**Пояснювальна записка  
до дипломного проєкту**

на тему: **Тема дипломного проєкту**

КПІ.ІП-1XXX.ХХХХХХ.02.81

Київ – 2025

Зміст

[Вступ 5](#_Toc193716787)

[1 ПЕРЕДПРОЄКТНЕ ОБСТЕЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ 6](#_Toc193716788)

[1.1 Постановка завдання дипломного проєктування 6](#_Toc193716789)

[1.2 Аналіз предметної області 6](#_Toc193716790)

[1.3 Аналіз існуючих рішень 7](#_Toc193716791)

[1.3.1 Аналіз відомих програмних продуктів 7](#_Toc193716792)

[1.3.2 Аналіз відомих алгоритмічних та технічних рішень 8](#_Toc193716793)

[1.4 Аналіз та моделювання бізнес-процесів 8](#_Toc193716794)

[Висновки до розділу 9](#_Toc193716795)

[2 РОЗРОБЛЕННЯ вимог до програмного забезпечення 10](#_Toc193716796)

[2.1 Варіанти використання програмного забезпечення 10](#_Toc193716797)

[2.2 Розроблення функціональних вимог 11](#_Toc193716798)

[2.3 Розроблення нефункціональних вимог 13](#_Toc193716799)

[2.4 Аналіз системних вимог 13](#_Toc193716800)

[2.5 Аналіз економічних показників програмного забезпечення 13](#_Toc193716801)

[2.6 Постановка завдання на розробку програмного забезпечення 13](#_Toc193716802)

[Висновки до розділу 13](#_Toc193716803)

[3 КОНСТРУЮВАННЯ ТА РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ 14](#_Toc193716804)

[3.1 Архітектура програмного забезпечення 14](#_Toc193716805)

[3.2 Архітектурні рішення та обґрунтування вибору засобів розробки 14](#_Toc193716806)

[3.3 Конструювання програмного забезпечення 15](#_Toc193716807)

[3.3.1 Розробка алгоритму … 15](#_Toc193716808)

[3.3.2 Опис структури бази даних … 16](#_Toc193716809)

[3.4 Аналіз безпеки даних 17](#_Toc193716810)

[Висновки до розділу 18](#_Toc193716811)

[4 АНАЛІЗ ЯКОСТІ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ 19](#_Toc193716812)

[4.1 Аналіз якості ПЗ 19](#_Toc193716813)

[4.2 Опис процесів тестування 19](#_Toc193716814)

[4.3 Опис контрольного прикладу 20](#_Toc193716815)

[Висновки до розділу 20](#_Toc193716816)

[5 РОЗГОРТАННЯ ТА СУПРОВІД ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ 21](#_Toc193716817)

[5.1 Розгортання програмного забезпечення 21](#_Toc193716818)

[5.2 Супровід програмного забезпечення 21](#_Toc193716819)

[Висновки до розділу 21](#_Toc193716820)

[ВИСНОВКИ 22](#_Toc193716821)

[СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 23](#_Toc193716822)

[ДОДАТКИ 24](#_Toc193716823)

Перелік умовних позначень

|  |  |  |
| --- | --- | --- |
| IDE | – | Integrated Development Environment – інтегроване середовище розробки. |
| API | – | Application programming interface, прикладний програмний Інтерфейс. |
| SDK | – | Software development kit. |
| IT | – | Інформаційні технології. |
| ER | – | Entity-Relation diagram. |
| OC | – | Операційна система. |
| БД | – | База даних. |

Вступ

<У вступі стисло викладають:

актуальність розробки програмного забезпечення за тематикою ДП; провідні тенденції розв’язання поставлених проблем і/або завдань; оцінку сучасного стану розробок за тематикою ДП, коротко розкриваючи практично розв’язані завдання провідними науковими установами та організаціями, а також провідними вченими й фахівцями певної галузі; коротко, що буде виконано в рамках даного ДП; призначення та мета ДП; можливі сфери застосування.>

# ПЕРЕДПРОЄКТНЕ ОБСТЕЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

## Постановка завдання дипломного проєктування

<У підрозділі викладають:

список завдань, які необхідно виконати у рамках дипломного проєктування.>

Дипломне проєктування передбачає виконання наступних завдань:

* аналіз предметної області та опис її ключових бізнес-процесів, визначення загального завдання розробки у рамках ДП;
* аналіз існуючих рішень (як програмних продуктів, так і алгоритмічних чи технічних рішень) обраного завдання розробки у рамках ДП;
* аналіз та моделювання бізнес-процесів;
* розроблення функціональних, нефункціональних та системних вимог до програмного забезпечення;
* аналіз економічних показників програмного забезпечення ДП;
* постановка завдання на розробку програмного забезпечення ДП;
* розроблення архітектури програмного забезпечення;
* розроблення архітектурних рішень та обґрунтування вибору засобів розробки програмного забезпечення;
* конструювання та розроблення програмного забезпечення;
* аналіз безпеки даних програмного забезпечення;
* аналіз якості та тестування програмного забезпечення;
* розгортання та супровід програмного забезпечення;
* створення супроводжувальної документації до розробленого програмного забезпечення.

## Аналіз предметної області

<У підрозділі викладають:

загальні положення по предметній області з посиланнями.

Почати варто з загального огляду предметної області, загальних визначень по ній, напрямків розробок, тощо. Допускається навести загальну теорію. Обов’язково надати посилання на джерела інформації.

Процес використання знань предметної області у програмному забезпеченні на поточний момент розвитку ІТ-технологій. Фактично описують процес імплементації предметної області у ІТ as-is (як є). Описати бізнес-процеси предметної області. Описати недоліки поточного стану речей з предметною областю у сфері ІТ. Описати можливі шляхи покращення ситуації з розробками у сфері ІТ за вашою предметною областю. Вказати який шлях обрали ви в рамках дипломного проєкту.>

## Аналіз існуючих рішень

Проаналізуємо відоме на сьогодні алгоритмічне забезпечення у даній області та технічні рішення, що допоможуть у реалізації <вашої розробки, вказати назву>. Далі будуть розглянуті готові програмні рішення, допоміжні програмні засоби та засоби розробки.

### Аналіз відомих програмних продуктів

<У пункті викладають:

опис готових програмних продуктів за вашою предметною областю, які частково чи повністю реалізують функціонал описаний у технічному завданні. Опис програмних продуктів має супроводжуватись ілюстраціями (скріншотами). Обов’язково має бути порівняльна таблиця функціоналу та особливостей з вашою розробкою.>

Для порівняння проєкту з аналогом можна скористатись таблицею 1.3.

Таблиця 1.3 – Порівняння з аналогами

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Функціонал | Дипломний проєкт (пишемо назву) | Аналог для порівняння 1 (пишемо назву) | Аналог для порівняння n (пишемо назву) | Пояснення |
|  |  |  |  |  |

### Аналіз відомих алгоритмічних та технічних рішень

<У пункті викладають:

теорію та порівняльний аналіз відомих алгоритмів для розв’язання задач чи підзадач, які будуть використовуватись у вашій розробці, в кінці обирається той, який ви будете використовувати при розробці програмного забезпечення ДП. Або робиться висновок про необхідність розробки оригінальних алгоритмів чи модифікацію існуючих. Розглядаються відомі технічні рішення, які допоможуть у реалізації розробки: відомі загальні архітектури ПЗ, платформи тощо. Для технічних рішень також робиться порівняльний аналіз та обираються ті, що будуть використані у розробці. Можливо зробити висновок про розробку оригінального технічного рішення, чи модифікацію існуючого.>

## Аналіз та моделювання бізнес-процесів

<У підрозділі викладають:

короткий аналіз бізнес-процесів, які необхідно моделювати (передбачає вибір декількох бізнес-процесів із загальної множини, які є ключовими для програмного забезпечення ДП), наочне представлення бізнес-процесів за допомогою засобів моделювання (BPMN або інші).>

Для моделювання бізнес-процесу використовується BPMN модель (рисунок 1.1) (рисунок слугує ілюстрацією BPMN моделі).

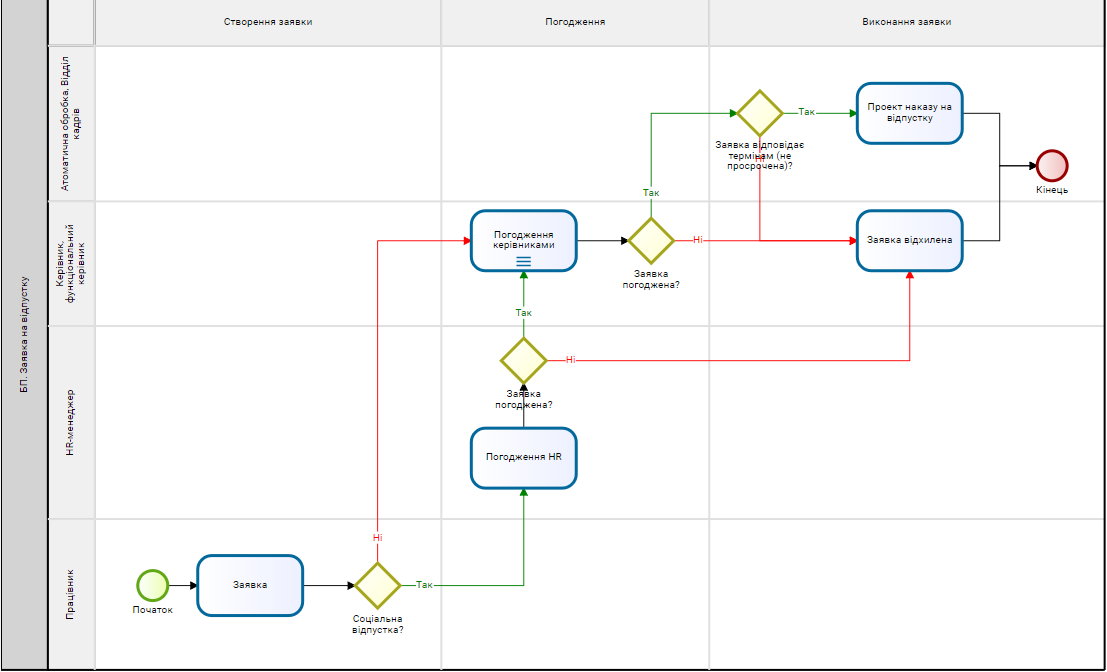


Рисунок 1.1 – Модель бізнес-процесу

Опис моделі бізнес-процесу створення облікового запису користувача:

* користувач переходить на сторінку реєстрації;
* користувач заповнює поля реєстрації;
* якщо введені поля не відповідають шаблону заповнення на клієнтській стороні, відповідні поля виділяються як заповнені некоректно.

## Висновки до розділу

<Необхідно стисло описати усе, що було виконано у даному розділі. Обсяг 0,75-1 сторінка>

# РОЗРОБЛЕННЯ вимог до програмного забезпечення

## Варіанти використання програмного забезпечення

<У підрозділі викладають:

діаграму варіантів використання та опис варіантів використання.

За згодою з керівником можна викласти інший спосіб виявлення вимог до програмного забезпечення ДП.>

Діаграма варіантів використання з ключовими акторами та основними функціями показана на рисунку 2.1 (дана діаграма є ілюстрацією, а не вашою діаграмою варіантів використання).

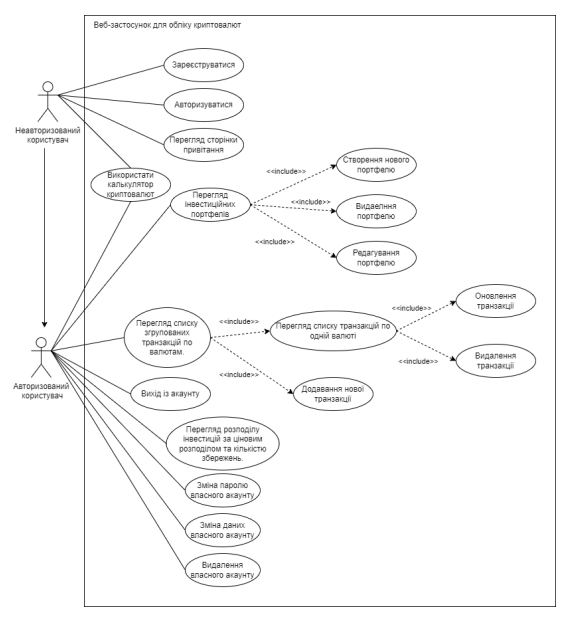


Рисунок 2.1 – Діаграма варіантів використання

В таблицях 2.1-2.14 наведено опис варіантів використання програмного забезпечення.

Таблиця 2.1 – Варіант використання UC-01

|  |  |
| --- | --- |
| Use case name | Реєстрація користувача |
| Use case ID | UC-01 |
| Goals | Реєстрація нового користувача в системі |
| Actors | Гість (незареєстрований користувач) |
| Trigger | Користувач бажає зареєструватися |
| Pre-conditions | - |
| Flow of Events | Користувач переходить на сторінку реєстрації. В поля для реєстрації вводяться відповідні дані: пошта користувача, пароль в системі, та його повтор для підтвердження, а також чек-бокс для підтвердження умов сервісу. Після заповнення даних користувача натискає кнопку реєстрації. Після цього з’являється повідомлення про успішну реєстрацію, і користувач перенаправляється на сторінку входу. |
| Extension | В випадку введення не коректних даних, кнопка реєстрації стає неактивною. Якщо якесь конкретне поле введено некоректно, то воно підсвічується червоним надписом. |
| Post-Condition | Створення сторінки користувача, перехід на сторінку входу |

<Якщо кількість таблиць описів варіантів використання більше 5, рекомендується за згодою з керівником винести їх у додатки.>

## Розроблення функціональних вимог

<У підрозділі викладають:

модель вимог, їх опис та матрицю трасування.>

Програмне забезпечення розділене на модулі. Кожен модуль має свій певний набір функцій. В таблиці 2.15 наведено загальну модель вимог, а в таблицях 2.16 – 2.27 наведений опис функціональних вимог до програмного забезпечення. Матрицю трасування вимог можна побачити на рисунку 2.3.

Таблиця 2.15 – Загальна модель вимог

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Назва | ID Вимоги | Пріоритети | Ризики |
| 1 | Перегляд сторінки привітання. | FR-1 | Низький | Низький |
| 2 | Система авторизації користувача. | FR-2 | Високий | Високий |
| 2.1 | Реєстрація користувача. | FR-3 | Високий | Високий |
| 2.2 | Авторизація користувача. | FR-4 | Високий | Високий |
| 2.3 | Вихід із акаунту. | FR-5 | Середній | Низький |

Таблиця 2.16 – Перелік функціональних вимог

|  |  |
| --- | --- |
| Назва | Опис |
| FR-1 | Перегляд сторінки привітання.  Неавторизований користувач бачить вебсторінку привітання. На ній же можна обрати можливість зареєструватися або авторизуватися. |
| FR-2 | … |

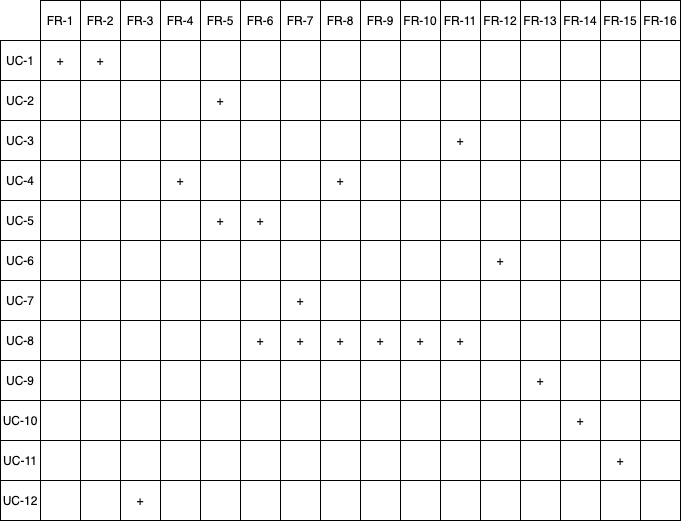


Рисунок 2.3 – Матриця трасування вимог

## Розроблення нефункціональних вимог

<У підрозділі викладають:

опис нефункціональних вимог за класифікацією.>

## Аналіз системних вимог

<У підрозділі викладають:

системні вимоги – підрозділ опціональний, залежить від роботи.>

## Аналіз економічних показників програмного забезпечення

<У підрозділі викладають:

виходячи з специфікації вимог до програмного забезпечення та враховуючи одного (двох) виконавця(-ів), розрахунок відповідних економічних показників програмного забезпечення.

Показати методику розрахунку, основні етапи, порівняти результати, якщо є аналогічні дані відомого програмного забезпечення.>

## Постановка завдання на розробку програмного забезпечення

<У підрозділі викладають:

Визначити мету розробки та що саме ви збираєтесь реалізувати. Це не зовсім вимоги і варіанти використання. Це узагальнена постановка завдання розробки програмного забезпечення з метою, цілями та задачами, що підлягають вирішенню у результаті розробки ПЗ на дипломний проєкт.>

## Висновки до розділу

< Необхідно стисло описати усе, що було виконано у даному розділі. Обсяг 0,75-1 сторінка>

За результатами розділу сформовано технічне завдання на розробку програмного забезпечення. <окремий документ за шаблоном «ДП\_шаблон\_ч1\_ТЗ»>

# КОНСТРУЮВАННЯ ТА РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## Архітектура програмного забезпечення

<У підрозділі викладають:

тип архітектури (архітектурний патерн) для вашого ПЗ та деталізація його компонентів за необхідності. Необхідно виконати повний опис архітектури. Для цього можуть бути використані (один з варіантів):

* діаграми C4 Model:
* архітектурні діаграми для представлення високорівневої архітектури (рівні 1 та 2 моделі C4 Model представлення архітектури);
* архітектурні діаграми для детального представлення архітектури (рівні 3 та 4);
* діаграми в нотації UML:
* діаграми компонентів та діаграми пакетів;
* діаграми для представлення поведінки (наприклад, діаграми послідовностей);
* діаграми класів для представлення логічної моделі даних;
* інші.>

## Архітектурні рішення та обґрунтування вибору засобів розробки

<У підрозділі викладають:

опис та обґрунтування важливих архітектурних рішень розробки. Наприклад:

* управління користувачами, як відбувається логін користувача в систему – за допомогою single-sign on, чи програмне забезпечення зберігає акаунти користувачів і виконує логін самостійно;
* інтеграції із зовнішніми системами, які системи та які протоколи використовуються;
* як саме забезпечується виконання нефункціональних вимог;
* які типи баз даних вибрані (типи БД, а не СУБД).

Вкажіть, який технологічний стек вибрано та обґрунтуйте рішення.

Також виконується опис допоміжних програмних засобів та їх порівняльний аналіз, якщо існує кілька аналогів (мова йде про сторонні бібліотеки, пакети, фреймворки, тощо). Розглядаються засоби розробки за допомогою яких можна виконати вашу розробку, робиться їх порівняльний аналіз та на його основі обираються ті, що вам підходять (маються на увазі мови програмування та IDE).>

## Конструювання програмного забезпечення

<У підрозділі викладають:

* опис оригінальних алгоритмів чи модифікацій існуючих;
* опис структур даних, програмних структур та ін.;
* опис бази даних з представленням концептуальної, логічної чи фізичної моделі та з описом сутностей чи таблиць;
* опис утиліт, бібліотек та іншого стороннього програмного забезпечення, що використовується у розробці.>

### Розробка алгоритму …

<Пункт опціональний, його назва залежить від алгоритму, що розробляється, та/або опису процесів обробки даних.>

Бібліотека xpring, яка виконує запити в мережу блокчейн – однопоточна, тобто не може оброблювати паралельні запити. Через таку специфіку бібліотеки довелось розробляти рішення яке б не блокувало запити з різних потоків, або інших частин коду. За основу ідеї алгоритму була взята багатопоточність самої мови програмування java. Java для вирішення проблем з потоками представляє ділянки коду які можна синхронізувати, що унеможливлює одночасний доступ до нього з різних потоків. Ця синхронізація проходить за допомогою передачі управління об’єкта-монітора. Алгоритм вирішення проблеми з доступом до однопоточної бібліотеки наведено на рисунку 3.1.

### Опис структури бази даних …

В якості системи управління базами даних використовується PostgreSQL. База даних серверу призначена для зберігання користувачів, а також даних про їх …. Опис таблиць бази даних наведено у таблицях 3.1-3.14. Модель бази даних наведена на рисунку 3.2.

Таблиця 3.1 – Опис таблиці user

|  |  |  |
| --- | --- | --- |
| Назва поля | Тип даних | Опис |
| id | serial | ідентифікаційний номер користувача |
| email | varchar | електронна пошта користувача |
| password\_id | int | посилання на запис у таблиці password, де зберігається пароль користувача |

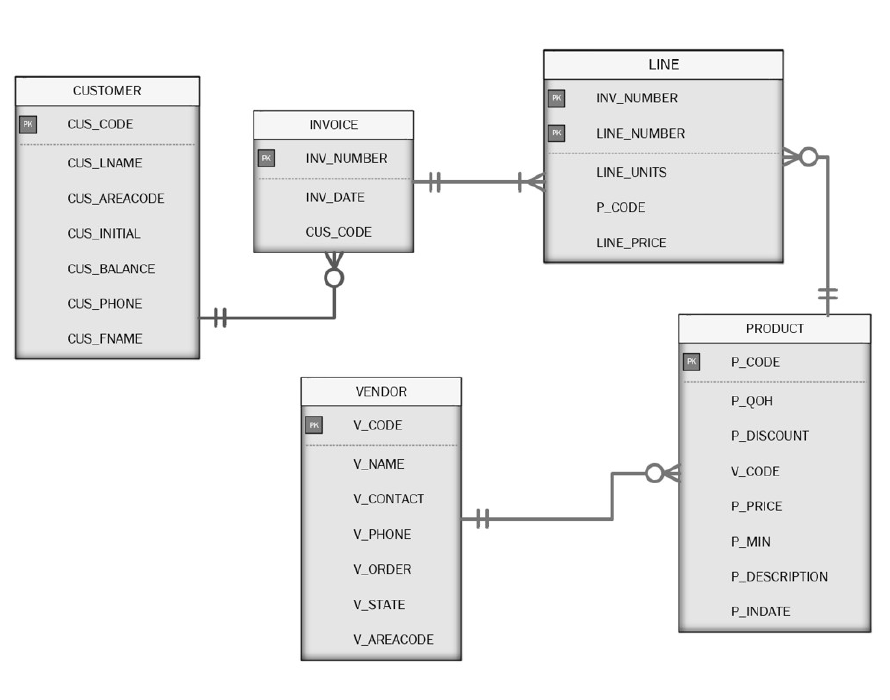


Рисунок 3.2 – ER діаграма сутностей User, Messages, Contacts

Опис утиліт, бібліотек та іншого стороннього програмного забезпечення, що використовується у розробці наведено в таблиці 3.15.

Таблиця 3.15 – Опис утиліт

|  |  |  |
| --- | --- | --- |
| №  п/п | Назва утиліти | Опис застосування |
| 1 | IntelliJ IDEA | Головне середовище розробки програмного забезпечення серверної частини курсової роботи. |
| 2 | Postman | Програмне забезпечення необхідне для тестування rest запитів. Використовувалось для тестування API інтерфейсів, та клієнтських запитів. |
| 3 | MySQL Workbench | Програмне забезпечення яке надає легкий графічний інтерфейс для доступу до бази даних. |

Тексти програмного коду наведені в окремому документі «Текст програми».

## Аналіз безпеки даних

<У підрозділі викладають:

аналіз вразливостей ПЗ та будь-які питання пов’язані з безпекою даних.

Рекомендовані напрямки:

* Vulnerability Management;
* Security Testing;
* Security for Container and Cloud.>

## Висновки до розділу

<Необхідно стисло описати усе, що було виконано у даному розділі. Обсяг 0,75-1 сторінка>

# АНАЛІЗ ЯКОСТІ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## Аналіз якості ПЗ

<У підрозділі викладають:

аналіз якості ПЗ за певними обґрунтованими метриками, перевіряють виконання нефункціональних вимог.>

Метриками для оцінки якості ПЗ обрано наступні:

* цикломатична складність …

## Опис процесів тестування

Тестування виконується згідно документу «Програма та методика тестування».

<У підрозділі викладають:

опис процесів тестування та приклади тестів, ступінь покриття коду тестами, опишіть не більше 15 тест-кейсів.>

Було виконане мануальне тестування програмного забезпечення, опис відповідних тестів наведено у таблицях 4.1 – 4.30.

Таблиця 4.1 – Тест 1.1 Реєстрація користувача

|  |  |
| --- | --- |
| Початковий стан системи | Користувач знаходиться на сторінці реєстрації |
| Вхідні дані | Електронна пошта, пароль, підтвердження паролю |
| Опис проведення тесту | У відповідні поля вводяться: коректна електронна пошта, яка до цього не була зареєстрована в системі, пароль від 10 до 64 символів, який містить хоча б з одну англійську літеру, одне число і один спеціальний символ, і який не входить у топ 10000 найпопулярніших паролей, підтвердження паролю, яке співпадає з раніше введеним паролем. Після цього натискається кнопка підтвердження реєстрації. |
| Очікуваний результат | Реєстрація проходить успішно, користувач додається у систему і перенаправляється на сторінку авторизації. |
| Фактичний результат | Збігається з очікуваним. |

## Опис контрольного прикладу

<У підрозділі викладають:

повний опис контрольного прикладу з усіма можливими розгалуженнями та особливостями. Кроки доповнюють ілюстраціями. Контрольний приклад описує ключовий функціонал.>

## Висновки до розділу

<Необхідно стисло описати усе, що було виконано у даному розділі. Обсяг 0,75-1 сторінка>

# РОЗГОРТАННЯ ТА СУПРОВІД ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## Розгортання програмного забезпечення

<У підрозділі викладають:

Аналіз засобів розгортання (за потреби) і повний опис покрокового розгортання ПЗ. Кроки доповнюють ілюстраціями (діаграма розгортання).>

## Супровід програмного забезпечення

Інструкція користувача / програміста / системного адміністратора наведена в окремому документі (вказати назву документа).

<У підрозділі викладають:

опис того, як буде виконуватись підтримка програмного забезпечення. Ілюстрації.>

Користувачі повинні мати можливість отримати нову версію консольного застосунку з кожною версією. До того ж кожна нова версія консольного застосунку повинна бути опублікована в npm. Для автоматизації цього процесу був використаний сервіс GitHub Actions.

Створення нового випуску починається, коли нова версія консольного застосунку доставляється у репозиторій у гілку main, тобто коли commit має tag формату “v\*.\*.\*.”, де замість “\*” знаходиться число. Тоді у середовищі GitHub Actions встановлюється NodeJS. Після цього для проекту встановлюються залежності і проект збирається. Bash скрипт за допомогою бібліотеки pkg генерує виконувані файли (executables) для Linux і для Windows, та пакує файл для Linux у .deb пакет. Після цього .deb пакет і файл для Windows архівуються,

## Висновки до розділу

<Необхідно стисло описати усе, що було виконано у даному розділі. Обсяг 0,75-1 сторінка>

# ВИСНОВКИ

У висновках викладають наступне:

* що розроблено в ході виконання ДП;
* ступінь досягнення мети;
* стан вирішення усіх поставлених в ДП завдань, в тому числі і завдань розробки ПЗ ДП;
* доцільність продовження розробки за відповідною предметною областю тощо.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. [Оформлення бібліографії ДСТУ 7.1:2006](https://drive.google.com/file/d/1VUr7fwKHOk8t1u8sh-Sv2dlEc_6OBoIS/view?usp=sharing)
2. [Оформлення бібліографії ДСТУ 8302-2015](https://drive.google.com/file/d/17RYEYVnSXolL0S8D1k85VfReYd2yC6WO/view?usp=sharing) з прикладами

# ДОДАТКИ