**Техническая документация приложения "Student Organizer"**

**Список требований в формате product backlog.**

**Версия документации:** 0.1  
**Дата:** 18 января 2025 года  
**Авторы:** Корчуганов Максим, Костюкевич Кирилл, Чикида Данила, Мицкевич Кирилл.

**Product Backlog(список требований)**

**Этап 1: Инициализация проекта**

1. Инициализировать проект.
2. Создать базовую структуру проекта на Node.js и Electron.
3. Настроить основные зависимости Electron.

**Этап 2: Расписание**

1. Добавить интерфейс для расписания.
2. Реализовать функциональность добавления занятий.
3. Поддержка примечаний для каждого занятия.
4. Добавить поддержку номера недели.
   * Выпадающий список для выбора первой или второй недели в расписании.
5. Добавить выбор дня недели.
   * Выпадающий список для выбора дня недели (понедельник–суббота).

**Этап 3: Напоминания**

1. Реализовать напоминания.
2. Добавить функциональность создания напоминаний с дедлайнами.

**Этап 4: Успеваемость**

1. Учет успеваемости.
   * Добавить возможность вводить оценки с комментариями.
2. Вывод статистики по оценкам (в будущем).

**Этап 5: Модальные окна**

1. Интеграция модальных окон.
   * Добавить модальные окна для ввода данных (занятия, напоминания, оценки).

**Этап 6: Валидация данных**

1. Валидация введенных данных.

**Этап 7: Сохранение и загрузка данных**

1. Сохранение данных.
   * Реализовать сохранение расписания, напоминаний и оценок с помощью API.
2. Загрузка данных.
   * Реализовать загрузку ранее сохраненных данных при старте приложения.

**Этап 8: Удаление данных**

1. Реализовать возможность удаления записей.
   * Добавить кнопки удаления для расписания, напоминаний и оценок.

**Этап 9: Стилизация интерфейса**

1. Добавить стилизацию интерфейса.
   * Обновить внешний вид с помощью CSS (если требуется).

**Этап 10: Устойчивость данных**

1. Обеспечить устойчивость данных.
   * Обработать возможные ошибки при сохранении/загрузке данных.

**Product Backlog(user story)**

|  |  |
| --- | --- |
| **ID** | **User Story** |
| a-001 | Как разработчик, я хочу инициализировать проект, чтобы начать работу над приложением. |
| a-002 | Как разработчик, я хочу создать базовую структуру проекта на Node.js и Electron, чтобы заложить основу приложения. |
| a-003 | Как разработчик, я хочу настроить основные зависимости Electron, чтобы обеспечить функциональность приложения. |
| a-004 | Как пользователь, я хочу иметь интерфейс для расписания, чтобы удобно планировать занятия. |
| a-005 | Как пользователь, я хочу добавлять занятия, чтобы составить своё расписание. |
| a-006 | Как пользователь, я хочу добавлять примечания к каждому занятию, чтобы сохранять важные детали. |
| a-007 | Как пользователь, я хочу получать напоминания, чтобы не забывать о предстоящих занятиях. |
| a-008 | Как пользователь, я хочу создавать напоминания с дедлайнами, чтобы управлять своими задачами. |
| a-009 | Как пользователь, я хочу вводить оценки с комментариями, чтобы учитывать успеваемость. |
| a-010 | Как пользователь, я хочу видеть статистику по оценкам, чтобы отслеживать свои достижения (в будущем). |
| a-011 | Как пользователь, я хочу использовать модальные окна для ввода данных, чтобы взаимодействие с интерфейсом было удобным. |
| a-012 | Как пользователь, я хочу, чтобы данные проходили валидацию, чтобы избежать ошибок при вводе. |
| a-013 | Как пользователь, я хочу выбирать номер недели, чтобы настроить расписание на определённую неделю. |
| a-014 | Как пользователь, я хочу выбирать день недели, чтобы добавлять занятия для конкретного дня. |
| a-015 | Как пользователь, я хочу сохранять расписание, напоминания и оценки через API, чтобы мои данные не терялись. |
| a-016 | Как пользователь, я хочу загружать сохранённые данные при старте приложения, чтобы продолжить работу. |
| a-017 | Как пользователь, я хочу удалять записи, чтобы управлять расписанием, напоминаниями и оценками. |
| a-018 | Как пользователь, я хочу видеть стильный интерфейс, чтобы приложение выглядело современно. |
| a-019 | Как разработчик, я хочу обработать возможные ошибки при сохранении и загрузке данных, чтобы обеспечить устойчивость приложения. |