**План тестування для**

***«Інтелектуальна система для аналізу міського розвитку»***

****

|  |  |
| --- | --- |
| **Загальна інформація** | |
| **Замовник** | *«Інтелектуальна система для аналізу міського розвитку»* |
| **Створено (Автор)** | Проць Вадим |
| **Дата підготовки** | 01.04.25 |
| **Версія** | 1.0 |
| **Статус** | В процесі роботи |

**Історія ревізій**

|  |  |  |  |
| --- | --- | --- | --- |
| **Версія** | **Опис** | **Автор Дата** | **ЗатвЗатверджено**  **Дата** |
| **1.0** | **В процесі** | **Проць Вадим** | **01.04.25** |

Зміст

[1. Вступ 4](#_Toc197178897)

[1.1. Загальна інформація 4](#_Toc197178898)

[1.2. Мета 4](#_Toc197178899)

[2. Масштаб проекту 5](#_Toc197178900)

[2.1. Обсяг веб-порталу 5](#_Toc197178901)

[2.2. Сфера застосування Веб-сайту 5](#_Toc197178902)

[3. План роботи 5](#_Toc197178903)

[4. План та стратегія тестування 6](#_Toc197178904)

[4.1. Функціональне тестування 6](#_Toc197178905)

[4.2. Процедура тестування 7](#_Toc197178906)

[4.3. Звіти про помилки 8](#_Toc197178907)

[5. Ресурси 9](#_Toc197178908)

[5.1. Інструменти 9](#_Toc197178909)

[5.2. Список браузерів 9](#_Toc197178910)

[6. Критерії якості 9](#_Toc197178911)

[7. Ризики процесу тестування 9](#_Toc197178912)

[8. Очікування від команди тестувальників 10](#_Toc197178913)

[9. Обов'язки членів тестової команди 10](#_Toc197178914)

[10. Результати 10](#_Toc197178915)

## 1. Вступ

### 1.1. Загальна інформація

Цей документ описує методи і процедури, які будуть використовуватися командою WEB NOVA у процесі функціонального тестування веб-додатку.

Він призначений для використання в якості інструкції під час проведення робіт з тестування. Описує порядок проведення процесу тестування. План тестування призначений для менеджерів проектів, розробників продукту та QA-інженерів.

Метою тестування є перевірка функцій та можливостей програмного продукту, розробленого для веб-браузерів.

### 1.2. Мета

**Цей документ Плану тестування для Фінансового трекера підтримує наступні цілі:**

* Визначення наявної інформації про проект та програмних компонентів, що підлягають тестуванню.
* Рекомендації та опис стратегій тестування, які будуть використані.
* Вкажіть необхідні ресурси та надайте оцінку зусиль з тестування
* Перелічіть елементи результату тестового проекту.

Результати виконання тестів будуть надіслані замовнику у вигляді звітів. Всі знайдені баги будуть відстежуватися за допомогою баг-трекера Trello.

## 2. Масштаб проекту

### 2.1. Обсяг веб-порталу

**Тестування Фінансового трекера знаходиться в межах цього плану тестування. Будуть протестовані наступні компоненти та функції:**

1. Реєстрація

2. Ввід витрат і балансу

3. Вивід статистики

4. Експорт даних

### 2.2. Сфера застосування Веб-сайту

**Тестування Веб-сайту входить в сферу застосування цього плану тестування. Будуть протестовані наступні компоненти та функції:**

1. Створення облікового запису

2. Введення фінансових даних

3. Редагування даних

4. Переглянути графіки і статистику

5. Експортувати дані

## 3. План роботи

**Сторони домовилися дотримуватися наступного робочого плану:**

1. Підготовка плану тестування

2. Затвердження плану випробувань

3. Функціональне тестування та повідомлення про помилки

5. Підготовка фінального звіту (Можливо)

## 4. План та стратегія тестування

### 4.1. Функціональне тестування

Мета функціонального тестування полягає в тому, щоб переконатися, що весь програмний продукт працює відповідно до вимог, і в додатку не з'являються суттєві помилки.

Функціональне тестування є найбільш значущою частиною тестування програмного забезпечення. Воно передбачає перевірку різних аспектів роботи системи. Програмний продукт повинен пройти всі заплановані тести. Тільки в цьому випадку можна гарантувати його якість.

|  |  |
| --- | --- |
| Мета тестування: | Забезпечити належну функціональність об'єкта тестування |
| Техніка: | Виконайте кожен варіант використання, потік варіантів використання або функцію, використовуючи допустимі та невірні дані, щоб перевірити наступне:   * Очікувані результати з'являються, коли використовуються дійсні дані. При використанні недостовірних даних з'являються відповідні повідомлення про помилку або попередження. * Відповідні повідомлення про помилки або попередження відображаються, коли використовуються невірні дані. * Кожне правило застосовано належним чином. |
| Вхідні критерії | * Розробка додатку завершена. * Виділено інженерів-випробувачів. * Підготовлено тестове середовище, і сайт випущено в тестове середовище. |
| Критерії завершення: | * Виконано всі заплановані тести. * Відсутні помилки, що зупиняють показ. * Виправлено всі помилки високого пріоритету та серйозності. * Результати тестування оцінюються, обговорюються та затверджуються. |
| Особливі міркування: | Ніяких. |

### 4.2. Процедура тестування

**Процедура тестування передбачає наступні пункти:**

* Звітування про знайдені помилки в програмному забезпеченні.

Різні аспекти протестованого програмного забезпечення повинні бути перевірені, для цього необхідно виконати різні типи тестування.

**Основні типи тестування, які будуть виконуватися:**

● Функціональне тестування

● Тестування інтерфейсу користувача

● Тестування сумісності

● Повторне тестування (під час другого раунду, якщо потрібно)

Також буде перевірено, як програмний продукт працює в браузерах і на пристроях, які повинні його підтримувати, як він запускається і зупиняється, і скільки часу йому потрібно для запуску.

**Під час цього раунду тестування наступні типи тестування НЕ застосовуються:**

**● Тестування безпеки**

### 4.3. Звіти про помилки

Звіти про помилки створюються для того, щоб надати команді розробників і менеджеру проекту вичерпну інформацію про виявлені дефекти. Вони повинні бути корисними для визначення причин помилок та їх виправлення.

**Серйозність дефектів можна класифікувати за чотирма категоріями:**

● Критичні (блокувальні) дефекти - це відмова всієї програмної системи або критично важливої підсистеми, і після виникнення дефекту жодна робота або тестування не можуть бути виконані. Це також стосується збоїв, пов'язаних із втратою даних, і процесів, які призводять до того, що в базі даних зберігаються неузгоджені дані.

● Серйозні дефекти (і збої) - це ті, які також спричиняють відмову всієї системи або її частини, але при цьому існують альтернативні варіанти обробки, які дозволяють продовжити роботу системи. Це також стосується збоїв або переривання роботи системи під час нормальної роботи некритичного потоку.

● Незначні дефекти не призводять до відмови, але призводять до того, що система показує неправильні, неповні або суперечливі результати.

● Тривіальні дефекти - це невеликі помилки, які не впливають на функціональність: друкарські помилки, граматичні помилки, неправильна термінологія тощо.

**Інформація, яка вказується в кожному повідомленні про помилку:**

● назва програмного продукту;

● браузер, в якому проводилося тестування.

**Кожен звіт надає наступну інформацію про дефект:**

* резюме, яке є коротким описом проблеми;
* місцезнаходження дефекту в програмному продукті;
* кроки для відтворення помилки;
* частота виникнення дефекту;
* серйозність дефекту;
* додаткову інформацію про дефект у вигляді прикріплених скріншотів

## 5. Ресурси

### 5.1. Інструменти

**У цьому проекті будуть використані наступні інструменти:**

|  |  |
| --- | --- |
| **Назва процесу** | **Інструмент** |
| Відстеження дефектів | **Jira, GitHub Issues** або **Redmine** — для реєстрації, відстеження й управління помилками та завданнями. |
| Тестові кейси | **TestRail, Zephyr**, або **Xray** — для створення, організації та виконання тестових сценаріїв. |
| Скріншоти | **Lightshot, Snipping Tool, ShareX**, або вбудовані інструменти **IDE** (наприклад, у **PyCharm** або **VS Code**) — для фіксації помилок, візуального підтвердження багів чи демонстрації функціоналу. |

### 5.2. Список браузерів

Chromium, Firefox

### 6. Критерії якості

Продукт повинен працювати відповідно до вимог та функціональної специфікації (за наявності).

Продукт не повинен містити критичних та блокуючих дефектів у фінальній версії проекту.

## 7. Ризики процесу тестування

**Наступні проблеми можуть вплинути на роботи з тестування:**

● зміни та модифікації програмного продукту, які не були заплановані та обговорені з командою тестувальників заздалегідь;

● зміни у вимогах до програмного забезпечення, які не були заздалегідь обговорені з командою тестувальників;

● затримки у виправленні/виправленні помилок;

● затримки з наданням нових збірок команді тестувальників.

## 8. Очікування від команди тестувальників

Тестувальник повинен бути забезпечений дійсними оновленими документами протягом усього процесу тестування.

Все необхідне обладнання, інструменти, прилади та програмне забезпечення повинні бути придбані та підготовлені до початку процесу тестування.

Всі помилки, що зупиняють роботу програми, повинні бути виправлені якомога швидше.

До кожного випуску програмного забезпечення слід додавати примітки до релізу для команди тестувальників. Примітка повинна пояснювати, які елементи, функції та можливості були додані до програми і як ці доповнення впливають на програмне забезпечення.

Розробники повинні виправити всі помилки в програмних модулях перед випуском нової версії.

## 9. Обов'язки членів тестової команди

**Керівник проекту**

● Управління всім процесом тестування.

● Забезпечення всіх необхідних ресурсів для проведення тестування.

**Тестувальник**

● Працювати

● Аналіз завдань та розподіл їх між членами команди.

● Спілкування з клієнтською командою та обговорення всіх питань, надання рекомендацій перед оновленням або релізом.

● Створення тестової документації, включаючи тестові кейси, тестові плани тощо.

● Пропонування найкращих практик та інструментів для проекту.

## 10. Результати

* План тестування.
* Звіти про помилки та звіти про хід тестування.