**Постановка задачи. Формирование требований к ИС**

**1. Заинтересованные лица:**

**Product Owner (Владелец продукта):**

* Управление бэклогом продукта.
* Контроль процесса разработки
* Оценка результатов проекта.
* Управление пользователями и ролями.

**Scrum Master:**

* Организация и проведение SCRUM-мероприятий (планирование, стендапы, ретроспективы).
* Контроль соблюдения SCRUM-процессов.
* Приоритизация задач
* Оценка результатов спринтов.
* Мониторинг прогресса проекта.
* Анализ метрик (скорость команды, выполнение спринтов).
* Принятие решений на основе отчетов.

**Разработчики:**

* Выполнение задач в рамках спринта.
* Отслеживание прогресса на доске задач.
* Участие в SCRUM-мероприятиях.

**2. Функциональные требования:**

**Для Product Owner:**

* Создание и редактирование элементов бэклога проекта.
* Возможность оценки спринтов.
* Просмотр отчетов о выполнении спринтов.
* Мониторинг прогресса команды
* Приглашение разработчиков в проект по ролям
* Регистрация в системе

**Для Scrum Master:**

* Организация SCRUM-мероприятий через систему (планирование, стендапы, ретроспективы).
* Управление беклога спринтов
* Управление доской задач (Kanban/Scrum board).
* Приглашение разработчиков в проект по ролям
* Мониторинг прогресса команды.
* Просмотр аналитических отчетов
* Регистрация в системе

**Для разработчиков:**

* Просмотр задач, назначенных на спринт.
* Обновление статуса задач (To Do, In Progress, Done).
* Участие в SCRUM-мероприятиях через систему.
* Регистрация в системе
* Получать на почту письмо с уведомлением о новых заданиях

**3. Системные требования:**

СУБД: PostgreSQL.

Языки программирования: Java, TypeScrypt и JavaScrypt.

Фронтенд: React.ts.

Веб-сервер: Nginx.

Сервер: Tomcat.

Контейнеризация: Docker.

**4. Нефункциональные требования:**

**Требования к графическому интерфейсу:**

* Интуитивно понятный и минималистичный дизайн.
* Поддержка темной и светлой темы.
* Кроссбраузерная разработка

**Требования к безопасности:**

* Аутентификация и авторизация пользователей (JWT).
* Шифрование данных при передаче (HTTPS).

**Требования к данным:**

* Хранение данных в структурированном виде.
* Формат данных JSON, HTML.

**5. Постановка задачи:**

**Цель:**

Спроектировать информационную систему для управления проектами с использованием методологии SCRUM, которая обеспечит автоматизацию процессов, повысит прозрачность и эффективность работы команд.

**Задачи:**

* Разработать функциональные модули для управления бэклогом, планирования спринтов и отслеживания задач.
* Реализовать аналитические инструменты для оценки производительности команды.
* Разработать удобный и адаптивный пользовательский интерфейс.
* Обеспечить безопасность данных и отказоустойчивость системы.

**Исходные данные:**

* Описание процессов SCRUM.
* Примеры существующих решений (Jira, Trello).

**Априорные модельные представления**:

* Единая платформа для управления проектами по SCRUM.
* Автоматизация рутинных процессов.

**Результат:**

Проект информационной системы, который будет соответствовать априорным модельным представлениям, включающий:

* Функциональные модули.
* Дизайн интерфейса.
* Описание архитектуры и технологического стека.
* Документацию по использованию и настройке.

**Критерии оценки результата:**

* Сокращение времени на планирование спринтов на 30%.
* Увеличение прозрачности процессов на 50%.
* Снижение количества ошибок при оценке задач на 20%.
* Улучшение скорости выполнения задач на 15%.

**Средства реализации:**

1. **CASE-средства:**

* Диаграммы UML (Use Case, Sequence, Activity).
* ER-диаграммы для проектирования базы данных.

1. **Инструменты разработки:**

* IDE: Visual Studio Code, IntelliJ.
* Система контроля версий: Git.
* Фреймворки: Express.js (Node.js), Spring Core, TailwindCSS, Flowbite
* Библиотеки фронтенда: React, Axios
* Сборщик: Vite