### **Паспорт проєкту: Онлайн-платформа для управління проєктами**

#### **1. Цілі проєкту:**

1.1. Створення сучасної онлайн-платформи для командної роботи з управління проєктами, що забезпечує ефективне планування, моніторинг та контроль виконання завдань.  
1.2. Надання користувачам можливості централізованого управління проєктами в режимі реального часу.  
1.3. Забезпечення функціоналу для взаємодії між членами команди, включаючи інструменти спільної роботи над проєктами.  
1.4. Створення інтуїтивного і зручного інтерфейсу для користувачів.  
1.5. Забезпечення безпеки даних через використання сучасних технологій захисту інформації.  
1.6. Розробка безпечної платформи, що захищає дані користувачів відповідно до міжнародних стандартів (GDPR).

#### **2. Орієнтована тривалість проєкту та можливі етапи:**

2.1. **Орієнтовна тривалість:** 6–7 місяців.

2.2. **Етапи реалізації:**

* **Аналіз і збір вимог:**
  + Збір вимог від потенційних користувачів і стейкхолдерів.
  + Аналіз конкурентів для виявлення унікальних переваг.
* **Проектування:**
  + Створення технічної документації та вибір технологічного стека.
  + Розробка архітектури системи, включаючи клієнтську і серверну частини.
* **Розробка:**
  + Реалізація базового функціоналу (авторизація, створення завдань, управління проектами).
  + Інтеграція API для зовнішніх сервісів.
* **Тестування:**
  + Функціональне тестування всіх модулів платформи.
  + Регресійне тестування після кожного етапу.
* **Впровадження:**
  + Переведення платформи на продакшн середовище.
  + Навчання користувачів і технічної підтримки.

#### **3. Опис продукту проєкту:**

3.1. **Призначення продукту:**

* Надання платформи для створення та управління проєктами, задачами та командними процесами.
* Підвищення ефективності роботи команд завдяки прозорості процесів і швидкому доступу до інформації.

3.2. **Основні функції:**

* Реєстрація та авторизація користувачів із використанням JSON Web Token (JWT).
* Створення, редагування та видалення проєктів.
* Додавання, редагування, видалення завдань у межах проєктів.
* Фільтрація та сортування проєктів за статусами, датами та іншими критеріями.
* Звіти про виконання завдань і статус проєктів.

3.3. **Користувацький досвід:**

* Зручний інтерфейс для командної роботи, розроблений на основі React та CSS.
* Адаптивність інтерфейсу для доступу з різних пристроїв.

3.4. **Безпека даних:**

* Захист від атак типу XSS і CSRF.
* Шифрування паролів і використання HTTPS для захищеного з'єднання.

3.5. **Можливості інтеграції:**

* Інтеграція з календарями для автоматичного планування завдань.
* API для з’єднання з популярними інструментами (Slack, Zoom, Microsoft Teams)

#### **4. Стратегія впровадження:**

4.1. **Організація роботи команди:**

* Визначення ролей і обов'язків кожного учасника.
* Проведення регулярних зустрічей для контролю прогресу.

4.2. **Розробка плану впровадження:**

* Тестування на внутрішній групі користувачів.
* Публічний запуск для широкої аудиторії після коригувань.

4.3. **Навчання:**

* Проведення вебінарів для демонстрації роботи платформи.
* Створення інструкцій для користувачів.

#### **5. Обсяг проєкту:**

5.1. **Цільова аудиторія:**

* Команди малого та середнього бізнесу, які працюють над спільними проектами.
* Фрілансери та індивідуальні підприємці.

5.2. **Обсяг користувачів:**

* Початково – 500 активних користувачів щомісяця.
* Масштабування до 10,000 користувачів у майбутньому.

5.3. **Обсяг даних:**

* Зберігання до 1 ТБ інформації на серверах платформи.

#### **6. Необхідні ресурси проєкту:**

6.1. **Людські ресурси:**

* **Project Manager** – 1.
* **Team Lead** – 1.
* **Розробники:** 2 (Frontend і Backend).
* **Тестувальник:** 2.
* **Бізнес аналітик:** 1.

6.2. **Фінансові ресурси:**Орієнтовний бюджет проєкту – $5,000, що включає:

* Розробку платформи.
* Тестування функціоналу.
* Підтримку системи.

6.3. **Матеріальні ресурси:**

* Сервери для хостингу (DigitalOcean).
* Ліцензії на програмне забезпечення (Visual Studio, MongoDB тощо).

#### **7. Технічний обсяг:**

7.1. **Архітектура:**

* Клієнт-серверна модель.
* Серверна частина на Node.js із використанням Express.js.

7.2. **База даних:** MongoDB.  
7.3. **Мови та технології:**

* **Frontend:** React, CSS.
* **Backend:** Node.js, Express.js.

#### **8. Можливі ризики:**

8.1. Відставання від графіка через непередбачувані технічні складнощі.  
8.2. Підвищене навантаження на сервери через велику кількість користувачів.  
8.3. Можливі вразливості в системі безпеки.

#### **9. Забезпечення якості:**

9.1. Регулярне функціональне та регресійне тестування.  
9.2. Автоматизоване тестування ключових компонентів платформи.  
9.3. Залучення реальних користувачів для тестування інтерфейсу.

#### **10. Терміни реалізації:**

* **Аналіз і проектування:** 2 місяці.
* **Розробка та інтеграція:** 3 місяці.
* **Тестування:** 1 місяць.

#### **11. Управління змінами обсягу проекту:**

11.1. Ідентифікація, фіксація та оцінка змін у встановлених вимогах;  
11.2. Оцінювання впливу змін на фінансові витрати, строки реалізації та використання ресурсів;  
11.3. Прийняття рішень щодо впровадження змін, їх відкладення або відмови від них;  
11.4. Оновлення документації та забезпечення інформування всіх зацікавлених сторін про внесені корективи.

**12. Формування бази знань:**

12.1. Створення централізованого репозиторію знань для зберігання ключових документів, інструкцій і технічної інформації.

12.2. Забезпечення доступу до бази знань для всіх членів команди відповідно до рівнів доступу.

12.3. Регулярне оновлення бази знань на основі змін у вимогах, нових розробок і зворотного зв’язку від користувачів.

12.4. Автоматизація процесу додавання нових даних через інтеграцію із системами управління завданнями.