## **Проект: BookManager – Личное приложение для управления библиотекой**

### **Часть 1: Обзор проекта**

#### **Выбор проекта:**

**BookManager** – десктопное приложение для управления личной библиотекой, позволяющее добавлять книги в формате PDF, автоматически генерировать их краткое содержание с помощью ИИ, запоминать последнюю позицию чтения и разделять книги на категории ("Прочитанные" и "Запланированные").

#### **Анализ проблемы:**

1. **Проблема:** Многие читатели сталкиваются с трудностями в управлении своими книгами, особенно в формате PDF. Запоминание последней страницы, подготовка краткого содержания или организация коллекции отнимают время.
2. **Почему проект важен и актуален:** С ростом популярности электронных книг и PDF-формата людям нужны инструменты для эффективного управления библиотекой и повышения удобства чтения.
3. **Целевая аудитория:** Книголюбы, студенты и исследователи в возрасте от 16 до 50 лет, активно работающие с текстами в формате PDF.

#### **Обзор конкурентов:**

1. **Конкурент 1:** **Calibre**
   * **Сильные стороны:** Поддержка множества форматов книг, управление коллекцией.
   * **Слабые стороны:** Нет встроенной функции создания аннотаций или запоминания прогресса чтения.
2. **Конкурент 2:** **Goodreads**
   * **Сильные стороны:** Возможность делиться списками книг и отзывами.
   * **Слабые стороны:** Невозможно загружать собственные PDF-файлы.
3. **Конкурент 3:** **Zotero**
   * **Сильные стороны:** Удобен для исследовательских работ и цитирования.
   * **Слабые стороны:** Сложный интерфейс, нет функций для обобщения содержания.

### **Часть 2: Постановка целей и задач**

#### **Цели:**

1. Создать бесплатное и удобное приложение для управления личной библиотекой в формате PDF.
2. Реализовать функции автоматизации, включая запоминание последней позиции чтения и генерацию краткого содержания.

#### **Задачи:**

1. Реализовать добавление книг в формате PDF.
2. Настроить функционал для запоминания последней страницы.
3. Создать систему категорий: "Прочитанные" и "Запланированные".
4. Интегрировать ИИ для создания краткого содержания.
5. Разработать систему экспорта данных о библиотеке в CSV/JSON.

### **Часть 3: Планирование этапов разработки**

#### **Этапы проекта:**

1. **Этап 1: Анализ требований и проектирование (1 неделя):**
   * Задачи: сбор требований, выбор технологий, создание макета интерфейса, проектирование базы данных.
   * Сроки: 1 неделя.
2. **Этап 2: Реализация базового функционала (2 недели):**
   * Задачи:
     + CRUD-операции для книг.
     + Добавление и управление категориями ("Прочитанные", "Запланированные").
     + Настройка SQLite для хранения информации.
   * Сроки: 2 недели.
3. **Этап 3: Подключение PDF и ИИ (2 недели):**
   * Задачи:
     + Реализация функции открытия PDF и запоминания последней страницы.
     + Интеграция модели GPT для создания краткого содержания.
   * Сроки: 2 недели.
4. **Этап 4: Тестирование и оптимизация (1 неделя):**
   * Задачи: тестирование функций, исправление багов, оптимизация работы с большими PDF.
   * Сроки: 1 неделя.
5. **Этап 5: Подготовка документации и выпуск (1 неделя):**
   * Задачи: создание пользовательской документации, выпуск готового приложения.
   * Сроки: 1 неделя.

### **Риски и способы их снижения:**

1. **Риск:** Ограничение в обработке больших PDF-файлов.  
    **Решение:** Ограничить объем текста, обрабатываемого ИИ, добавить прогресс-бар.
2. **Риск:** Ошибки в запоминании последней позиции.  
    **Решение:** Использовать надёжный механизм сохранения прогресса через SQLite.
3. **Риск:** Сложности с интеграцией ИИ.  
    **Решение:** Использовать проверенные библиотеки и API (например, OpenAI GPT или Hugging Face).

### **Контроль выполнения:**

1. Ежедневные отчёты о прогрессе.
2. Еженедельные контрольные точки после завершения каждого этапа.

### **Результат работы:**

* Краткий обзор проекта, включая анализ проблемы и аудитории.
* Детализированный план разработки с этапами и сроками.
* Список рисков и способов их устранения.
* Готовый отчёт в формате Word или PDF.

