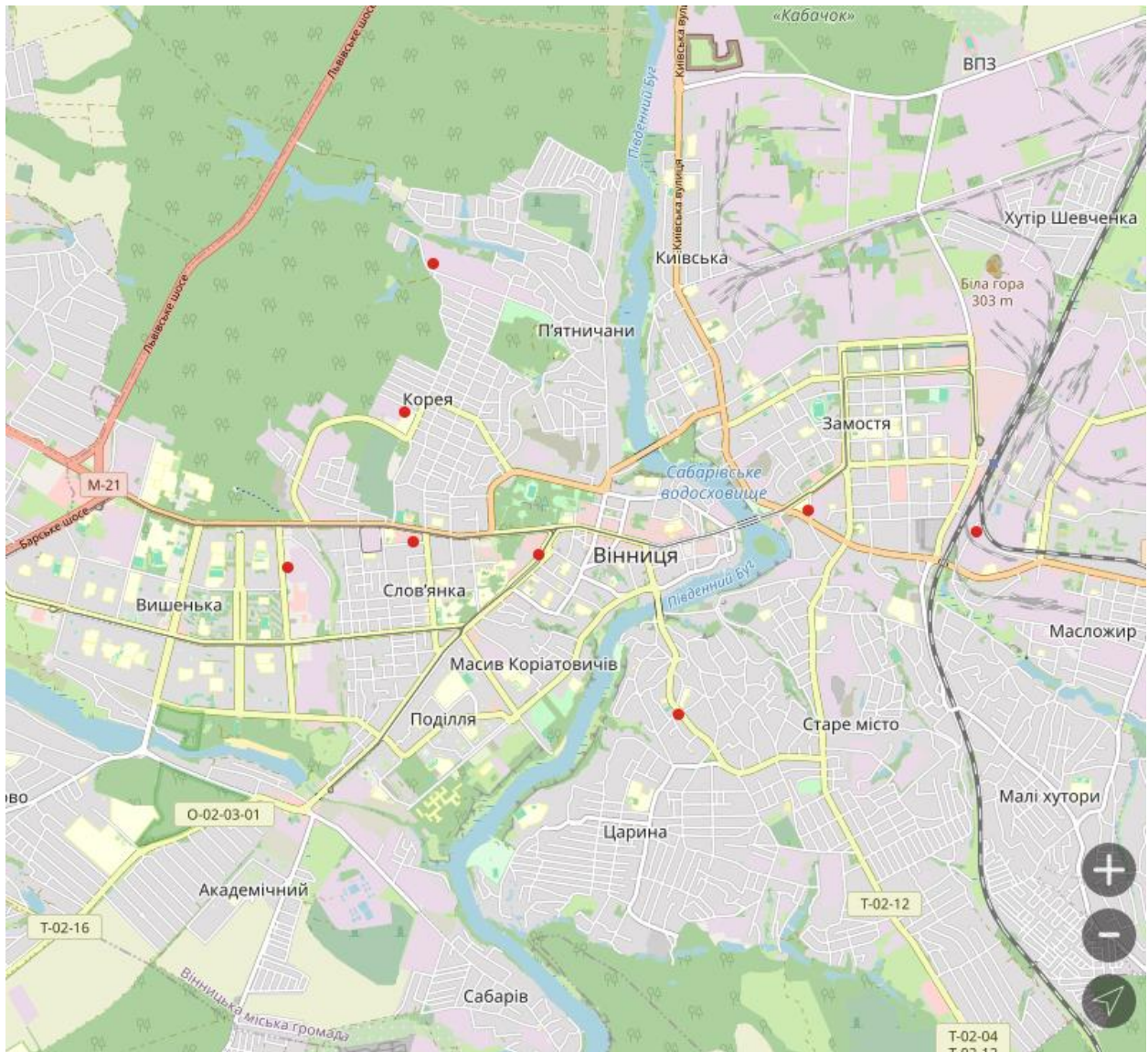


Рисунок 1.3 — Пожежні частини м. Вінниця

Таблиця 1.1 — Пожежні частини м. Вінниця

ДПРЧ 2	вулиця Нансена, 8, тел. +380(43)226-59-22 Сьогодні ця організація працює цілодобово.
Вінницьке обласне спеціалізоване ремонтно-будівельне підприємство протипожежних робіт Добровільного пожежного товариства України	вулиця Генерала Арабея, 13, тел. +380(43)266-22-67
Пожежна частина	Вінниця, вулиця 600-річчя

## Продовження таблиці 1.1

Пожежна частина №4	Вінниця, вулиця Пирогова,
Пожежна частина №3	вулиця Северина Наливайка
ДПРЧ-5	Залізничний провулок
Явір-2000	Вінниця, вулиця Дмитра Майбороди, 2, тел. +380(43)264-54-84 Сьогодні ця організація працює за графіком з 08:00 до 17:00.
Пожежна частина	Вінниця, вулиця Ботанічна

Українська міліція є озброєним органом виконавчої влади держави, який захищає життя, здоров'я, права і свободи громадян, власність, природу, інтереси суспільства і держави від протиправних посягань (рис.1.4, табл.1.2). Звернутись за допомогою до поліції, можна за номером — 102.

## Таблиця 1.2 — Поліцейські частини м. Вінниця

Вінницький відділ поліції ГУНП	вулиця Пирогова, , тел. +380(43)259-33-29
Поліція	вулиця Соборна
Головне управління Національної поліції у Вінницькій області	вул. Театральна, 10, тел. +380(43)259-33-23 Сьогодні ця організація працює цілодобово.
Ленінський РВВС	Вінниця, вулиця Пирогова.
УМВС Беркут	вулиця Ботанічна, 27, тел. +380(43)259-33-40 Сьогодні ця організація працює цілодобово.
Поліція	Хмельницьке шосе.
Управління державної служби охорони при ГУМВС України	Залізничний провулок, 6, тел. 0(800)502-220
Кінологічний центр	Вінниця, вулиця Ботанічна



## Продовження таблиці 1.2

Замостянський РВ ВДУ ГУ МВС	вулиця Київська.
Лівобережний відділ поліції Вінницького відділення поліції	вулиця Київська, тел. 0 (43) 259-33-30 Сьогодні ця організація працює цілодобово.
Митниця	Немирівське шосе
ЕКЦ МВС	вулиця Василя Порики.
Центр технічної охорони відділу Державної служби охорони при УМВС України	Вінниця, вулиця Пирогова, тел. 0 (43) 243-78-00.
Поліція	вулиця Максима Шимка.

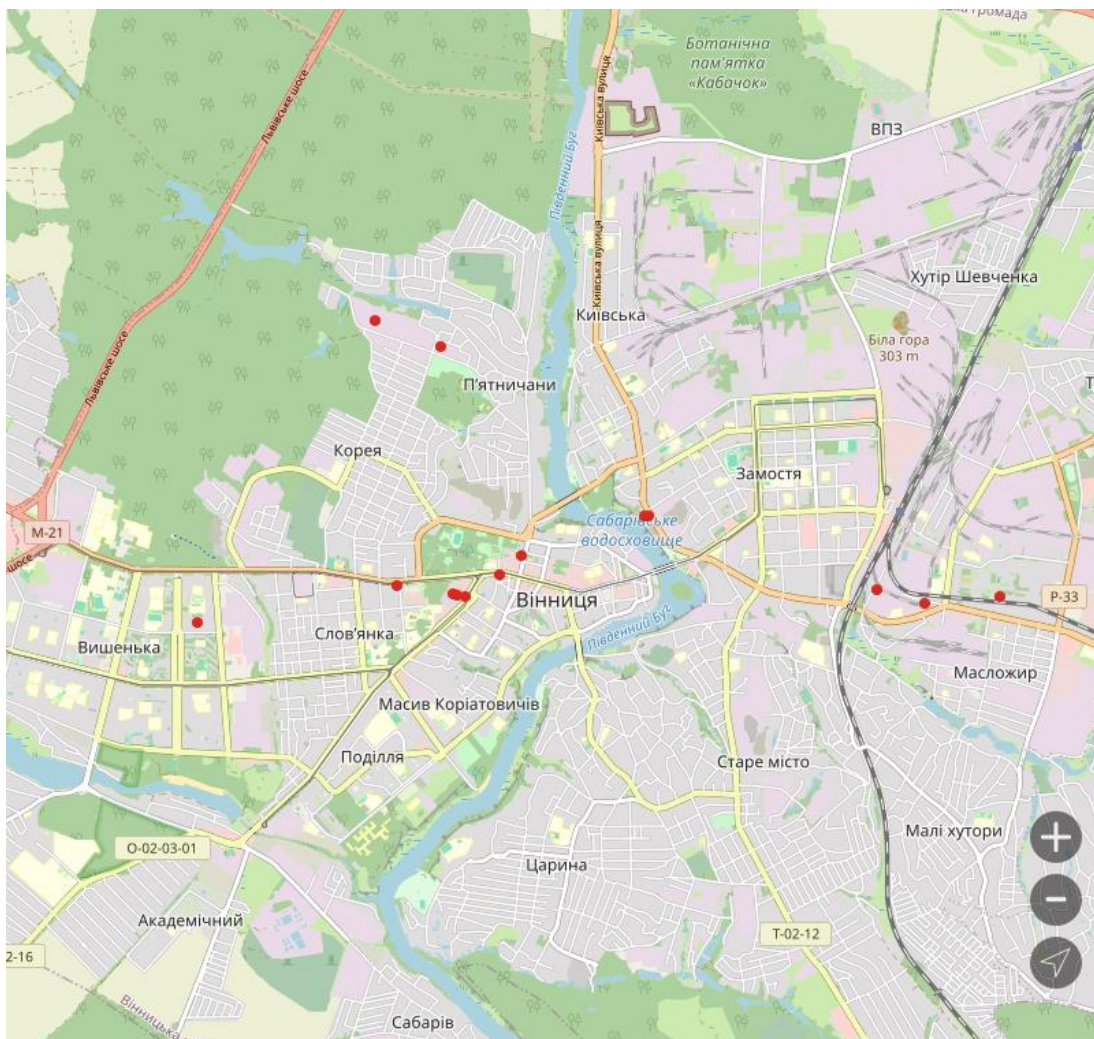


Рисунок 1.2 — Поліцейські частини м. Вінниця



Перша медична допомога (ШМД) — це система, яка організовує цілодобове надання першої медичної допомоги при загрозливих для життя станах і захворюваннях на місці пригоди та по дорозі до лікувальних закладів. Найважливішою характеристикою першої допомоги, яка відрізняє її від інших видів медичної допомоги, є швидкість її дії. Небезпечний стан виникає раптово, а його жертва, як правило, знаходиться далеко від людей, які можуть надати професійну медичну допомогу, тому лікарі повинні бути доставлені до хворого якомога швидше (табл.1.3). Звернутись за допомогою до швидкої можна за номером — 103.

Таблиця 1.3 — Перша медична допомога м. Вінниця

1. Станція швидкої медичної допомоги Вінницької обласної ради	Адреса: 21001, Вінниця, вул. Ширшова, 43 (Лівобережний) Телефон: +380 (432) 27-43-33 Режим роботи: щоденно, цілодобово
2. Комунальний заклад Територіальне медичне об'єднання Вінницький обласний центр екстреної медичної допомоги та медицини катастроф	Адреса: Вінницька область, Vinnytsia, вул. Пирогова, 46а (Правобережний) Телефон: +380 (432) 67-01-20
3. Міська клінічна лікарня швидкої медичної допомоги	Адреса: Вінниця, вул. Київська, 68 (Лівобережний) Телефон: +380 (432) 66-41-26; +380 (432) 67-07-21; +380 (432) 66-41-10 Режим роботи: щоденно, цілодобово
4. Травмпункт Вінницької міської клінічної лікарні швидкої медичної допомоги	Адреса: Вінницька область, Vinnytsia, вул. Київська, 68а (Лівобережний) Телефон: +380 (432) 66-41-10 Режим роботи: щоденно, цілодобово

Інші екстерні служби м. Вінниця вказані у табл. 1.4.

Таблиця 1.4 — Інші екстерні служби м. Вінниця

Аварійна служба Вінницяліфт	тел.: 16-40, 67-54-40
Аварійне відкриття замків	тел.: (067) 587-09-08
Аварійно-диспетчерська служба (водовідведення)	тел.: 43-33-85, 46-66-40, 43-32-65
Асинація (відкачування водойм)	тел.: 61-90-17
Водоканал (аварійна, відправник)	тел.: 67-16-17, 61-89-87
Розрахунковий центр житлово- комунального господарства вул. Єрмака, 7а	тел.: 35-10-44, 67-32-93
Газова аварійна служба	тел.: 104, 15-51, 26-12-42 (довідкова), 27-25-25 (відправник), 15-53, 26-51-82 (ремонт)
Електромережі міста (аварійна, чергова) вул. Пирогова 174	тел.: 53-60-66, 68-28-32; 65-95-44, 35-15-26 (відділ замовлення)
Горсвіт (вуличне освітлення) вул. Р. Скалецького 17	тел.: 35-57-45, 55-07-74
Гаряча лінія міськвиконкому	тел.: 59-51-11
Департамент інформації "Вінницяобленерго"	тел.: 52-50-64 (комутатор)
Морги	тел.: 67-00-68 (Лікарня Пирогова), 43-97-72 (Онкологічний диспансер) 68-03-89 (Лікарня Юценка)
Ремонт таксофонів	тел.: 15-64, 46-65-99
Сектор оперативного реагування з ліквідації надзвичайних ситуацій у місті	тел.: 35-12-20
Теплові мережі (відправник)	тел.: 51-31-70

## Продовження таблиці 1.4

Теплові мережі (замовний відділ) вул. Першотравнева 2	тел.: 63-48-87, 63-46-85
--	--------------------------

## 2 ВИБІР ТЕОРЕТИЧНИХ МЕТОДІВ РОЗРОБКИ ПРОГРАМНОГО ЗАСОБУ

### 2.1 Математична модель швидкого пошуку інформації в структурах даних

В інформатиці лінійний пошук або послідовний пошук — це метод пошуку елемента в списку. Він перевіряє кожен елемент списку один за іншим, доки не буде знайдено збіг або не буде виконано пошук по всьому списку [5].

Лінійний пошук виконується за найгірший лінійний час і робить щонайбільше  $n$  порівнянь, де  $n$  — довжина списку. Якщо для кожного елемента однакова ймовірність пошуку, лінійний пошук має середній випадок  $n+1/2$  порівнянь, але середній випадок може бути змінений, якщо ймовірності пошуку для кожного елемента відрізняються. Лінійний пошук рідко буває практичним, тому що інші алгоритми та моделі пошуку, такі як бінарний пошук і хеш-таблиці, дозволяють набагато швидше шукати всі списки, крім коротких.

Лінійний пошук перевіряє кожен елемент списку один за іншим, поки не знайде елемент, який відповідає цільовому значенню. Якщо алгоритм досягає кінця списку, пошук припиняється з помилкою.

В інформатиці двійковий пошук, також відомий як пошук напівінтервалів, логарифмічний пошук або двійкове око, — це алгоритм пошуку, який знаходить розташування цільового значення в упорядкованому масиві [4]. Двійковий пошук порівнює цільове значення із середнім елементом масиву. Якщо вони не рівні, половина, де ціль не може бути, усувається, і пошук продовжується з половиною, що залишилася, знову порівнюючи середній елемент із цільовим значенням і повторюючи це, доки не буде знайдено цільове значення. Якщо пошук завершується порожньою частиною, що залишилася, елемента немає в таблиці.

Двійковий пошук виконується за найгірший логарифмічний час і порівнює кількість елементів у масиві. Двійковий пошук швидший за лінійний, за винятком менших масивів. Однак перед використанням двійкового пошуку матрицю необхідно відсортувати. Існують спеціалізовані структури даних, розроблені для швидкого пошуку, наприклад хеш-таблиці, пошук у яких можна здійснювати ефективніше, ніж двійковий пошук. Однак двійковий пошук можна використовувати для вирішення більшої кількості проблем, таких як пошук наступного найменшого або наступного за величиною елемента в масиві відносно об'єкта, навіть якщо його в масиві немає.

Існує кілька варіантів бінарного пошуку. Зокрема, каскад дробів прискорює двійковий пошук однакового значення в кількох таблицях. Рішення дробового каскаду ефективно вирішує багато проблем пошуку в обчислювальній геометрії та багатьох інших областях. Експоненціальний пошук розширює двійковий пошук до необмеженої кількості списків. Двійкове дерево пошуку та структури даних В-дерева базуються на двійковому пошуку.

Двійковий пошук працює на відсортованих таблицях. Двійковий пошук починається з порівняння елемента в середині масиву з цільовим значенням. Якщо цільове значення відповідає елементу, повертається його позиція в масиві. Якщо цільове значення менше за елемент, пошук продовжується внизу таблиці. Якщо цільове значення перевищує елемент, пошук продовжується у верхній частині таблиці. Таким чином, алгоритм усуває половину, де цільове значення не може бути в кожній ітерації.

Пошук у глибину (DFS) — це алгоритм для обходу або пошуку структур даних дерева або графа. Алгоритм починається з кореневого вузла (вибирає довільний вузол як кореневий вузол у випадку графа) і досліджує, наскільки це можливо, кожен гілку перед поверненням. Додаткова пам'ять, як правило, стек, необхідна для запису вузлів, знайдених на даний момент із певної гілки, що допомагає з відстеженням графіка [5].

Пошук по ширині (BFS) — це алгоритм, який шукає вузол у структурі даних дерева, який відповідає певній властивості. Він починається з кореня

дерева та перевіряє всі вузли на поточній глибині перед переходом до вузлів на наступному рівні глибини. Додаткова пам'ять, зазвичай черга, потрібна для відстеження виявлених, але ще не досліджених дочірніх вузлів [6].

Наприклад, у шаховому ендшпілі шаховий механізм може побудувати дерево гри з поточної позиції, використовуючи всі можливі ходи, і використовувати пошук у ширину, щоб знайти виграшну позицію для білих. Неявні дерева (такі як ігрові дерева або інші дерева вирішення проблем) можуть мати нескінченний розмір; пошук у ширину гарантовано знайде вузол рішення, якщо він існує.

Навпаки, (стандартний) пошук у глибину, який досліджує гілку вузла настільки далеко, наскільки це можливо, перш ніж повернутися назад і розширити інші вузли, може загубитися в нескінченній гілці та ніколи не досягти вузла рішення. Ітеративний пошук у глибину дозволяє уникнути останнього недоліку дослідження верхніх частин дерева знову і знову. З іншого боку, обидва алгоритми глибини обходяться без додаткової пам'яті.

Пошук по ширині може бути узагальнений як для неорієнтованих, так і для орієнтованих графів із заданим початковим вузлом (іноді званим «ключем пошуку»). Під час пошуку штучного інтелекту в просторі станів часто допускаються повторні пошуки вершин, тоді як у теоретичному аналізі алгоритмів, спрямованих у ширину, зазвичай вживаються запобіжні заходи, щоб запобігти повторам.

У 1945 році Конрад Цузе винайшов BFS і його застосування для пошуку зв'язних компонентів графів у (відхиленій) докторській роботі. дисертацію про мову програмування Plankalkül, але вона була опублікована лише в 1972 році. Він був повторно винайдений у 1959 році Едвардом Ф. Муром, який використовував його для пошуку найкоротшого шляху в лабіринті, а пізніше був розроблений К. Й. Лі як алгоритм маршрутизації проводів (опублікований у 1961 році).

A\* (вимовляється як «A star») — це алгоритм обходу графів і пошуку шляхів, який використовується в багатьох галузях інформатики за повноту,

оптимальність і оптимальну ефективність [6]. Маючи зважений граф, вихідний вузол і цільовий вузол, алгоритм знаходить найкоротший шлях (відносно заданих ваг) від джерела до цільового.

Одним з головних недоліків практики є її складність стану, оскільки вона зберігає всі згенеровані вузли в пам'яті. Таким чином, у практичних системах маршрутизації подорожей його продуктивність зазвичай перевершує алгоритми, які можуть попередньо обробляти граф для кращої продуктивності, а також підходи з обмеженням пам'яті. Проте  $A^*$  все ще є найкращим рішенням у багатьох випадках.

Пітер Харт, Нільс Нільссон і Бертрам Рафаель зі Стенфордського науково-дослідного інституту (тепер SRI International) вперше опублікували алгоритм у 1968 році. Його можна розглядати як розширення алгоритму Дейкстри.  $A^*$  досягає кращої продуктивності, керуючи евристикою пошуку.

Порівняно з алгоритмом Дейкстри, алгоритм  $A^*$  знаходить лише найкоротший шлях від заданого джерела до заданої цілі, а не найкоротше дерево шляхів від заданого джерела до всіх можливих цілей. Це необхідний компроміс для використання евристик, орієнтованих на конкретну мету. В алгоритмі Дейкстри, оскільки створюється все дерево найкоротших шляхів, кожен вузол є ціллю, і не може бути евристик, спрямованих на конкретну мету.

## 2.2 Вибір бази даних

Вся інформація повинна десь зберігатися, будь то список співробітників компанії, меню ресторану або паролі користувачів веб-сайтів. Найчастіше для цього використовують бази даних. Існує кілька типів, і вони структурують інформацію про об'єкти по-різному. Залежно від процесу розробки продукту компанія може створити базу даних із відповідною логічною структурою та ефективно керувати нею за допомогою систем керування базами даних (СУБД) [7-14].

Бази даних дозволяють:

— швидко знаходити потрібну інформацію у великому масиві даних;

- отримувати, додавати та оновлювати інформацію без зусиль;
- визначити різні рівні доступу, наприклад, деяким співробітникам можна заборонити переглядати дані, а іншим можна дозволити їх редагувати;
- працювати одночасно всією командою, над базою можуть працювати декілька людей одночасно;
- зберігати великі обсяги даних, які інші інструменти не дозволяють.

Найпростіші бази даних можна створювати за допомогою звичайних текстових редакторів без спеціального програмного забезпечення. Головне, щоб дані мали чітку структуру, а вміст полів було легко розрізнити. Такі бази підходять, наприклад, для списків співробітників, номерів карток або програми лояльності. Переважно це невеликі обсяги даних.

Дані в текстовому форматі (txt, csv), де поля розділені пробілами, табуляцією та розділовими знаками: комами, крапками з комами та двокрапками. Найпростіші бази даних підходять, коли потрібна проста база даних, що містить 50-200 записів. Наприклад, для телефонної книги.

Переваги:

- можна редагувати стандартним текстовим редактором;
- зручна робота з конфігураційною інформацією.

Помилки:

- встановлення зв'язків між даними складно;
- не підходить для великих обсягів даних.

Приклади: etc/passwd і etc/fstab, файли csv, файли ini та інші.

Ієрархічна база даних — це система з кореневим каталогом та ієрархією підкаталогів і файлів. Усі елементи розділені між вашими предками та нащадками та тісно пов'язані між собою. Кожен запис може мати не більше одного предка, тому база даних має деревовидну структуру. Ієрархічні бази даних підходять, наприклад, для створення організаційної структури компанії (рис. 2.1).



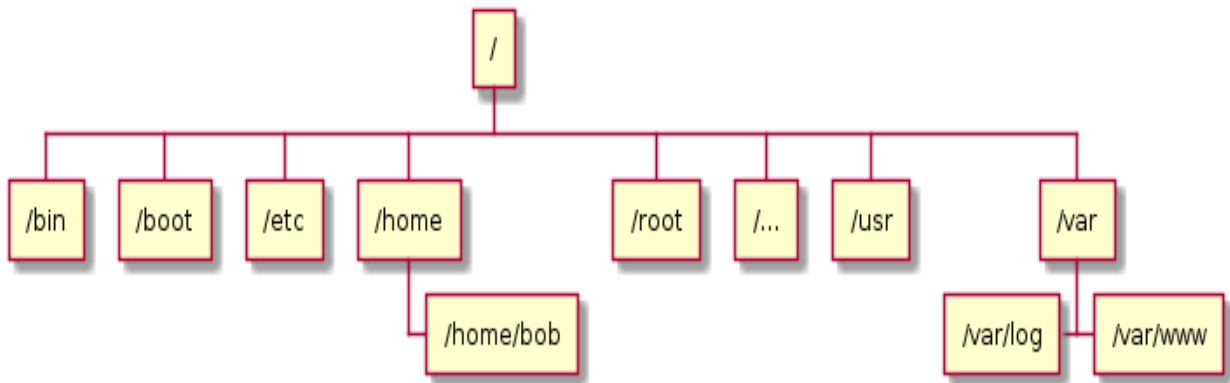


Рисунок 2.1 — Ієрархічна база даних

Переваги:

- шлях до об'єкта будується з назв кореня і підкаталогів;
- можна моделювати відносини підпорядкування.

Помилки:

- система зберігання обмежена.

Приклади: файлові системи, DNS і LDAP.

Мережеві бази даних схожі на ієрархічні бази даних, але можуть мати більше одного батька. Це дозволяє моделювати складні зв'язки між об'єктами. Таким чином упорядковуються величезні обсяги інформації в Інтернеті, де гіперпосилання з'єднують мільйони документів.

Приклад: IDMS.

У реляційній базі даних дані формуються в таблиці і мають жорстку структуру, де рядки є значеннями властивостей, а стовпці - властивостями об'єктів. Цей тип часто використовується на ранніх стадіях розробки програм і сервісів. Наприклад, при створенні меню для нового ресторану або при створенні бібліотечної бази даних, де важливо розділити об'єкти на різні категорії (рис. 2.2).



Рисунок 2.2 — Реляційні бази даних

Переваги:

- зручне відображення даних у вигляді рядків і стовпців;
- дозволяє використовувати функції мови запитів SQL;
- можливість горизонтального і вертикального масштабування даних;
- складні зв'язки дозволяють будувати бази даних графів, реляційні зв'язки дозволяють будувати лише примітивні зв'язки «батько-нащадок».

Помилки:

- низька продуктивність через складні зв'язки даних;
- сувора структура реляційних таблиць, яка не дозволяє програмістам писати довільні набори атрибутів для своїх об'єктів.

Приклади: Oracle, Microsoft SQL Server, MySQL, PostgreSQL, SQLite.

У реляційній базі даних важко зберігати й обробляти погано структуровані або неструктуровані дані. Для вирішення цих проблем існує окрема база даних — нереляційна (рис. 2.3).

key:	value
user_id:	f5badc33-5bd7-4b65-a737-b5304675f476
color:	blue
repetitions:	3
text:	hello world
data:	{ ... }

Рисунок 2.3 — Нереляційні бази даних

Дані зберігаються у вигляді таблиці з ключами та певними значеннями. Бази даних ключ-значення часто використовуються разом з іншими базами даних як механізм кешування (рис. 2.4).

Переваги:

- підходить для різних даних (файлів, тексту, чисел) з різними ключами;
- швидкий доступ до інформації завдяки зберіганню адрес;
- правила розподілу для конкретних ключів.

Помилки:

— відсутні стандартні функції бази даних: атомарність транзакцій, узгодженість даних при виконанні кількох транзакцій одночасно;

— важко підтримувати унікальні ключі, оскільки обсяги даних зростають.

Приклади: Amazon DynamoDB, Redis, Riak, LevelDB, Memcached тощо.

Дані зберігаються як документи JSON. Вони згруповані в колекції, які утворюють системи баз даних. Вміст документа може мати різні характеристики та властивості. Тому такі бази даних використовуються для створення каталогів, книг, блогів і платформ потокового відео.

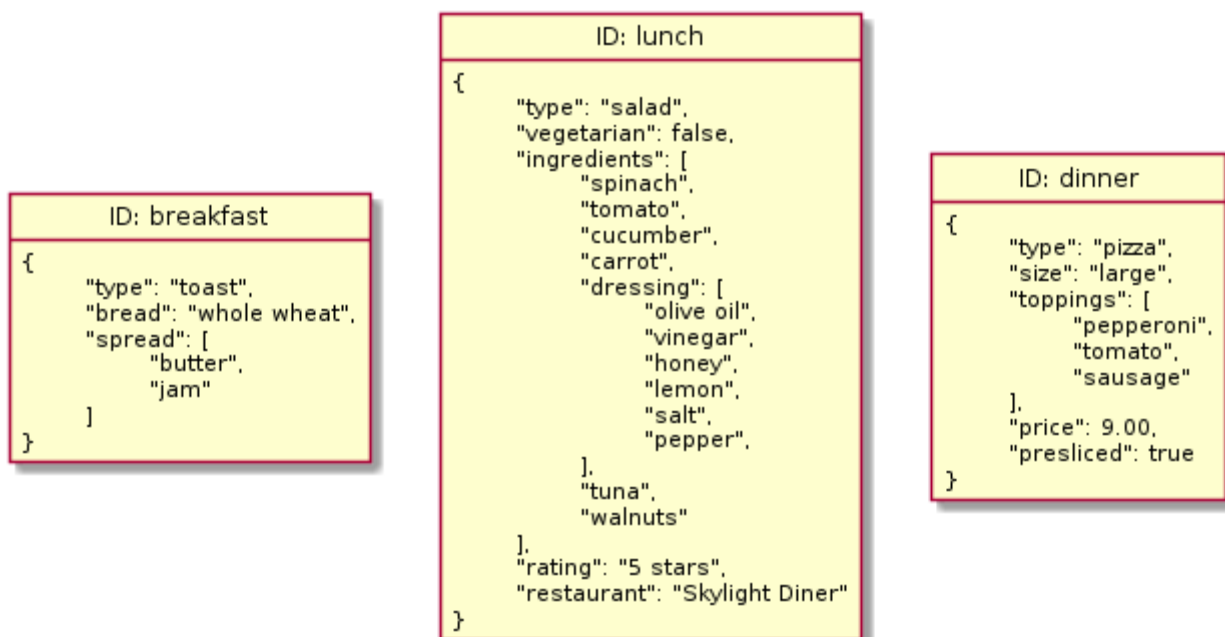


Рисунок 2.4 — Бази даних ключ-значення

### Переваги:

- підходить для розробки сервісів, які працюють з різноструктурованими даними;
- легко масштабована і змінна структура;
- швидке створення документів і мінімальне обслуговування;
- немає зовнішніх ключів, тому всі документи незалежні один від одного;
- використовує XML, JSON та інші документи для опису.

### Помилки:

- обмеження перевірки узгодженості, що знижує продуктивність бази даних;
- труднощі доступу до інформації з багатьох взаємопов'язаних джерел;
- ризик витоку конфіденційної інформації з веб-додатків.

Приклади: MongoDB, RethinkDB, CouchDB, DocumentDB.

Графові бази даних моделюють зв'язки між складними зв'язаними об'єктами. Вони базуються на топографічній структурі мережі та математичній теорії графів. Графіки представляють набори даних у вигляді вузлів, ребер і об'єктів. Такі бази даних допомагають надавати користувачам ресурсів рекомендації в реальному часі та покращують взаємодію з ними. Наприклад, музичні служби пропонують індивідуальні добірки пісень, а ринки пропонують каталоги продуктів на основі улюблених (рис. 2.5).

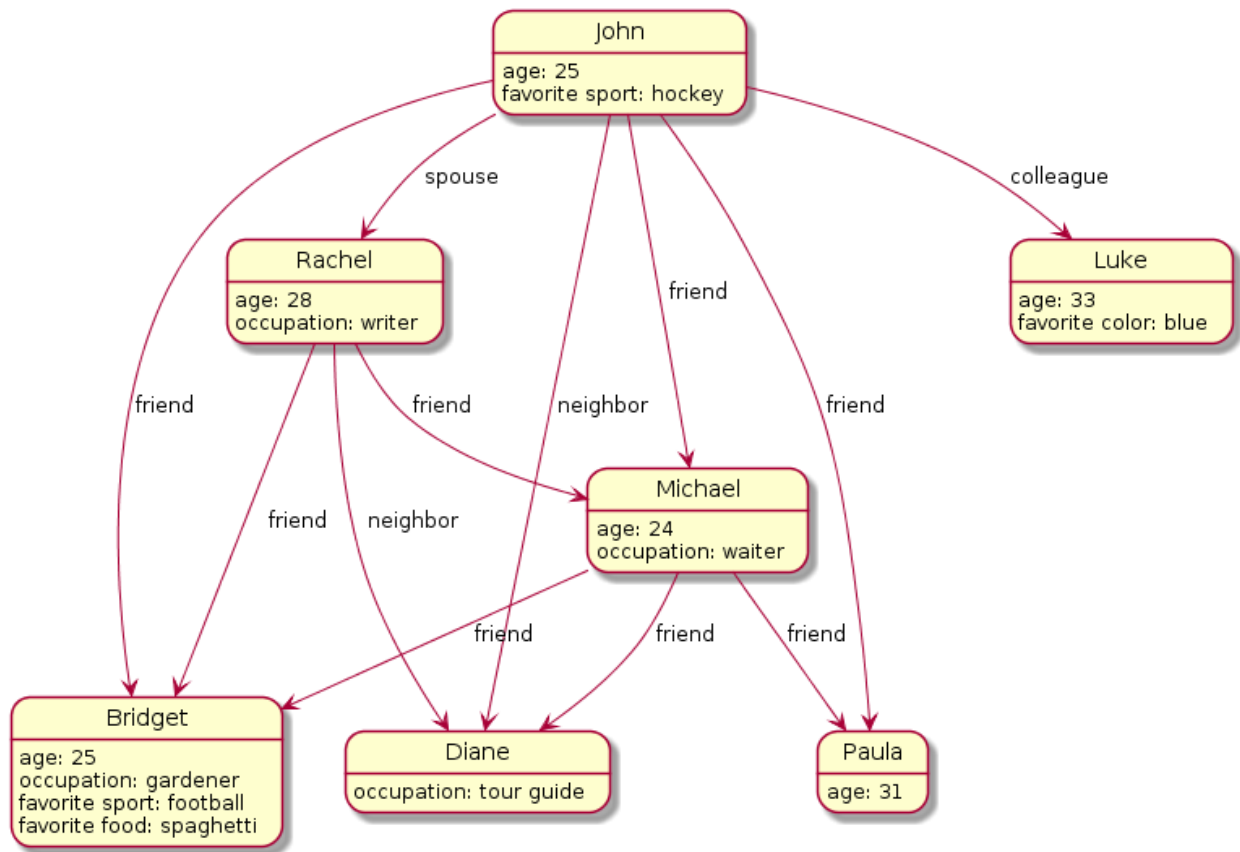


Рисунок 2.5 — Графові бази даних

**Переваги:**

- висока продуктивність і адаптивна структура;
- чіткий зв'язок між сутностями;
- результати в реальному часі.

**Помилки:**

- немає стандартизованої мови запитів;
- діаграми не підходять для подійних систем.

Приклади: Neo4J, JanusGraph, Dgraph, OrientDB.

У колонкових базах даних дані зберігаються не в таблицях, а в стовпцях. Ключі вказують формат рядка для зберігання даних об'єкта. Кожен рядок має власні властивості, тому ви можете зберігати дані з різною структурою в одному сімействі. Стовпчасті бази даних використовуються для аналізу великих обсягів даних. Наприклад, фільтрувати та сортувати товари в інтернет-

магазині або отримувати статистику з сайту за днями, категоріями, ціною тощо (рис. 2.6).

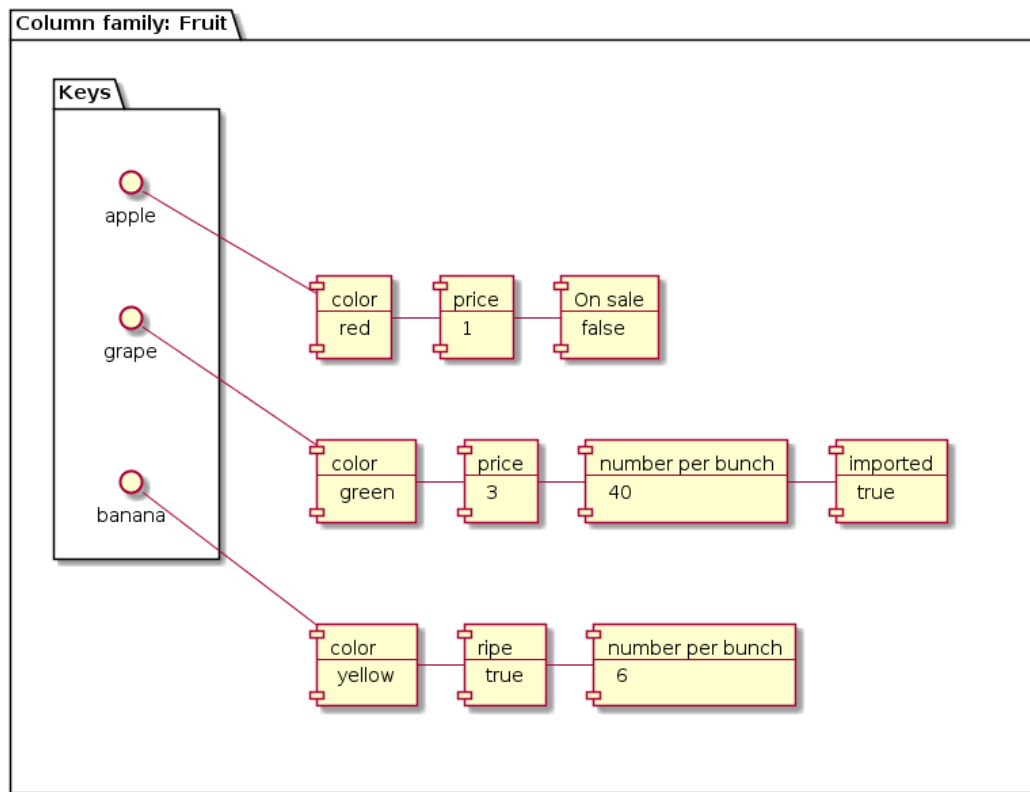


Рисунок 2.6 — Колонкові базиданих

#### Переваги:

- можливість робити складний вибір за великими столами;
- миттєва зміна структури великих таблиць;
- швидкі запити;
- легка масштабованість і трансформованість конструкції.

#### Помилки:

- запис відбувається повільно;
- не підтримує транзакції;
- обмеження для розробників.

Приклади: Cassandra, HBase, ClickHouse.

Часові ряди — це все, що можна виміряти за періоди часу: година, день, місяць або рік тощо. Метриками можуть бути різні типи даних: зміни

дорожнього руху, рівня води в річці, завантаження сервера та інші. Дані можна запитувати, будувати графіки, аналізувати та знаходити зв'язки між різними показниками. Продуктивність бази даних залежить від кількох факторів: обсягу даних, кількості запитів і показників.

Приклади: OpenTSDB, Prometheus, InfluxDB, TimescaleDB.

Гібридні типи поєднують підходи SQL і NoSQL і включають NewSQL і багатовимірні рішення. Компромісні і досить молоді рішення. Бази даних NewSQL підходять для галузей, які обробляють критичні дані: фінансовий і банківський сектори, охорона здоров'я та інші. Компанії, які цінують масштабованість і надійність транзакцій.

Переваги:

- легко масштабувати;
- висока продуктивність і доступність даних.

Помилки:

- високі вимоги до апаратних ресурсів розробників.

Приклади: MemSQL, VoltDB, Wrench тощо.

Багатовимірні бази даних поєднують кілька підходів до організації даних в єдину серверну систему для зберігання, отримання та індексування даних з кількох моделей. Такі бази даних можуть конвертувати дані з одного формату в інший. Зазвичай вони використовуються для керування ІТ-системами, централізації даних і пошуку великих даних.

Переваги:

- узгодженість даних між моделями;
- відмовостійкість завдяки сумісності з ACID;
- легка інтеграція в систему нових моделей баз даних;
- Підходить для складних проектів.

Помилки:

- складність багатовимірних баз даних;
- багатовимірна модель ще дуже молода і не має заповненої форми;
- обмежена доступність різних методів моделювання;



— не підходить для простих проектів або систем.

Приклад: ArangoDB.

## 2.3 Вибір технологій розробки

### 2.3.1 Основні поняття HTML

HTML (англ. HyperTextMarkupLanguage) — це мова розмітки гіпертексту, мова тегів, за допомогою якої пишуться гіпертекстові документи для мережі Інтернет.

HTML-документи надходять до веб-браузера з локальної пам'яті або веб-сервера і передають документи в мультимедійні веб-сторінки. Загалом він описує будову та структуру самої веб-сторінки для зовнішнього вигляду документа.

Все будується за допомогою HTML-блоків, які є елементами цього файлу. Використовуючи цю концепцію, до веб-сторінки можна додавати медіа-файли, зображення та інші об'єкти, для створення візуалізації проекту. За допомогою HTML можна створювати засоби для структурованих документів, цим самим позначаючи структуру тексту, такі як посилання, списки, абзаци, заголовки та інші елементи. Ці елементи прописуються використовуючи теги, що являє собою написання тексту всередині кутових дужок. Вони містять структурну інформацію, яка подається на сторінку. Існують теги, такі як `<img />`, `<input />`, `<p>`, які можуть включати в себе інші теги, як піделементи, а також надавати інформацію про текст документа. Браузерине відображають теги на самій сторінці, за допомогою них створюється візуалізація вмісту сторінки [15-18].

HTML передбачає засоби для:

- структурування розробки документа шляхом написання структурного вмісту тексту: заголовки, абзаци, списки, таблиці, тексти та т.п.;
- використання інформації з мережі Інтернет за допомогою гіперпосилань;
- підключення медіа контенту та інших елементів до тексту.

Документимаютьпідтримувати роботу в різних браузерах і на різних платформах. Створення сумісності необхідне для універсальності та зниженню витрати, аджеавотри можуть розробляти тільки одну версію документа.

Файл HTML використовує найповнішу кодову таблицю UCS. Для коректного відображення документів через браузер, лише однієї кодової таблиці буде замало. Тому потрібно вказувати шлях, створивши специфічну кодову таблицю для цього документа, завдяки якій можна під'єднувати інші файли, вона обов'язково прописується в елементі meta із параметром charset.

В HTML розмітка складається з чотирьох основних компонентів: декларації типу документа, символних мнемонік, базових типів даних і елементів (та їхніх атрибутів).

Загальна структура HTML-документа (рис. 2.7):

— декларація типу документа (англ. Documenttypedeclaration, Doctype), вказується на початку файлу для визначення типу документа (DTD);

— шапка документа (пишеться в параметрах елемента head), а також містить в собі додаткову інформацію про документ, яка не відображається в браузері або технічні відомості загального положення;

— тіло документа (знаходиться в параметрах елемента body), в якому знаходиться основна інформація документа.

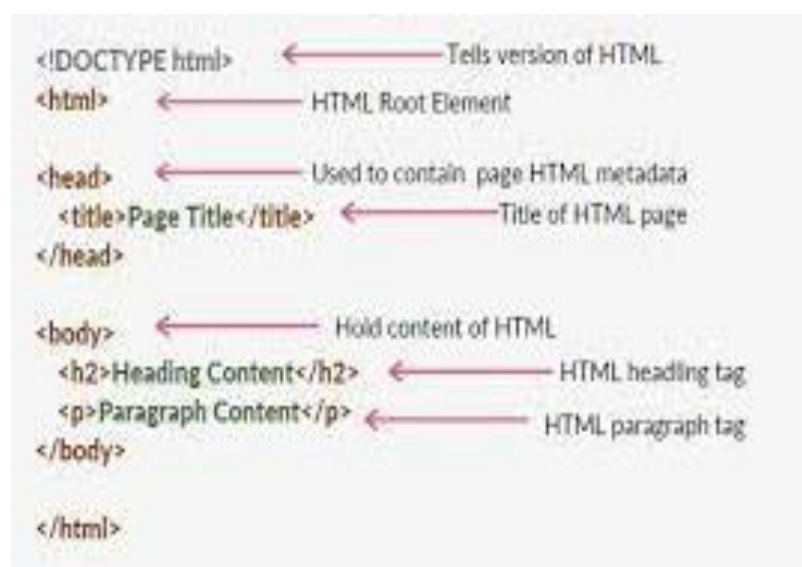


Рисунок 2.7 — Загальна структура

У елементів, які мають парний тип є початковий тег, який має вигляд `<element-name>`, та кінцевий тег, який має вигляд `</element-name>`. В початковому прописуються атрибути елемента одразу після його назви елемента, а контент елемента заповнюється між його двома тегами. Наприклад: `<element-name element-attribute="attribute-value">контент елемента</element-name>`.

В дослідницькому центрі CERN в Женеві 1989 року було представлено HTML Тімом Бернсом-Лі:

- він не мав номера своєї версії, 3 листопада 1992, вона була орієнтована лише для тексту — це була початкова версія;

- також без номера версії, 30 квітня 1993, для використання курсивного або жирного стилів написання літер, до тексту було додано атрибути та зображення;

- HTML+ (листопад 1993): все що містилось у наступних версіях, але елементи не були порізно як окрема версія, додаткові доповнення;

- HTML 2.0 (листопад 1995): за визначенням стандарту RFC 1866 версія з підтримкою форм, цей стандарт має «історичний» статус, а також попередні версії вже не є актуальними;

- Версія 3.0: завдяки випуску браузера NetscapeNavigator версії 3, цей стандарт вже був застарілим, не зазнала поширення;

- Версія 3.2 (14 січня 1997): було додано таблиці, а також обтікання текстом зображень, інтеграція аплетів і різні інші можливості;

- HTML 4.0 (18 грудня 1997): були додані скрипти, таблиці стилів та фрейми.

Також відбулось розділення на Strict (що є суворим дотриманням стандартів), Frameset (підтримка фреймів) та Transitional (перехідний). 24 квітня 1998 було випущено відкореговану версію цього стандарту:

- HTML 4.01 (24 грудня 1999): містить численні дрібні корективи, а також заміна минулої версії;

— Версія 5 (WorkingDraft, 5 квітня 2008): була створена на основі HTML 4.01 та XHTML 1.0., також модифікована і розширена специфікація DOM, а ще має новий словник;

— XHTML 1.0 (26 січня 2000): 1 серпня 2002 створено оновлення для висловлення стандарту HTML 4.01 засобами XML;

— Версія 1.1 (31 травня 2001): після того як основна версія була розділена на модулі, стандарт визначатиме версію, в якій не буде раніше запроваджених можливостей Frameset та Transitional;

— XHTML 2.0 (розробка припинена в 2010 році): ця версія вже не базується на HTML 4.01 і додає деякі нові теги, буде закінчено розділення між представленням та вмістом.

### 2.3.2 Основні поняття CSS

CSS (англ. CascadingStyleSheets) — це спеціальна мова для створення стилю веб-сторінок, що використовується для опису їхнього зовнішнього вигляду. Ця мова необхідна для користувачів та авторівбраузерів, щоб завдяки параметрам та певним елементам коду, створити основні елементи вигляду цієї сторінки. Одна з головних переваг —розділення змісту сторінки (або наповнення, контент, зазвичай HTML, XML або схожа мова розмітки) від візуального вигляду документу. Таким чином можна покращити контент в плані сприйняття і взаємодії з користувачами, а також надати адаптивності та кращого відображення контенту на веб-сторінках, зробити контент більш структурованим та простим, для звичайних користувачів.

CSS також надає можливість користуванню веб-сторінками на різних девайсах (на екрані планшету, смартфоні і т.д.). Щоб використати таблицю стилів, необхідноCSSкод зберегти у файл (це можна зробити будь-яким текстовим редактором), наприклад `example.css`, і потім імпортувати його в HTML-сторінку або прописати підключення (рис. 2.8).

```

<!DOCTYPE html>
<html>
  <head>
    .....
    <link rel="stylesheet" type="text/css" href="style.css">
  </head>
  <body>
    .....
  </body>
</html>

```

Рисунок 2.8 — Приклад включення CSS-файлу до сторінки

CSS має особливі позначення (селектори, від англ. selector) для визначення діапазону застосування правила, яке буде працювати лише для виділеної зони та видозмінювати лише ті елементи які входять у цю зону.

Використання селекторів \* (часто не пишеться, наприклад \*.p та p — рівнозначні) потрібно для підключення можливих маніпуляцій властивостей.

Завдяки такій команді, можна задати параметри усім однаковим селекторам у файлі, щоб не прописувати все для кожного окремо. Наприклад для всіх p або h2.

Для наслідування візьмемо наприклад селектор вигляду li a, для всіх a елементів, які знаходяться в li елементах (посилання всередині списків) будуть задатись якісь параметри та зміни.

Для прямого наслідування наприклад для всіх span елементів всередині p, але не глибших за перший рівень наслідування span, селектор буде p>span. Тобто інші елементи, які знаходяться всередині, не будуть корегувати та змінюватись. Наприклад .class або #id для елементів class=«class» або id=«id», які прописуються в HTML-документі, для встановлення зв'язку між файлами.

Для елементів h2, послідуочим до яких є елемент p, тоді селектор буде мати вигляд h2 + p. Отже селектор буде задіяний саме для зміни цих параметрів. Також для всіх селекторів буде input[type=«text»].

Таблиці стилів в тому чи іншому вигляді існували з зародження SGML в 1970-тих. Створення каскадних таблиць стилів були для спрощення додавання інформації про стилі для веб-сторінок. З розвитком HTML з'явилася можливість використовувати різні функціонали для додавання стилів до елементів сторінки.

У вебдизайнерів з'явилось більше можливостей для створення вигляду сайту при послідовному прогресі, але код ставав складнішим для написання та зміни. Через різницю у відображенні сторінки в різних браузерах збереження стилю сторінки було складним, користувачі мали менше контролю над відображенням контенту. Тому на розгляд було запропоновано дев'ять різних варіантів таблиць стилів від W3C. Було обрано два, після обговорення що до спеціального списку розсилки, вони створили основу для того, що стало CSS, англ. Cascading HTML StyleSheets (CHSS) та англ. Stream-based Style Sheet Proposal (SSP). Спочатку в жовтні 1994, HåkonWiumLie (зараз генеральний технічний директор OperaSoftware) запропонував CHSS, що дещо подібний до сучасного CSS. BertBos працював над браузером Argo, що використовував власний варіант таблиць стилів, Stream-based SSP. Lie і Bos почали співпрацювати для вироблення стандарту CSS (літера 'H' була виключена з назви, оскільки таблиці стилів могли застосовуватися до інших мов розмітки, не лише HTML).

CSS дозволяв використання різних таблиць стилів до документу (сторінки), на відміну від наявних тоді таблиць стилів, таких як DSSSL та FOSI. Для контролю використання, таблиці стилів могли наслідувати правила з інших (тобто створювати каскади), для спрощення роботи як користувачам сайту так і дизайнерам. Пропозиції до стандарту CSS обговорювалися на конференціях в 1994 та 1995 роках. В 1994 році було створено World Wide Web Consortium (W3C), серед інших питань W3C займався також і стилями. Робочу групу очолив StevenPemberton, як провідні технічні спеціалісти до неї входили HåkonWiumLie та BertBos. В грудні 1996 було опубліковано рівень 1 — CSS level 1 Recommendation. В квітні 2016, опубліковано CSS рівень 2 —

CascadingStyleSheetsLevel 2 Revision 2 (CSS 2.2), робота над яким ще продовжується.

### 2.3.3 Основні поняття JavaScript

JavaScript (JS) — динамічна, об'єктно-орієнтована мова програмування [9]. Найчастіше використовується для надання можливості керувати браузером для зміни структури, асинхронному обміні даними з сервером, створенні сценаріїв веб-сторінок та зовнішнього вигляду вебсторінки. JavaScript класифікують як скриптову мову програмування з динамічною типізацією. Також частково підтримує деякі відповідні архітектурні властивості, такі як: прототипне наслідування, функції як об'єкти першого класу, автоматичне керування пам'яттю, динамічна та слабка типізація, та інші парадигми програмування (імперативну та частково функціональну) [19-21]. Мова JavaScript використовується для:

- веб-сторінок при створенні сценаріїв для надання їм інтерактивності;
- застосування односторінкових веб-застосунків;
- мобільних застосунків (ReactNative, Cordova);
- сценаріїв в прикладному ПЗ (наприклад, в програмах зі складу ApacheJMeter чи AdobeCreativeSuite);
- всередині PDF-документів тощо. Ця мова, наразі, є однією з найпопулярніших мов програмування в інтернеті.

Але спочатку було скептичне ставлення зі сторони професійних програмістів, цільова аудиторія якої складалася з програмістів-любителів (рис. 2.9).



```

18  const LOCALE = globalThis.navigator.language
19
20  const div = document.body.appendChild(document.createElement('div'))
21  const list = div.appendChild(document.createElement('ol'))
22
23  const dayNames = new Map()
24
25  for (let i = 0; i < 7; ++i) {
26    const d = Temporal.PlainDate.from({
27      year: Temporal.Now.plainDateISO().year,
28      month: 1,
29      day: 1 + i,
30    })
31
32    dayNames.set(d.dayOfWeek, d.toLocaleString(LOCALE, { weekday: 'long' }))
33  }
34
35  for (const sum of [...dayNames.keys()].sort((a, b) => a - b)) {
36    list.appendChild(Object.assign(
37      document.createElement('li'),
38      { textContent: dayNames.get(sum) },
39    ))
40  }
41

```

Рисунок 2.9 — підключення JS файлу в HTML

JavaScript має C-подібний синтаксис, але в порівнянні з мовою C має такі корінні відмінності:

- об'єкти, динамічної зміни типу через механізм прототипів з можливістю інтроспекції;
- об'єкти першого класу які є функціями;
- обробка винятків;
- автоматичне приведення типів;
- автоматичне підчищення сміття;
- функції анонімного використання.

Він містить декілька вбудованих об'єктів: Object, Global, Function, String, Error, Array, Date, Number, RegExp, Boolean, Math. Крім того, у ньому є набір вбудованих операцій, які, не обов'язково є функціями або методами, а також набір вбудованих операторів, що управляють логікою виконання програм. Синтаксис JS спрощений порівняно з Java (тобто, успадкований від C), але в основному відповідає йому, щоб зробити мову сценаріїв легкою для вивчення. Так, властивості не мають типів, декларація змінної не містить її типу, а декларація функції може стояти в тексті програми після неї.

Враховуючи, що JavaScript це мова подій для веб-сторінок, то ми маємо підключити її до нашого HTML файлу (рис. 2.9).

Цей спосіб простіше для сприйняття, адже розробник шаблону зможе сфокусуватися на верстці і не вмішуватись в код програміста.

## 3 РОЗРОБКА ПРОГРАМНОГО ЗАСОБУ

### 3.1 Створення бази даних

PostgreSQL є відомою системою баз даних з відкритим кодом, який підтримує тисячі веб-сайтів, служб і програм. Це ACID-сумісна RDBMS. Інакше кажучи, він забезпечує послідовність, атомарність, ізоляцію та довговічність. Розширені функції PostgreSQL включають збережені функції, тригери, визначені користувачем функції, події та реплікація.

PostgreSQL забезпечує продуктивність, доступність і функціональність масштабованість на рівні підприємства. Тож можна розгорнути PostgreSQL у розподіленій архітектурі що може обробляти великі обсяги даних.

Іншою важливою особливістю PostgreSQL є те, що він має відкритий код, і тому він є доступним рішенням для компаній зберегти витрати на низькому рівні. Крім того, PostgreSQL має добру історію надійності та інформаційна безпека. Замовники весь час використовували його у виробництві десятиліть, і йому довіряють багато організацій і компаній у всьому світі.

PostgreSQL повністю сумісний з кількома інструментами та технології. Ви можете легко об'єднати його з багатьма програмами. Він також підтримуються сторонні інструменти, плагіни та інші мови програмування, такі як Python.

Перед тим як створювати таблиці, форми та інші об'єкти на своєму сайті, потрібно зробити базу даних (БД). Відображаються описи предметних областей. Схема внутрішньої моделі даних є непростим процесом проектування бази даних. Цей процес є серією простіших процесів для розробки не таких складних відображень. Цей порядок процесу планування постійно уточнюється і

вдосконалюється, визначити об'єкти, їхні властивості та зв'язки, які знадобляться в майбутньому користувачів системи.

Проектування бази даних — це ітеративний багатоетапний процес, у якому при аналізі моделі даних приймаються зважені рішення предметна область, потреби програмістів і користувачів, синтез фізичних і логічної структури даних, аналіз та обґрунтування вибору програмного забезпечення надання. Типи представлення даних при проектуванні бази даних: зовнішній, інформативний, логічний (логіка даних), внутрішній.

Зовнішній рівень: необхідно визначити функції керованого об'єкта, для якого створюється база даних, усі вихідні та отримані дані з точки зору визначення потреб зберігання даних. Зовнішня презентація містить рівні даних, де будується модель даних (канонічна) даних не просто сума зовнішніх інтерпретацій даних.

Інформаційний рівень — інформаційно-логічна модель (ІЛМ) теми. Область, де виключається надмірність даних і відображаються інформативні властивості керованих об'єктів незалежно від властивостей і деталей бази. Це означає, що цей рівень насамперед стосується користувачів, які розробляють або використовувати бази даних.

Логічні (концептуальні) рівні створюються з урахуванням деталей і властивостей бази даних. Цей рівень зосереджений на роботі з комп'ютерами та програмістів, які ним керують. Тут створюється концептуальна модель даних, тобто впорядкована модель предметної області, яка враховує особливості і обмеження бази даних.

Внутрішні рівні пов'язані з фізичним розміщенням даних у пам'яті комп'ютера. Ось фізична модель бази даних, яка містить дані про формати записів, їх порядок, розміщення за типом пристрою, властивості і методи використання даних. Обсяг пам'яті та час відгуку системи залежать параметри фізичної моделі.

Постановка мети налаштування елемента, інформація та основна інформація функції, які вони повинні виконувати. База даних має відповідати вимогам користувача.

Визначення таблиці при створенні таблиць слід керуватися наступним, основні принципи: дані в таблицях не повинні повторюватися, кожна таблиця містить інформацію лише по одній темі.

Визначення обов'язкових полів, тут потрібно враховувати будь-яку галузь мають бути пов'язані з темою таблиці. Додавати отримані дані до таблиці небажано під час розрахунків. Таблиця повинна містити всю необхідну інформацію, бажано постатейну найменші логічні одиниці.

Призначте індивідуальні значення для всіх полів. Визначення зв'язку між таблицями. Коли ви ділитеся даними в таблицях і щоб визначити ключові поля, необхідно вибрати типи зв'язку між столами.

Перевірка бази даних на існування дублікатів і помилок, додаючи інформацію для формування інших об'єктів бази даних. Якщо структура таблиці відповідає вимогам, можна ввести всю інформацію.

### 3.2 Створення інтерфейсу користувача

Проектування підсистем і розробка інтерфейсів підсистем є важливим етапом створення програмного продукту. Ця робота залежить як користувач бачить створену підсистему, подобається йому це чи ні підсистема зручна у використанні та чи стане популярною.

Основна мета будь-якої підсистеми — забезпечити максимальний комфорт користувача або ефективність обробки даних. Важливою частиною є інтерфейс підсистеми під час створення.

Конструкція підсистеми повинна бути зрозумілою і простою. Правильно спроектована підсистема повинна сама інтегруватися наступні три основні особливості. Створена підсистема має дійсний і правильний дизайн, інтеграція та використання всіх можливих функцій. Наступною характеристикою є важливість відданості користувача у створеній підсистемі. Найкращий спосіб

зробити людинувикористовувати підсистему з великою кількістю функцій – для підвищення мотивації тацікавить користувача. Остання риса – практичність, оскільки високкласифікації оцінки мають підсистеми, які дійсно необхідні. Під час розробкипідсистеми дотримувалися всіх правил, тому вони були створені красиво і зрозумілопідсистема відповідності термінів

Сайт досить простий і зручний, тому все необхідне під рукою.Для створення використовувалися мови програмування html і css, а такожмова javascript. Зайшовши на сайт, користувач спочатку бачить, що перед нимголовна сторінка, тобто логотип сайту, який можна натиснути та перенестинас на першу сторінку, такий момент також створений для зручностіпанель з кнопками, необхідними для використання та пересуванняцей сайт. Переходьте на інші сторінки за допомогою логотипу та кнопок панелізалишаються на місці, тобто вони однакові на всіх сторінках. Ти можешсказати, що все, що вам потрібно, відразу видно, цього достатньоозручно та полегшує користування цим сайтом.

Користувач може взаємодіяти із зображеннями, де позначені ці послугизаходити на сайти цих сервісів і також користуватися необхідною інформацієюякий розміщений на їх веб-сайті.

Перейшовши в нижню частину головної сторінки, користувач можепрочитайте короткі замітки про поради цих служб.

Також у нижній частині сторінки є така ж панель із кнопками, що й у верхній частиніпочаток сайту. Це було зроблено для зручності та швидкого переходусторінки, якщо це необхідно. Є посилання на соціальні мережі, якщо ви швидківи повинні використовувати їх. Це економить багато часу і робитьцей проект ще швидше знаходить потрібну інформацію.

Головне меню сайту, зображено на рис 3.1, короткі відомості для користувачів на рис. 3.2, меню соціальних мереж на рис. 3.3.

Головна      Перелік      Контакти      Про нас

Рисунок 3.1 — Головне меню сайту

Код блоку меню наведений нижче

```
<nav>
  <div class="pages" id="myPages">
    <a href="golovna.html">Головна</a>
    <a href="perelik.html">Перелік</a>
    <a href="kontakty.html">Контакти</a>
    <a href="pro_nas.html">Про нас</a>
    <a id="menu" href="#" class="icon">&#9776;</a>
  </div>
</nav>
```



Рисунок 3.2 — Короткі відомості для користувачів

## Код до меню з соціальними мережами

```
<div class="social">  
    <a href="https://www.facebook.com"></a>  
    <a href="https://www.instagram.com"></a>  
    <a href="https://web.telegram.org"></a>  
    <a href="https://www.pinterest.com"></a>  
</div>
```



Рисунок 3.3 — Короткі відомості про соціальні мережі

Вміст, який знаходиться на іншій сторінці як "Список" — це перелік різних типів компаній і послуг для кожної сфери призначається декілька експертів. Список представлений у вигляді довідкової системи з переліком і мінімальною інформацією, якою можна використовувати без переходу на головні сторінки цих компаній і установи.

Фрагмент коду блоку установи наведено нижче



```

<div class="perelik-logo">
    <a href="https://vokl.org/"></a>
</div>
<div class="perelik-tema">
    <h1>обласна лікарня ім. М.І.Пирогова</h1>
</div>
<div class="perelik-photo">
    
    <h2>Смола Андрій</h2>
    <p>хірург</p>
</div>
<div class="perelik-photo">
    
    <h2>Кодак Едуард</h2>
    <p>психолог</p>
</div>
<div class="perelik-text">
    <p>Звернутись можна за адресою<br> м.Вінниця,
вул. Пирогова 46<br> номер реєстратури: (0432) 671-361</p>
</div>

```

Коли ви взаємодієте із зображеннями установ, ви також можете перемикатися на головні сторінки цих закладів, що досить зручно та економить час. Може бути важливим в якийсь момент. Перелік послуг і компаній може бути досить великим, але всі розміщуються послідовно.

На останній сторінці знаходиться «Про нас». Інформацію, особливо про нашу інформаційно-довідкову систему та місцезнаходження компанії для взаємодії в автономному режимі, якщо така є необхідність. Надана інформація досить зручна, ви також можете зв'язатися з нами за номером телефону. Якщо вам потрібна технічна підтримка або особисте звернення до нас.

Код до блоку персональної інформації

```

<div class="pro_info">
    <h2>Загальна інформація</h2>
    <p>Розташування за адресою вул. Соборна 59 на 2
поверсі, кабінет 211</p>
    <p>Надаємо можливість пошуку потрібних служб та
спеціалістів на одному сайті</p>
    <p>Номери для дзвінків: <h3>+380995671380 <br>
+380676612145</h3></p>
</div>

```

## 4 ТЕСТУВАННЯ СИСТЕМИ

### 4.1 Огляд рішень для тестування веб-додатків

Перед тим як опублікувати створений веб-застосунок, треба це зробити кілька кроків, щоб знайти та виправити помилки у веб-дизайні, макеті та взаємодія з користувачем. Без цього неможливо досягти хороших результатів. Це означає, що вашому проекту буде важче потрапити на вершину пошукових систем.

Тестування веб-додатків дозволяє перевірити працездатність усіх функцій у відношенні кількості прав користувача, швидкість і стабільність, читабельність інформації, зручність навігація по сайту, запобігання веб-атакам тощо. Ваші послуги та методи перевірки може відрізнитися, але ви повинні вирішити з самого початку стратегію та дотримуйтеся її. Необхідно добре продумати послідовність дій виконуються послідовно [17].

Основні види або роботи під час тестування:

- співвідношення фактичного плану та відповідного зразка в ТЗ;
- перевірка правильності розкладки;
- функціональне тестування;
- навантажене тестування;
- тестування безпеки сайту;

— тестування на сумісність.

Не всі розробки веб-сайтів мають недоліки дизайну. Точно порівнювати моделі та готові моделі нелегко, тому що потрібно зрозуміти, як це зробити як виглядає кожен елемент і сторінка. Крім того, версії для мобільних пристроїв і планшети вимагають більше уваги та створюють модель, яка підходить кожен гаджет і браузер.

Перевірка є важливою частиною тестування для розробки онлайн-ресурсів, які виглядають однаково в усіх браузерах, завантажувалися швидше та дозволяють пошуковим системам краще розуміти вміст.

Існує стандарт Консорціуму Всесвітньої павутини (W3C), який визначає правила та положення про створення сайтів. На його основі була розроблена W3C Markup.

Один з трудомістких тестів, що вимагає точності — функціональні розрахунки. Для прикладу розглянемо інтернет-магазин скільки місць для покупки певної кількості предметів, а не лише запасів. Усі комбінації повинні бути перевірені.

Функціональність залежить від типу ресурсу, що перевіряється, але крім цього є основні елементи, на які варто звернути увагу:

- тестування форм введення даних;
- перевірка охоплення пошуку та релевантності результатів;
- тестування навігації – перевірка посилань;
- перевірка даних, завантажених до бази даних;
- перевірка форми реєстрації та входу.

Тестування навантаження є важливим для запобігання несподіваним подіям закриття вашого сайту в години пік. Ось чому є деякі служби, які можуть, щоб імітувати поступове збільшення кількості користувачів вашого сайту. Якщо час завантаження вашого сайту таке ж, як і при низькому трафіку, далі все добре.

В інтернет-магазинах можна протестувати паралельне виконання транзакцій, щоб отримати час відповіді для всіх критичних бізнес-функцій.

Тестування безпеки сайту залежить від списку вразливостей сайту. Важливо видалити їх вчасно, щоб хакери не змогли «зламати» канал для залучення нових клієнтів через Інтернет. Системи безпеки перевіряють методи виявлення вторгнень, оцінки безпеки сайту та аналіз ризиків доступу зловмисників конфіденційної інформації.

Основні етапи тестування вразливостей:

- управління доступом;
- діагностика аутентифікації;
- перевірка даних;
- криптографія;
- механізм обробки помилок;
- інтеграція зі сторонніми сервісами;
- перевірити стійкість сайту до Dos/DDos атак;
- конфігурація сервера.

## 4.2 Інструкція по роботі з додатком

Коли ви відкриваєте цей сайт, ви повинні це пам'ятати користувачеві необхідний доступ до Інтернету, оскільки без нього використання сайту неможливо.

Доходячи до першої сторінки, користувач бачить перед собою головне меню сайту, через яке можуть перемикатися сторінки.

На першій сторінці представлені найважливіші послуги підтримки та потреби. Коли ви взаємодієте з зображеннями, користувач може перейти на офіційну сторінку цих сервісів або для компаній.

Прокручуючи сторінку вперед, користувач стикається з короткою інформацією, з якої ви можете взяти для себе щось корисне.

У нижній частині сторінки є ще один рядок меню доступ до соціальних мереж, з якими також взаємодіє користувач можна перейти на офіційні сторінки цих сайтів.

Коли користувач переходить на іншу сторінку, він знаходить там головну сторінку контент у вигляді послуг, компаній і експертів у цих сферах. Рядок меню залишається незмінним на всіх сторінках, тому це полегшує переміщення між ними.

Основний зміст цієї сторінки складається з короткого. Також виділено інформацію про компанію або послугу, декілька експертів із цих галузей та взаємодії з іміджем. Перейшовши на офіційну сторінку тієї чи іншої компанії.

Панель підменю та соціальні мережі також не змінилися на всіх сторінках, що також полегшує навігацію цими сторінками, оскільки не потрібно прокручувати сторінку вгору, щоб перейти на іншу.

При переході на третю сторінку користувача зустрічає панель введення персональних даних. Ця панель виглядає як три блоки які знаходяться один під одним.

У першому блоці пишеться назва користувача, його поточна електронна адреса вводиться у другому блоці, для здійснення зручного спілкування між користувачем і сайтом. Останній блок, де користувач може залишити повідомлення. Що зазвичай, це може бути скарга або повідомлення про недоліки або навіть інтереси чи особиста привабливість тієї чи іншої інституції експерт, а також ввічливість у формі подяки або підкреслення її.

Ця сторінка також має ті самі панелі меню та соціальних мережах, а й позитивний імідж, який демонструє легкість взаємодії між експертами та користувачами.

Коли користувач доходить до останньої сторінки, він бачить короткий інформація про нашу компанію, що надає послуги допоміжна система у вигляді переадресації іншим службам і для компаній.

Ця інформація включає адресу та номер телефону. І також зверху сторінка містить зручнішу для користувача карту відправляйтеся до нашої компанії. Можливо використовувати в особистих цілях, щоб все було під рукою зручне використання.

## 5 ЕКОНОМІЧНИЙ РОЗДІЛ

### 5.1 Оцінювання комерційного потенціалу розробки

Метою здійснення технологічного оцінювання є комерційний потенціал розробки проекту, створений в результаті науково-технічної діяльності.

Розробка проекту передбачає розрахунок економічної доцільності та ефективності для створення програмного продукту.

Дана система повинна допомагати своїм користувачам у необхідних для них ситуаціях. Для цього, в економічній частині потрібно розрахувати:

- кошторис витрат на розробку програмного продукту;
- експлуатаційні витрати, пов'язані з використанням нового програмного продукту;
- обсяг роботи, який може бути виконаний з застосуванням нового програмного продукту;

Оцінювання комерційного потенціалу здійснене за критеріями, що наведені в таблиці 5.1.

Таблиця 5.1 — Критерії оцінювання комерційного потенціалу розробки проекту та їх приблизна бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
критерій	0	1	2	3	4
Технічна здійсненність концепції:					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку

3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи:					
6	Ринок малий і немає позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкуренція немає

Продовження таблиці 5.1

Практична здійсненність:					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування

10	Необхідна розробканових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відома давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термінокупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Для чіткого оцінювання комерційного потенціалу розробки проекту, необхідно результати оцінювання внести в таблицю. Завдяки цьому можна буде розрахувати суму балів і оцінити рівень і потенціал даного проекту, який розробляється.

Експертну оцінку виконали три експерти, яким було запропоновано ознайомитись з даною розробкою. Результати оцінки комерційного потенціалу наведено у таблиці 5.2.

Таблиця 5.2 – Результати оцінювання комерційного потенціалу розробки

Критерії	Експерт		
	Експерт 1	Експерт 2	Експерт 3
	Бали, виставлені експертами:		
1. Технічна здійсненість концепції	2	2	2



2.Ринкові переваги (наявність аналогів)	2	2	3
3.Ринкові переваги (ціна)	3	3	4
4. Ринкові переваги (технічні властивості)	3	3	2
5.Ринкові переваги (експлуатаційні витрати)	2	3	2
6.Ринкові перспективи (розмір ринку)	4	3	3
7.Ринкові перспективи (конкуренція)	3	2	4
8.Практична здійсненість (наявність фахівців)	3	2	2
9.Практична здійсненість (наявність фінансів)	2	2	2
10.Практична здійсненість (необхідність нових матеріалів)	2	3	4
11.Термін реалізації	3	2	3
12.Розробка документів	3	3	3
Сума балів	32	30	34
Середньоарифметична сума балів	32		

Враховуючи який результат вийшов в таблиці 5.2 та беручи до уваги рекомендації, що знаходяться в таблиці 5.3, можна зробити висновок про рівень потенціалу комерційного розвитку.

Таблиця 5.3 — Рівні комерційного потенціалу розробки

Середньоарифметична сума балів розрахована на основі висновків експертів	Рівень комерційного потенціалу розробки
10 – 20	Низький
21–30	Нижчесереднього
31–40	Середній
41 –50	Вищесереднього
51– 60	Високий

Тому враховуючи експертну думку, та результати, які виходять, можна зробити висновок, що рівень комерційного потенціалу буде середній.

Що до аналогів, то їх як таких дуже незначна кількість. Є щось приблизно схоже, але має не той функціонал або основну ціль у використанні.

Основними недоліками аналогів є їх неповна змістова складова та недостатня функціональність для пошуку. У таблиці 5.4 наведено порівняння характеристик з аналогами на ринку.

Таблиця 5.4 — Порівняння характеристик розробки із аналогом

Показники	Розробка	Аналог 1	Аналог 2
Функціонал	9	7	6
Простота взаємодії	9	6	7
Надійність пошуку	7	8	7
Візуальна складова	8	7	6
Інтерфейс	8	6	7

Продукт буде просуватись за рахунок додавання до його функціоналу корисної реклами, а також використання буде здійснюватись за рахунок місцевої влади і їх сайту, тому усім їхнім користувачам відкриються нові можливості і загалом нова система, завдяки якій вони зможуть шукати необхідні їм служби порятунку.

Новизна дослідження полягає в тому, що створюється проект, який мінімізує необхідний пошук та зменшить витрати в часі на проведення дій при надзвичайних ситуаціях.

В порівнянні з аналогом, можна зауважити, що новий проект має більш зручний функціонал, візуальну складову та зручніший інтерфейс, для взаємодії з користувачем, тому цим сайтом зможе скористатись будь-хто, адже на ньому вказані інструкції по використанню, а також він доволі інтуїтивно-орієнтований, тому користування ним не є важким.

Продукт не є новим на ринку, але застосовані в ньому моделі та методи є новими в даній предметній області. Даний продукт планується реалізувати на

місцевому ринку, тому як він актуальний лише серед жителів конкретного міста.

На даний час програмний продукт готовий і знаходиться на стадії дослідного зразка.

Розробка потребує незначних фінансових ресурсів, а більшість матеріалів для її реалізації відомі і давно використовуються. На ринку праці наявні фахівці відповідної кваліфікації для обслуговування та підтримки нової розробки.

## 5.2 Розрахунок витрат на здійснення науково-дослідницької роботи

У магістерській роботі розглядається пошукова система, яка здійснюється за рахунок сайту на якому вона знаходиться, але значну частину витрат складають витрати на розробку проекту, а не на процес створення. Відповідно, є певна специфіка розрахунків.

### 5.2.1 Витрати на оплату праці

Основна заробітна плата розробників, що працюють над проектом, визначена у формулі 5.1:

$$Z_o = MT \cdot t, \quad (5.1)$$

де  $k$  — кількість дослідників, які безпосередньо брали участь у розробці;

$M$  — місячний посадовий оклад конкретного дослідника, грн;

$T$  — середня кількість робочих днів в місяці,  $T_p = 13 \dots 15$  днів; було обрано 14 днів;

$t$  — кількість днів роботи конкретного дослідника.

Над створенням розробки працював розробник та візуальний дизайнер, тому потрібно виконати для цих працівників усі необхідні розрахунки. Отже вносимо їх до таблиці 5.5:

$$Z_{o.p.} = (20000 \cdot 20) / 14 = 28\,570 \text{ (грн)}, Z_{o.d.} = (15000 \cdot 22) / 14 = 23\,570 \text{ (грн)}.$$

Під час розробки програмного продукту використовувались лише безкоштовні програмні засоби, які можна знайти у відкритому доступі.

Інші витрати охоплюють: витрати на обідню перерву, керування проектом, Інтернет, витрати на утримання, витрати на опалення, освітлення, водопостачання, тощо.

Таблиця 5.5 — Витрати на заробітну плату дослідників

Найменування посади	Місячний оклад	Оплата за робочий день	Кількість днівроботи	Витрати на заробітну плату
Розробник	20000	1000	20	28570
Дизайнер	15000	680	22	23570
Всього				52140

### 5.2.2 Основна заробітня плата робітників

Витрати на основну заробітну плату робітників ( $Z_p$ ) відзначено у формулі 5.2. За відповідними найменуваннями робіт НДР на тему «Впорядковане методичне забезпечення для доступу до локальних служб порятунку» розраховуємо за формулою:

$$Z_p = \sum_{i=1}^n C_i \cdot t_i \quad (5.2)$$

де  $C_i$  – погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год (форм. 5.3);

$t_i$  – час роботи робітника при виконанні визначеної роботи, год.

Величина витрат вказана у таблиці 5.6. Погодинну тарифну ставку робітника відповідного розряду  $C_i$  можна визначити за формулою:

$$C_i = \frac{M_M \cdot K_i \cdot K_c}{T_p \cdot t_{зм}} \quad (5.3)$$

де  $M_M$  — розмір заробітної плати робітника, прийmemo середню між двома нашими працівниками  $M_M = 17000$  грн;

$K_i$  — коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного стажу;

$K_c$  — мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників за кількість років стажу.

$T_p$  — середнє число робочих днів в місяці, приблизно  $T_p = 22$  дн;

$t_{зм}$  — тривалість зміни, год.

$$C_1 = 17000 \cdot 1 \cdot 5 / (22 \cdot 8) = 482,9 \text{ грн.}$$

$$З_{р1} = 965,9 \cdot 1 = 482,9 \text{ грн.}$$

Таблиця 5.6 — Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Стаж	Погодинна тарифна ставка, грн	Величина оплати на робітника, грн
1. Практичні	8	4	482,9	3863,6
2. Дизайнерські	6	3	321,9	1931,8
Всього				5795,4

### 5.2.3 Додаткова заробітня плата робітників

Додаткова заробітна плата (форм. 5.4). Додаткову заробітну плату розраховуємо як 10 ... 20% від суми основної заробітної плати дослідників та робітників за формулою:

$$З_{дод} = (З_o + З_p) \cdot \frac{N_{дод}}{100\%} \quad (5.4)$$

де  $N_{дод}$  — норма нарахування додаткової заробітної плати. Приймемо 15%.

$$З_{дод} = (52140 + 5795,4) \cdot 15 / 100\% = 8690,3 \text{ грн.}$$

#### 5.2.4 Відрахування на соціальні заходи

Нарахування на заробітну плату (форм. 5.5) дослідників та робітників розраховуємо як 24% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$Z_n = (Z_o + Z_p + Z_{\text{дод}}) \cdot \frac{H_{\text{зн}}}{100\%} \quad (5.5)$$

де  $H_{\text{зн}}$  – норма нарахування на заробітну плату. Приймаємо 24%.

$$Z_n = (52140 + 5795,4 + 8690,31) \cdot 24 / 100\% = 15990 \text{ грн.}$$

#### 5.2.5 Допоміжні матеріали та функції

До статті «допоміжні матеріали та функції» належать витрати на оренду домену для організації функціонування сайту, інструменти для задання параметрів для візуальної складової та придбані ресурси для тестування даного продукту, витрачені для проведення досліджень за темою «Впорядковане методичне забезпечення для доступу до локальних служб порятунку».

Витрати на матеріали ( $M$ ), у вартісному вираженні розраховуються окремо по кожному виду ресурсів за формулою:

$$M = \sum_{j=1}^n H_j \cdot C_j \cdot K_j - \sum_{j=1}^n B_j \cdot C_{\text{в}j} \quad (5.6)$$

де  $H_j$  — норма витрат матеріалу  $j$ -го найменування, шт.;

$n$  — кількість видів тестування на різних девайсах;

$C_j$  — вартість матеріалу  $j$ -го найменування, грн/шт.;

$K_j$  — коефіцієнт витрат на доставлення матеріалів;

$B_j$  — маса відходів  $j$ -го найменування, кг;

$C_{\text{в}j}$  — вартість відходів  $j$ -го найменування, грн/кг;

Проведені розрахунки зведемо до таблиці 5.7.

Таблиця 5.7 — Витрати на матеріали

Найменування матеріалу, тип	Ціна за 1 кг, грн	Норма витрат, кг	Вартість витраченого матеріалу, грн
Міні-драйвер	80	0,05	20
Всього			20
З врахуванням коефіцієнта транспортування			25

### 5.2.6 Розрахунок витрат на комплектуючі

Витрати на комплектуючі (К), які використовують при проведенні НДР на тему «Впорядковане методичне забезпечення для доступу до локальних служб порятунку».

Витрати на комплектуючі К, що були використані під час виконання даного етапу роботи, розраховуються за формулою 5.7.:

$$K = \sum_{i=1}^n N_i \cdot C_i \cdot K_i \quad (5.7)$$

де  $N_i$  — кількість комплектуючих  $i$ -го виду, шт.;

$C_i$  — ціна комплектуючих  $i$ -го виду, грн.;

$K_i$  — коефіцієнт витрат на доставлення матеріалів;

$n$  — кількість видів тестування на різних девайсах.

Зроблені розрахунки бажано звести до таблиці 5.8:

Таблиця 5.8 — Витрати на комплектуючі

Найменування комплектувальних	Кількість	Ціна за штуку, грн	Сума, грн
Дрід для зарядки	1	500	500
Міні-драйвер	7	20	140
Міні-монітор	1	1500	1500
Всього з врахуванням коефіцієнта транспортних витрат			2145

### 5.2.7 Амортизація обладнання, програмних засобів та приміщень

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо, розраховуємо з використанням прямолінійного методу амортизації за формулою:

$$A_{обл} = \frac{Ц_б}{T_в} \cdot \frac{t_{вик}}{12} \quad (5.8)$$

де  $Ц_б$  – балансова вартість обладнання, програмних засобів, приміщень тощо, які використовувались для проведення досліджень, грн;

$t_{вик}$  – термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців;

$T_в$  – строк корисного використання обладнання, програмних засобів, приміщень тощо, років.

$$A_{обл} = (30000 \cdot 1) / (2 \cdot 12) = 1250 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці 5.9.

Таблиця 5.9 – Амортизаційні відрахування по кожному виду обладнання

Найменування обладнання	Балансова вартість, грн	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн
Персональний комп'ютер	30000	4	2	1250
Робоче місце розробника ПЗ	120000	15	2	1333,3
Всього				2583,3

### 5.2.8 Службові відрядження

До статті «Службові відрядження» дослідної роботи належать витрати на відрядження штатних працівників, працівників організацій, які працюють за



договорами цивільно-правового характеру а також витрати на відрядження на наукові з'їзди, конференції, наради, пов'язані з виконанням конкретних досліджень.

Витрати за статтею розраховуємо як 20...25% від суми основної заробітної плати дослідників та робітників за формулою 5.9:

$$B_{cv} = (Z_o + Z_p) \cdot \frac{H_{cv}}{100\%} \quad (5.9)$$

де  $H_{cv}$  — норма нарахування, приймемо  $H_{cv} = 20\%$ .

$$B_{cv} = (52140 + 5795,4) \cdot 20 / 100\% = 11587 \text{ грн.}$$

### 5.2.9 Інші та накладні витрати

До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками.

Витрати за статтею «Інші витрати» розраховуємо як 50...100% від суми основної заробітної плати дослідників та робітників за формулою 5.10:

$$I_s = (Z_o + Z_p) \cdot \frac{H_{is}}{100\%} \quad (5.10)$$

де  $H_{is}$  – норма нарахування, приймемо  $H_{is} = 50\%$ .

$$I_s = (52140 + 5795,4) \cdot 50 / 100\% = 28967,7 \text{ грн.}$$

До статті «Накладні (загальновиробничі) витрати» належать: витрати, пов'язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов'язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов'язані з освоєнням виробництва продукції.

Витрати за статтею «Накладні (загально виробничі) витрати» розраховуємо як 100...150% від суми основної заробітної плати дослідників та робітників за формулою 5.11:

$$B_{нзв} = (Z_o + Z_p) \cdot \frac{H_{нзв}}{100\%} \quad (5.11)$$

де  $H_{нзв}$  – норма нарахування, прийmemo  $H_{нзв} = 100\%$ .

$$B_{нзв} = (52140 + 5795,4) \cdot 100 / 100\% = 57935,4 \text{ грн.}$$

Витрати на проведення науково-дослідної роботи на тему «Впорядковане методичне забезпечення для доступу до локальних служб порятунку», розраховуємо як суму всіх попередніх статей витрат за формулою 5.12:

$$B_{заг} = Z_o + Z_p + Z_{дод} + Z_n + M + K_e + B_{спец} + B_{прз} + A_{обл} + B_e + B_{св} + B_{сп} + I_e + B_{нзв} \quad (5.12)$$

$$B_{заг} = 52140 + 5795,4 + 8690,3 + 15990 + 25 + 2145 + 2583,3 + 11587 + 28967,7 + 57935,4 = 185859,1 \text{ грн.}$$

Загальні витрати  $ЗВ$  на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховується за формулою 5.13:

$$ЗВ = \frac{B_{заг}}{\eta} \quad (5.13)$$

де  $\eta$  — коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи, прийmemo  $\eta = 0,5$ .

$$ЗВ = 76828,03 / 0,5 = 371718,2 \text{ грн.}$$

### 5.3 Розрахунок ефективності вкладених інвестицій

В ринкових умовах узагальнюючим позитивним результатом, що його може отримати потенційний інвестор від можливого впровадження результатів тієї чи іншої науково-технічної розробки, є збільшення у потенційного інвестора величини чистого прибутку.

В цьому випадку основу майбутнього економічного ефекту будуть формувати:

$\Delta N$  – збільшення кількості споживачів яким надається відповідна інформаційна послуга у періоди часу, що аналізуються;

$N$  – кількість споживачів яким надавалась відповідна інформаційна послуга у році до впровадження результатів нової науково-технічної розробки, прийmemo 1 особа;

$C_o$  – вартість послуги у році до впровадження інформаційної системи, прийmemo 5000 грн;

$\pm \Delta C_o$  – зміна вартості послуги від впровадження результатів, прийmemo зростання на 200 грн.

Можливе збільшення чистого прибутку розраховуємо за формулою 5.14:

$$\Delta \Pi_i = (\pm \Delta C_o \cdot N + C_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\mathcal{G}}{100}\right) \quad (5.14)$$

де  $\lambda$  – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2021 році ставка податку на додану вартість складає 20%, а коефіцієнт  $\lambda = 0,8333$ ;

$\rho$  – коефіцієнт, який враховує рентабельність інноваційного продукту. Приймемо  $\rho = 40\%$ ;

$\mathcal{G}$  – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2023 році  $\mathcal{G} = 18\%$ ;

Збільшення чистого прибутку 1-го року:

$$\Delta \Pi_1 = (1 \cdot 200 + 5000 \cdot 400) \cdot 0,83 \cdot 0,4 \cdot (1 - 0,18/100\%) = 662871 \text{ грн.}$$

Збільшення чистого прибутку 2-го року:

$$\Delta\Pi_2 = (1 \cdot 200 + 5000 \cdot (400 + 300)) \cdot 0,83 \cdot 0,4 \cdot (1 - 0,18/100\%) = 1159974,6 \text{ грн.}$$

Збільшення чистого прибутку 3-го року:

$$\Delta\Pi_3 = (1 \cdot 200 + 5000 \cdot (400 + 300 + 200)) \cdot 0,83 \cdot 0,4 \cdot (1 - 0,18/100\%) = 1491377 \text{ грн.}$$

Приведена вартість збільшення всіх чистих прибутків  $ПП$ , що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки за формулою 5.15:

$$ПП = \sum_{i=1}^T \frac{\Delta\Pi_i}{(1 + \tau)^t} \quad (5.15)$$

де  $\Delta\Pi_i$  – збільшення чистого прибутку у кожному з років, протягом яких виявляються результати впровадження науково-технічної розробки, грн;

$T$  – період часу, протягом якого очікується отримання позитивних результатів від впровадження та комерціалізації науково-технічної розробки, роки;

$\tau$  – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні,  $\tau = 18\%$ ;

$t$  – період часу (в роках) від моменту початку впровадження науково-технічної розробки до моменту отримання потенційним інвестором додаткових чистих прибутків у цьому році.

$$ПП = 662871/(1+0,18)^1 + 1159974,6/(1+0,18)^2 + 1491377/(1+0,18)^3 = 2808663,1 \text{ грн.}$$

Величина початкових інвестицій  $PV$  які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки у формулі 5.16:

$$PV = k_{инв} \cdot ЗВ \quad (5.16)$$

де  $k_{инв}$  – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, приймаємо  $k_{инв}=2$ ;

$ЗВ$  – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, приймаємо 371718,2 грн.

$$PV = k_{инв} \cdot ЗВ = 2 \cdot 371718,2 = 743436,4 \text{ грн.}$$

Абсолютний економічний ефект  $E_{абс}$  для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме у формулі 5.17:

$$E_{абс} = ПП - PV \quad (5.17)$$

де  $ПП$  – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, 2808663,1 грн;

$PV$  – теперішня вартість початкових інвестицій, 743436,4 грн.

$$E_{абс} = ПП - PV = 2808663,1 - 743436,4 = 2065226,7 \text{ грн.}$$

Внутрішня економічна дохідність інвестицій  $E_e$  які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$E_e = T_{ж} \sqrt{1 + \frac{E_{абс}}{PV}} - 1 \quad (5.18)$$

де  $E_{абс}$  – абсолютний економічний ефект вкладених інвестицій, грн;

$PV$  – теперішня вартість початкових інвестицій, грн;

$T_{ж}$  – життєвий цикл науково-технічної розробки, тобто час від початку її розробки до закінчення отримання позитивних результатів від її впровадження, 3 роки.

$$E_{\epsilon} = \sqrt[3]{1 + \frac{E_{abc}}{PV}} - 1 = (1 + 2065226,7 / 743436,4)^{1/3} - 1 = 0,94.$$

Мінімальна внутрішня економічна дохідність вкладених інвестицій  $\tau_{min}$ :

$$\tau_{min} = d + f \quad (5.19)$$

де  $d$  – середньозважена ставка за депозитними операціями в комерційних банках; в 2023 році в Україні  $d = 0,1$ ;

$f$  – показник, що характеризує ризикованість вкладення інвестицій, приймемо  $0,25$ .

$\tau_{min} = 0,1 + 0,25 = 0,35 < 0,94$  це свідчить про те, що внутрішня економічна дохідність інвестицій  $E_{\epsilon}$ , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки вища мінімальної внутрішньої дохідності. Тобто інвестувати в науково-дослідну роботу за темою «Впорядковане методичне забезпечення для доступу до локальних служб порятунку» доцільно.

Період окупності інвестицій  $T_{ок}$  які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$T_{ок} = \frac{1}{E_{\epsilon}} \quad (5.20)$$

де  $E_{\epsilon}$  – внутрішня економічна дохідність вкладених інвестицій.

$$T_{ок} = 1 / 0,94 = 1,06 \text{ р.}$$

$T_{ок} < 3$ -х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

При оцінюванні рівня конкурентоспроможності, згідно узагальненого коефіцієнту конкурентоспроможності розробки, науково-технічна розробка загалом переважає існуючі аналоги.

Також термін окупності становить 1 рік, що менше 3-х років, тому це свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

Отже можна зробити висновок про доцільність проведення науково-дослідної роботи за темою «Впорядковане методичне забезпечення для доступу до локальних служб порятунку».

## ВИСНОВКИ

Магістерська кваліфікаційна робота присвячена розробці та створенню впорядкованого методичного забезпечення для доступу до локальних служб порядку, за допомогою HTML та CSS.

Для визначення напрямків дослідження було розглянуто існуючі сервіси для пошуку такої інформації, враховано їх переваги та недоліки. Закріплено вивчений матеріал по розробці веб-сайтів.

У першому розділі магістерської роботи ми проаналізували та оглянули сучасний стан служб порядку, їх різновид, класифікації та можливості.

У другому розділі ми ознайомились теоретичними методами та засобами розробки даного проекту. Описали зручність використання баз даних для збереження інформації, а також продемонстрували схеми зв'язків. Також провели аналіз існуючих методів створення сайтів.

У третьому розділі описали процес створення баз даних та процес створення інтерфейсу користувача, для зручного користування сайтом та для легкої взаємодії при інтуїтивно-орієнтованій обстановці проекту.

У четвертому розділі ми проводимо тестування веб-сайтів, а також огляд рішень що до практичної складової даного проекту. Проглядаємо функціонал та особливі складові цієї роботи. Вказуємо інструкції по користуванню даним сайтом та як орієнтуватись у ньому, щоб все було зрозуміло і легко навіть для недосвідчених або нових користувачів.

У п'ятому розділі було проведено економічний підрахунок доцільності розробки та загальної рентабельності цього проекту. Було проведено аналіз кількості конкурентів на ринку, перевірена їх актуальність і усі наші сильні та слабкі сторони. Також було проведено загальний підрахунок скільки було витрачено коштів на створення даного проекту та завдяки чому цей сайт буде приносити прибуток і яким чином він буде збільшуватись. Була підрахована зарплата працівників проекту. Тому з вище перерахованого було зрозуміло, що проект може створювати конкуренцію та бути успішним не тільки в плані функціоналу, але й фінансового.



## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Каляянов Д. П. Поліцейська діяльність демократичних держав та застосування її досвіду в організації роботи органів внутрішніх справ України: монографія. — Одеса: НДРВВ ОЮІ ХНУВС, 2007. — 248 с.
2. Організація пожежно-профілактичної роботи: навч. посіб. / О. В. Міллер, О. М. Парубок, А. І. Харчук ; М-во України з питань надзвичайних ситуацій та у справах захисту населення від наслідків Чорнобильської катастрофи, Львів. держ. ун-т безпеки життєдіяльності. — Л. : ЛДУ БЖД, 2009. — 392 с. : іл. — ISBN 97-966-655-202-3
3. Швидка та невідкладна медична допомога / І. С. Зозуля, І. С. Чекман, А. В. Вершигора та ін. — К.: Здоров'я, 2002. — 728 с.
4. AWS vs Azure vs Google Cloud - Detailed Cloud Comparison [Електронний ресурс]. Режим доступу: <https://intellipaat.com/blog/aws-vs-azure-vs-google-cloud/>
5. Google Classroom [Електронний ресурс]. Режим доступу: [https://uk.wikipedia.org/wiki/Google\\_Classroom](https://uk.wikipedia.org/wiki/Google_Classroom)
6. NEO LMS [Електронний ресурс]. Режим доступу: [https://uk.wikipedia.org/wiki/NEO\\_LMS](https://uk.wikipedia.org/wiki/NEO_LMS)
7. Guide to Web Application Architecture [Електронний ресурс]. Режим доступу: <https://www.intellectsoft.net/blog/web-application-architecture>
8. Web Application Architecture: Diagram and 8 Best Practices to Follow [Електронний ресурс]. Режим доступу: <https://asperbrothers.com/blog/web-application-architecture/>
9. Progressivewebapps [Електронний ресурс]. Режим доступу: [https://developer.mozilla.org/en-US/docs/Web/Progressive\\_web\\_apps](https://developer.mozilla.org/en-US/docs/Web/Progressive_web_apps) Guide to Web Application Architecture [Електронний ресурс]. Режим доступу: <https://www.intellectsoft.net/blog/web-application-architecture>

10. Pros and Cons of Using C# as Your Backend Programming Language[Электронныйресурс]. Режим доступа: <https://www.agilites.com/pros-and-cons-of-using-c-as-your-backend-programming-language.html>
11. Nodejs Vs. Go [Электронный ресурс]. Режим доступа:<https://www.peerbits.com/blog/nodejs-vs-golang.html>
12. JavaScript For Backend Development [Электронныйресурс]. Режимдоступу: <https://blog.hubspot.com/website/java-backend>
13. Pros and Cons of JavaScript [Электронныйресурс]. Режимдоступу:<https://data-flair.training/blogs/advantages-disadvantages-javascript>
14. The Good and the Bad of Node.js Web App Programming[Электронныйресурс]. Режим доступа: <https://www.altexsoft.com/blog/the-good-and-the-bad-of-node-js-web-app-development/>
15. Advantages and Disadvantages of PHP You Should Know WhenStarting a New Project [Электронныйресурс]. Режим доступа:<https://anywhere.epam.com/business/pros-and-cons-of-php>
16. Advantages and Disadvantages of PHP [Электронныйресурс].Режимдоступу: <https://www.geeksforgeeks.org/advantages-and-disadvantages-of-php/>
17. The Beginner's Guide to Python Back-End Development[Электронныйресурс]. Режим доступа:<https://blog.hubspot.com/website/python-backend>
18. Python vs. Node.js: Making the Right Choice for Backend[Электронныйресурс]. Режимдоступу: <https://elitex.systems/blog/python-vs-nodejs/Pros and Cons of Python Programming Language> [Электронныйресурс]. Режим доступа: <https://www.pixelcrayons.com/blog/software-development/python-pros-and-cons>
19. Ruby [Электронный ресурс]. Режим доступа: <https://www.ruby-lang.org/en/>

20. Ruby for backend development [Електронний ресурс]. Режим доступу: <https://talent500.co/ruby-for-backend-development/>

21. What Is Java: The Beginner's Guide to the Java Programming Language [Електронний ресурс]. Режим доступу: <https://blog.hubspot.com/website/javascript-for-backend>

22. JVM: що це у світі Java [Електронний ресурс]. Режим доступу: <https://foxminded.ua/jvm-tse/30>. Disadvantages of Java

[Електронний ресурс]. Режим доступу: <https://www.theserverside.com/blog/Coffee-Talk-Java-News-Stories-and-Opinions/Java-Disadvantages-Drawbacks-Advantages-17-Speed-21-Performance-8-Oracle-Vendor-Memory-CPU>

**ДОДАТОК А**

Міністерство освіти і науки України  
Вінницький національний технічний університет  
Факультет інформаційних технологій та комп'ютерної інженерії  
Кафедра обчислювальної техніки

ЗАТВЕРДЖУЮ  
Завідувачка кафедри ОТ  
проф., д.т.н. Азаров О.Д.  
«\_\_» \_\_\_\_\_ 2023 р.

**ТЕХНІЧНЕ ЗАВДАННЯ**

на виконання магістерської кваліфікаційної роботи  
«Впорядковане методичне забезпечення для доступу до локальних служб  
порятунку»

08-54.МКР.022.00.000 ПЗ

Керівник роботи:  
д.ф., ст. викл. каф. ОТ, Обертюх М.Р.  
«\_\_» \_\_\_\_\_ 2023 р.

Виконав: студент гр. 1КІ-22м  
Яросевич Ігор Русланович  
«\_\_» \_\_\_\_\_ 2023 р.

## 1 Підстава виконання магістерської кваліфікаційної роботи

1.1 У необхідні для людей моменти, де необхідно швидко реагувати і діяти, повинна бути під рукою система, в якій зручно, без лишніх затрат сил і часу можна знайти потрібну службу допомоги. Особиста безпека і безпека оточуючих це те чим потрібно дорожити у наш час.

### 1.2 Наказ про затвердження теми МКР

## 2 Мета і призначенням МКР

2.1 Мета роботи — є підвищення оперативності й швидкодії доступу до служб порятунку.

2.2 Призначення розробки — впорядковане методичне забезпечення для доступу до локальних служб порятунку.

## 3 Вихідні дані для виконання МКР

3.1 Проведення аналізу сучасного стану локальних служб порятунку;

3.2 Розробка бази даних та інтерфейсного додатку програмного засобу;

3.3 На здійснено тестування системи системи, а також написання інструкції по роботі з додатком;

3.4 Виконання розрахунків для доведення доцільності нової розробки з економічної точки зору;

## 4 Вимоги до виконання МКР

Головна вимога — розробка інтуїтивно-орієнтованого сайту, з зручним інтерфейсом та легким функціоналом.

## 5 Етапи МКР та очікувані результати

Етапи роботи та очікувані результати приведено в таблиці А.1.

Таблиця А.1—Етапи МКР

№ етапу	Назва етапу	Термін виконання		Очікувані результати
		початок	кінець	
1	Аналітичний огляд сучасного стану локальних служб порятунку	26.09.23	9.10.23	Аналітичний огляд джерел, розділ 1

## Продовження таблиці А.1

2	Вибір теоретичних методів розробки програмного засобу	10.10. 23	23.10.2 3	Розділ 2
3	Розробка програмного засобу	14.10. 23	6.11.23	Розділ 3
4	Тестування системи	7.11.2 3	20.11.2 3	Розділ 4
5	Підготовка економічної частини	21.11. 23	27.11.2 3	Розділ 5
7	Оформлення пояснювальної записки, графічного матеріалу і презентації	28.11. 23	4.12.23	ПЗ, графічний матеріал і презентація
8	Підготовка і підпис супроводжуючих документів, нормоконтроль та тест на плагіат	5.12.2 3	11.12.2 3	Оформлені документи

## 6 Матеріали, що подаються до захисту МКР

До захисту МКР подаються: пояснювальна записка МКР, ілюстративні та графічні матеріали, протокол попереднього захисту МКР на кафедрі, відгук наукового керівника, відгук опонента, протоколи складання державних екзаменів, анотації до МКР українською та іноземною мовами, довідка про відповідність оформлення МКР діючим вимогам.

## 7 Порядок контролю виконання та захисту МКР

Виконання етапів розрахункової та графічної документації МКР контролюється науковим керівником згідно зі встановленими термінами. Захист МКР відбувається на засіданні Державної екзаменаційної комісії, затвердженою наказом ректора.

## 8 Вимоги до оформлення МКР

При оформлюванні МКР використовуються:

— ДСТУ3008: 2015 «Звіти в сфері науки і техніки. Структура та

правила оформлювання»;

— ДСТУ8302:2015 «Бібліографічні посилання. Загальні положення та правила складання»;

— методичні вказівки до виконання магістерських кваліфікаційних робіт зі спеціальності 123 — «Комп'ютерна інженерія». Кафедра обчислювальної техніки ВНТУ 2022;

— документи на які посилаються вище вказаних.

8.2 Порядок виконання МКР викладено в «Положення про кваліфікаційні роботи на другому (магістерському) рівні вищої освіти СУЯ ВНТУ–03.02.02 П.001.01:21».

## ДОДАТОК Б

### Лістинг програмного коду

```
<!Doctype html>
<html>
<head>
<meta http-equiv="Content-Language" Content="ua">
<title>Системапослуг</title>
<link rel="stylesheet" href="css/style.css">
<link rel="apple-touch-icon" sizes="57x57" href="img/icon/apple-
icon-57x57.png">
<link rel="apple-touch-icon" sizes="60x60" href="img/icon/apple-
icon-60x60.png">
<link rel="apple-touch-icon" sizes="72x72" href="img/icon/apple-
icon-72x72.png">
<link rel="apple-touch-icon" sizes="76x76" href="img/icon/apple-
icon-76x76.png">
<link rel="apple-touch-icon" sizes="114x114" href="img/icon/apple-
icon-114x114.png">
<link rel="apple-touch-icon" sizes="120x120" href="img/icon/apple-
icon-120x120.png">
<link rel="apple-touch-icon" sizes="144x144" href="img/icon/apple-
icon-144x144.png">
<link rel="apple-touch-icon" sizes="152x152" href="img/icon/apple-
icon-152x152.png">
```



```
<link rel="apple-touch-icon" sizes="180x180" href="img/icon/apple-  
icon-180x180.png">
```

```
<link rel="icon" type="image/png" sizes="192x192"  
href="img/icon/android-icon-192x192.png">
```

```
<link rel="icon" type="image/png" sizes="32x32"  
href="img/icon/favicon-32x32.png">
```

```
<link rel="icon" type="image/png" sizes="96x96"  
href="img/icon/favicon-96x96.png">
```

```
<link rel="icon" type="image/png" sizes="16x16"  
href="img/icon/favicon-16x16.png">
```

```
<link rel="manifest" href="img/icon/manifest.json">
```

```
<meta name="msapplication-TileColor" content="#ffffff">
```

```
<meta name="msapplication-TileImage" content="img/icon/ms-icon-  
144x144.png">
```

```
<meta name="theme-color" content="#ffffff">
```

```
</head>
```

```
<body>
```

```
<header>
```

```
<div class="logo">
```

```
<a href="golovna.html"></a>
```

```
</div>
```

```
<nav>
```

```
<div class="pages" id="myPages">
```

```
<a href="golovna.html">ГОЛОВНА</a>
```

```

<a href="perelik.html">Перелік</a>
<a href="kontakty.html">Контакти</a>
<a href="pro_nas.html">Про нас</a>
<a id="menu" href="#" class="icon">&#9776;</a>
</div>
</nav>
</header>
<div class="mlogo">
<a
href="https://mistaua.com/21331/?setc
ity=181"></a>
</div>
<div class="mlogo">
<a
href="https://mistaua.com/8642/?setcity=181"></a>
</div>
<div class="mlogo">
<a href="https://micto.ua/vinnytska-miska-klinichna-
likarnia-shvydkoi-medychnoi-dopomohy-i157338/"></a>
</div>
<main>
<div class="advantages_container">
<div class="advantages">

```

```
<div class="advantages_item">
```

```

```

```
<h2>Безпека</h2>
```

```
<p>дотримання правил та вказівок, запобігання  
неприємностей у побуті</p>
```

```
</div>
```

```
<div class="advantages_item">
```

```

```

```
<h2>Закони</h2>
```

```
<p>зберігання порядку та моральних цінностей  
у суспільстві</p>
```

```
</div>
```

```
<div class="advantages_item">
```

```

```

```
<h2>Медицина</h2>
```

```
<p>піклуйтесь про здоров'я та довкілля</p>
```

```
</div>
```

```
<div class="advantages_item">
```

```

```

```
<h2>Життя</h2>
```

```
<p>кожен завжди повинен залишатись  
людиною</p>
```

```
</div>
```

```
<div class="advantages_item">
```

```

```

```
<h2>Порядок</h2>
```

```
<p>стратегія побудови нормальної  
спільноти</p>
```

```
</div>
```

```
<div class="advantages_item">
```

```

```

```
<h2>Волонтерство</h2>
```

```
50
```

```
<p>допомога у різних ситуаціях</p>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</main>
```

```

```

```
<footer>
```

```
<nav>
```

```
<a href="golovna.html">Головна</a>
```

```
<a href="perelik.html">Перелік</a>
```

```
<a href="kontakty.html">Контакти</a>
```

```
<a href="pro_nas.html">Про нас</a>
```

```
</nav>
```

```
<div class="social">
```

```
<a href="https://www.facebook.com"></a>
```

```
<a href="https://www.instagram.com"></a>
<a href="https://web.telegram.org"></a>
<a href="https://www.pinterest.com"></a>
</div>
<p>створив Яросевич Ігор 2022</p>
</footer>
<script src="js/script.js"></script>
</div>
</body>
</html>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Language" Content="ua">
<title>Система послуг</title>
<link rel="stylesheet" href="css/style.css">
<link rel="apple-touch-icon" sizes="57x57" href="img/icon/apple-
icon-57x57.png">
<link rel="apple-touch-icon" sizes="60x60" href="img/icon/apple-
icon-60x60.png">
<link rel="apple-touch-icon" sizes="72x72" href="img/icon/apple-
icon-72x72.png">
<link rel="apple-touch-icon" sizes="76x76" href="img/icon/apple-
```

```
icon-76x76.png">
<link rel="apple-touch-icon" sizes="114x114" href="img/icon/apple-
icon-114x114.png">
<link rel="apple-touch-icon" sizes="120x120" href="img/icon/apple-
icon-120x120.png">
<link rel="apple-touch-icon" sizes="144x144" href="img/icon/apple-
icon-144x144.png">
<link rel="apple-touch-icon" sizes="152x152" href="img/icon/apple-
icon-152x152.png">
<link rel="apple-touch-icon" sizes="180x180" href="img/icon/apple-
icon-180x180.png">
<link rel="icon" type="image/png" sizes="192x192"
href="img/icon/android-icon-192x192.png">
<link rel="icon" type="image/png" sizes="32x32"
href="img/icon/favicon-32x32.png">
<link rel="icon" type="image/png" sizes="96x96"
href="img/icon/favicon-96x96.png">
<link rel="icon" type="image/png" sizes="16x16"
href="img/icon/favicon-16x16.png">
<link rel="manifest" href="img/icon/manifest.json">
<meta name="msapplication-TileColor" content="#ffffff">
<meta name="msapplication-TileImage" content="img/icon/ms-icon-
144x144.png">
<meta name="theme-color" content="#ffffff">
</head>
```

```

<body>
<header>
<div class="logo">
<a href="golovna.html"></a>
</div>
<nav>
<div class="pages" id="myPages">
<a href="golovna.html">ГОЛОВНА</a>
<a href="perelik.html">Перелік</a>
<a href="kontakty.html">КОНТАКТИ</a>
<a href="pro_nas.html">Про нас</a>
<a id="menu" href="#" class="icon">&#9776;</a>
</div>
</nav>
</header>
<main>
<div class="perelik-perelik">
<div class="perelik-container">
<div class="perelik-logo">
<a href="https://vokl.org/"></a>
</div>
<div class="perelik-tema">
<h1>обласна лікарня ім. М.І.Пирогова</h1>

```

</div>

<div class="perelik-photo">



<h2>Смола Андрій</h2>

<p>хірург</p>

</div>

<div class="perelik-photo">



<h2>Кодак Едуард</h2>

<p>психолог</p>

</div>

<div class="perelik-text">

<p>Звернутись можна за адресою<br> м.Вінниця, вул.

Пирогова 46<br> номер реєстрації: (0432) 671-361</p>

</div>

<div class="perelik-logo">

<a href="https://stomaline.com.ua/ua/"></a>

</div>

<div class="perelik-tema">

<h1>стоматологічна клініка</h1>

</div>

<div class="perelik-photo">



<h2>Луг Микола</h2>



<p>лікар-терапевт</p>

</div>

<div class="perelik-photo">



<h2>Пасіка Ірина</h2>

<p>стоматолог</p>

</div>

<div class="perelik-text">

<p>Звернутись можна за адресою<br> м.Вінниця, вул.

Архітектора Артинова, 13<br> номер: +38 (097) 440-80-40</p>

</div>

<div class="perelik-logo">

<a href="https://vn.dsoua.com/ua/"></a>

</div>

<div class="perelik-tema">

<h1>газова служба</h1>

</div>

<div class="perelik-photo">



<h2>Енішин Роман</h2>

<p>контролер</p>

</div>

<div class="perelik-photo">



<h2>Грунт Олексій</h2>

<p>технік</p>

</div>

<div class="perelik-text">

<p>Звернутись можна за адресою<br> м.Вінниця, вул.

провулок Костя Широцького 24,<br> номер: (0432) 509 104</p>

</div>

<div class="perelik-logo">

<a href="https://vinvk.com.ua/"></a>

</div>

<div class="perelik-tema">

<h1>ВОДОКАНАЛ</h1>

</div>

<div class="perelik-photo">



<h2>Щедрик Оксана</h2>

<p>директор</p>

</div>

<div class="perelik-photo">



<h2>Судак Антон</h2>

<p>технік</p>

</div>

<div class="perelik-text">

<p>Звернутись можна за адресою<br> м.Вінниця, вул.

Київська, 173<br> номер: 67-03-12</p>

</div>

<div class="perelik-logo">

<a

href="https://www.facebook.com/dsnsvin/"></a>

</div>

<div class="perelik-tema">

<h1>рятувальна служба</h1>

</div>

<div class="perelik-photo">



<h2>Смерека Андрій</h2>

<p>пожежник</p>

</div>

<div class="perelik-photo">



<h2>Розова Анна</h2>

<p>рятувальник</p>

</div>

<div class="perelik-text">

<p>Звернутись можна за адресою<br> м.Вінниця, вул.

600-річчя, 11<br> номер: 0432 566 500</p>

</div>

```
</div>
</div>
</main>
<footer>
<nav>
<a href="golovna.html">ГОЛОВНА</a>
<a href="perelik.html">Перелік</a>
<a href="kontakty.html">Контакти</a>
<a href="pro_nas.html">Про нас</a>
</nav>
<div class="social">
<a href="https://www.facebook.com"></a>
<a href="https://www.instagram.com"></a>
<a href="https://web.telegram.org"></a>
<a href="https://www.pinterest.com"></a>
</div>
<p>створив Ярославич Ігор 2022</p>
</footer>
<script src="js/script.js"></script>
</div>
</body>
```

```
</html>

<!Doctype html>

<html>

<head>

<meta http-equiv="Content-Language" Content="ua">

<title>Система послуг</title>

<link rel="stylesheet" href="css/style.css">

<link rel="apple-touch-icon" sizes="57x57" href="img/icon/apple-
icon-57x57.png">

<link rel="apple-touch-icon" sizes="60x60" href="img/icon/apple-
icon-60x60.png">

<link rel="apple-touch-icon" sizes="72x72" href="img/icon/apple-
icon-72x72.png">

<link rel="apple-touch-icon" sizes="76x76" href="img/icon/apple-
icon-76x76.png">

<link rel="apple-touch-icon" sizes="114x114" href="img/icon/apple-
icon-114x114.png">

<link rel="apple-touch-icon" sizes="120x120" href="img/icon/apple-
icon-120x120.png">

<link rel="apple-touch-icon" sizes="144x144" href="img/icon/apple-
icon-144x144.png">

<link rel="apple-touch-icon" sizes="152x152" href="img/icon/apple-
icon-152x152.png">

62

<link rel="apple-touch-icon" sizes="180x180" href="img/icon/apple-
```

```
icon-180x180.png">
<link rel="icon" type="image/png" sizes="192x192"
href="img/icon/android-icon-192x192.png">
<link rel="icon" type="image/png" sizes="32x32"
href="img/icon/favicon-32x32.png">
<link rel="icon" type="image/png" sizes="96x96"
href="img/icon/favicon-96x96.png">
<link rel="icon" type="image/png" sizes="16x16"
href="img/icon/favicon-16x16.png">
<link rel="manifest" href="img/icon/manifest.json">
<meta name="msapplication-TileColor" content="#ffffff">
<meta name="msapplication-TileImage" content="img/icon/ms-icon-
144x144.png">
<meta name="theme-color" content="#ffffff">
</head>
<body>
<header>
<div class="logo">
<a href="golovna.html"></a>
</div>
<nav>
<div class="pages" id="myPages">
<a href="golovna.html">Головна</a>
<a href="perelik.html">Перелік</a>
```

```
<a href="kontakty.html">Контакти</a>
<a href="pro_nas.html">Про нас</a>
<a id="menu" href="#" class="icon">&#9776;</a>
</div>
</nav>
</header>
<main>
<form class="form">
<p><input type="text" name="name" class="feedback-input"
placeholder="Ім'я" id="name"></p>
<p><input type="email" name="email" class="feedback-input"
placeholder="Пошта" id="email"></p>
<p><textarea name="text" id="comment" class="feedback-
input" placeholder="Повідомлення"></textarea></p>
<input type="submit" value="Додати" class="button-submit">
<div class="ease"></div>
</form>
<div class="mlogo">

</div>
</main>
64
<footer>
<nav>
<a href="golovna.html">Головна</a>
```

```
<a href="perelik.html">Перелік</a>
<a href="kontakty.html">Контакти</a>
<a href="pro_nas.html">Про нас</a>
</nav>
<div class="social">
<a href="https://www.facebook.com"></a>
<a href="https://www.instagram.com"></a>
<a href="https://web.telegram.org"></a>
<a href="https://www.pinterest.com"></a>
</div>
<p>створив Яросевич Ігор 2022</p>
</footer>
<script src="js/script.js"></script>
</div>
</body>
</html>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Language" Content="ua">
<title>Система послуг</title>
```



```
<link rel="stylesheet" href="css/style.css">
```

```
<link rel="apple-touch-icon" sizes="57x57" href="img/icon/apple-  
icon-57x57.png">
```

```
<link rel="apple-touch-icon" sizes="60x60" href="img/icon/apple-  
icon-60x60.png">
```

```
<link rel="apple-touch-icon" sizes="72x72" href="img/icon/apple-  
icon-72x72.png">
```

```
<link rel="apple-touch-icon" sizes="76x76" href="img/icon/apple-  
icon-76x76.png">
```

```
<link rel="apple-touch-icon" sizes="114x114" href="img/icon/apple-  
icon-114x114.png">
```

```
<link rel="apple-touch-icon" sizes="120x120" href="img/icon/apple-  
icon-120x120.png">
```

```
<link rel="apple-touch-icon" sizes="144x144" href="img/icon/apple-  
icon-144x144.png">
```

```
<link rel="apple-touch-icon" sizes="152x152" href="img/icon/apple-  
icon-152x152.png">
```

66

```
<link rel="apple-touch-icon" sizes="180x180" href="img/icon/apple-  
icon-180x180.png">
```

```
<link rel="icon" type="image/png" sizes="192x192"  
href="img/icon/android-icon-192x192.png">
```

```
<link rel="icon" type="image/png" sizes="32x32"  
href="img/icon/favicon-32x32.png">
```

```
<link rel="icon" type="image/png" sizes="96x96"
```

```
href="img/icon/favicon-96x96.png">
<link rel="icon" type="image/png" sizes="16x16"
href="img/icon/favicon-16x16.png">
<link rel="manifest" href="img/icon/manifest.json">
<meta name="msapplication-TileColor" content="#ffffff">
<meta name="msapplication-TileImage" content="img/icon/ms-icon-
144x144.png">
<meta name="theme-color" content="#ffffff">
</head>
<body>
<header>
<div class="logo">
<a href="golovna.html"></a>
</div>
<nav>
<div class="pages" id="myPages">
<a href="golovna.html">Головна</a>
<a href="perelik.html">Перелік</a>
<a href="kontakty.html">Контакти</a>
<a href="pro_nas.html">Про нас</a>
<a id="menu" href="#" class="icon">&#9776;</a>
</div>
</nav>
</header>
```

```
<main>
<div class="pro_info">
<h2>Загальна інформація</h2>
<p>Розташування за адресою вул. Соборна 59 на 2 поверсі,
кабінет 211</p>
<p>Надаємо можливість пошуку потрібних служб та
спеціалістів на одному сайті</p>
<p>Номери для дзвінків: <h3>+380995671380 <br>
+380676612145</h3></p>
</div>
<div id="map"></div>
</main>
<footer>
<nav>
<a href="golovna.html">Головна</a>
<a href="perelik.html">Перелік</a>
<a href="kontakty.html">Контакти</a>
<a href="pro_nas.html">Про нас</a>
</nav>
<div class="social">
<a href="https://www.facebook.com"></a>
<a href="https://www.instagram.com"></a>
<a href="https://web.telegram.org"></a>
<a href="https://www.pinterest.com"></a>
</div>

<p>створив Ярославич Ігор 2022</p>
</footer>

<script>
function initMap() {
var uluru = {lat: 49.23261, lng: 28.46849};
var map = new
google.maps.Map(document.getElementById('map'), {
zoom: 5.7,
center: uluru
});
var marker = new google.maps.Marker({
position: uluru,
map: map
});
}
</script>

<script async defer
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyA4sl0ZcelJ
ECN3zFgLK3NU9h4ltMnVr4g&callback=initMap">
</script>

<script src="js/script.js"></script>
```

</div>

</body>

</html>

