**ЛАБОРАТОРНАЯ РАБОТА №1**

**Курс «Технологии разработки программного обеспечения»**

**Тема:** Составление спецификаций требований заказчика программного продукта.

**Цель: научиться** формализовать требования заказчика программного продукта и оформлять техническое задание на создание программного продукта.

**Примечание к лабораторным работам.**

В рамках лабораторных работ №1-№5 студент занимается разработкой одного программного продукта. В роли заказчика продукта выступает проработчик курса «Технологии создания программного продукта» на кафедре компьютерных технологий. Каждому студенту необходимо разработать простейшую информационную систему по определенной предметной области. Предметную область студент может либо предложить самостоятельно (например, совпадающую с тематикой его курсовой работы), либо получить по варианту (приложение 1.1). Предметные области у разных студентов не могут повторяться.

**I. Общие сведения и цели:**

* **Что это за приложение и какую проблему оно решает?** (Основная цель, ниша)

Приложение для клуба любителей чтения. Там будет основная библиотека с электронными книгами от известных писателей классики и современности. Обсуждения под каждой книгой.

* **Какова целевая аудитория приложения?** (Демография, потребности, техническая грамотность)

Целевая аудитория это любители читать книги, и делиться впечатлениями. Они ищут единомышленников. Так же смотря отзывы на произведения решаются прочитать.

* **Какие задачи пользователи будут выполнять с помощью приложения?** (Основные сценарии использования)
  1. Искать любимые книги
  2. Искать единомышленников
  3. Длиться впечатлениями от прочитанного
  4. Читательский дневник
  5. Искать новые любимые книги
* **Какие бизнес-цели преследуются при создании приложения?** (Прибыль, узнаваемость бренда, автоматизация)
  1. Привлекать людей читать

**II. Функциональность и особенности:**

**Какие функции обязательны, а какие желательны?**

Основная функция — это библиотека с онлайн книгами. Комментарии под каждой книгой. Личный дневник пользователя

* **Будет ли приложение работать онлайн/оффлайн?** (Синхронизация данных)

Приложение должно быть онлайн

* **Нужны ли push-уведомления, геолокация, авторизация, личный кабинет?**
  1. Авторизация
  2. Личный кабинет с читательским дневником
  3. Уведомления об ответе на сообщение или добавленных книг любимого автора.
* **Как будет организован поиск и фильтрация данных?**

Да будет. Поиск книг по названию или автору. Поиск пользователей по никнейму.

* **Нужно ли многоязычие?**

Если книга представляет перевод с иностранного языка, то указать его оригинальное название на родном языке (если это не английский, то указать еще, английский вариант)

**IV. Технические вопросы:**

* **На каких платформах будет разрабатываться приложение (iOS, Android, Web)?**
* **Нужна ли интеграция с другими сервисами (CRM, платежные системы, социальные сети)?**
  1. Ссылки на аккаунты в других соцсетях
* **Где будут храниться данные (облако, локальный сервер)?**
* **Какие требования к безопасности данных?**

Позаботиться об информации пользователя. Его личные данные

**Как будет происходить тестирование и развертывание приложения?**

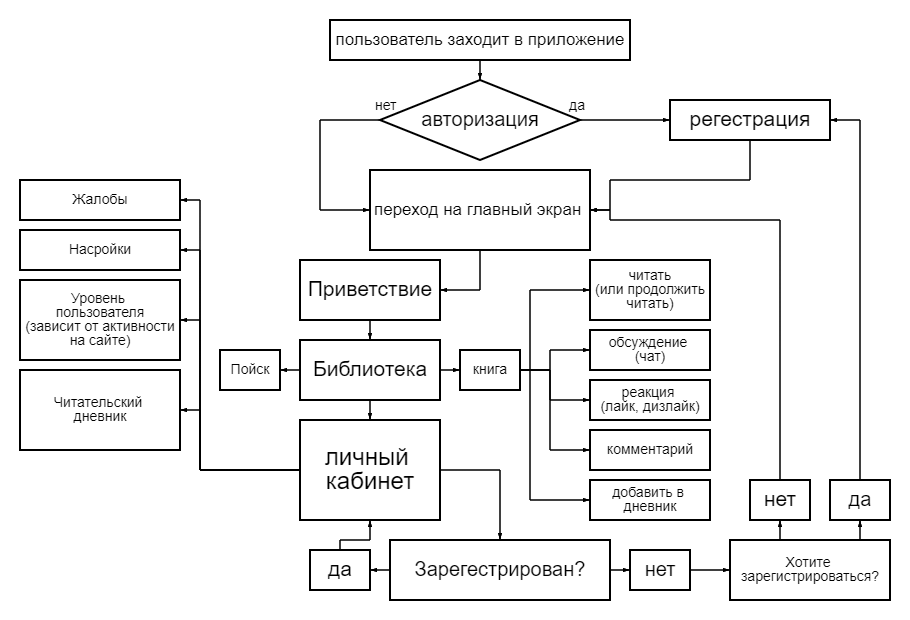
**V. Поддержка и развитие:**

* **Как будет осуществляться техническая поддержка приложения после запуска?**

Нанятый специалист будет фиксить возникающие ошибки и проблемы. Модераторы будут следить за правилами и цензурой в комментариях.

* **Планируется ли дальнейшее развитие и добавление новых функций?**

По мере использования приложения будут добавляться новые функции.



1.

Основные этапы разработки программного продукта:

• Сбор требований: Определение функциональных и нефункциональных требований.

• Анализ: Анализ требований и создание спецификаций.

• Проектирование: Разработка архитектуры и дизайна системы.

• Реализация: Написание кода и его тестирование.

• Тестирование: Проверка и верификация программного продукта.

• Внедрение: Развертывание системы в рабочей среде.

• Сопровождение: Поддержка и обновление системы.

2.

Артефакты программного продукта:

• Документация (техническая, пользовательская)

• Исходный код

• Тестовые сценарии и результаты тестирования

• Проектные документы (диаграммы, спецификации)

• Исполняемые файлы и библиотеки

3.

Основные стратегии разработки ПО:

• Водопадная модель: Последовательное выполнение этапов.

• Итеративная модель: Разработка через повторяющиеся циклы.

• Гибкие методологии (Agile): Инкрементальная и итеративная разработка с акцентом на гибкость и адаптацию.

• Модель V: Расширение водопадной модели с акцентом на тестирование на каждом этапе.

4.

Системы принципов разработки ПО:

• PSP (Personal Software Process): Индивидуальный процесс разработки, направленный на улучшение личной производительности.

• TSP (Team Software Process): Командный процесс разработки, направленный на улучшение производительности команды.

• CMM (Capability Maturity Model): Модель зрелости процессов, оценивающая уровень зрелости процессов разработки в организации.

5.

Состав персонала разработки программного продукта:

• Менеджер проекта

• Аналтик требований

• Архтектор системы

• Разработчики

• Тестировщики

• Технический писатель

• Специалист по DevOps

6.

Основные метрики качества программного продукта:

• Количество дефектов

• Покрытие кода тестами

• Время отклика системы

• Надежность и устойчивость к сбоям

• Удобство использования (usability)

• Производительность

7.

Процесс инспектирования проекта:

• Инспектирование включает в себя проверку артефактов проекта на соответствие требованиям и стандартам.

• Принципы инспектирования:

• Независимость инспекторов

• Сисематичность и планирование

• Документирование результатов

• Обратная связь и корректирующие действия

8.

Примеры функциональных и нефункциональных требований:

• Функциональные требования: Возможность регистрации пользователей, добавление и удаление записей, генерация отчетов.

• Нефункциональные требования: Производительность системы, безопасность данных удобство интерфейса, совместимость с другими системами.

9.

Варианты использования программного продукта:

• Варианты использования (use cases) описывают взаимодействие пользователя с системой для достижения определенной цели. Они помогают определить функциональные требования и сценарии использования.

10.

Схема процесса анализа С-требований:

• Сбор требований

• Анализ и уточнение требований

• Документирование требований

• Верификация и утверждение требований

11.

Схема процесса анализа D-требований:

• Определение архитектуры системы

• Разработка детальных проектных решений

• Создание проектной документации

• Верификация проектных решений