

Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования «Уральский федеральный университет имени первого Президента России Б. Н. Ельцина» (УрФУ)

Институт радиоэлектроники и информационных технологий – РТФ

ОТЧЕТ

о проектной работе

по теме: \_\_\_Планировщик задач\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

по дисциплине: Проектный практикум 1A

Команда:\_\_\_\_\_Mukero Team’s\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Тимлид: Коновалов Ростислав Андреевич РИ-130913

Аналитик: АндреевичАгаев Турабхан Рафаил Оглы РИ-130942

Дизайнер: Воробьев Евгений Алексеевич РИ-130913

Фронтенд-разработчик: Левашев Максим Александрович РИ-130941

Бэкенд-разработчик: Соболев Василий Витальевич РИ-130913

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc165369900)

[1. Целевая аудитория 5](#_Toc165369901)

[2. Определение проблемы 8](#_Toc165369902)

[3. Подходы к решению проблемы 12](#_Toc165369903)

[4. Анализ аналогов 15](#_Toc165369904)

[5. Календарный план проекта 19](#_Toc165369905)

[6. Сценарии использования 21](#_Toc165369906)

[7. Требования к продукту и к MVP 23](#_Toc165369907)

[8. Стек для разработки 27](#_Toc165369908)

[9. Прототипирование 29](#_Toc165369909)

[10. Проектирование и разработка системы 33](#_Toc165369910)

[ЗАКЛЮЧЕНИЕ 36](#_Toc165369911)

ВВЕДЕНИЕ

* Актуальность работы. В современном мире люди сталкиваются с огромным количеством информации и задач, которые необходимо выполнять ежедневно. Эффективное управление временем и задачами становится важным элементом успешной жизни, как в профессиональной, так и в личной сфере. Традиционные методы планирования задач, такие как бумажные списки или простые приложения, часто не удовлетворяют потребности пользователей в гибкости и безопасности. Существует необходимость в создании интуитивно понятного и минималистичного планировщика задач, который будет удовлетворять высокие требования к удобству использования и конфиденциальности данных.
* Проблемная ситуация. Существующие на рынке приложения для управления задачами часто перегружены ненужными функциями, имеют сложные интерфейсы и требуют постоянного интернет-соединения для полноценной работы. Это приводит к высокой когнитивной нагрузке на пользователей и снижает их эффективность. Вопросы конфиденциальности также остаются нерешенными: многие популярные приложения собирают и используют данные о пользователях в коммерческих целях, что вызывает недоверие и опасения у пользователей.
* Граница между знанием и незнанием. Несмотря на обилие приложений для управления задачами, остается проблема создания такого продукта, который бы сочетал в себе простоту, функциональность и безопасность. Необходимы исследования и разработки, направленные на создание планировщика задач, который бы удовлетворял требованиям широкой аудитории, был интуитивно понятным и не нарушал конфиденциальность данных пользователей.
* Цель работы – разработка инновационного планировщика задач TaskCraft, который будет эффективным инструментом для управления задачами и временем, обеспечивая простоту использования, минимализм и защиту данных пользователей.

Задачи проекта:

* Изучить текущие решения на рынке планировщиков задач и выявить их основные недостатки.
* Описать целевую аудиторию и их потребности в области управления задачами.
* Установить основные функциональные и нефункциональные требования к планировщику задач TaskCraft.
* Выявить и описать ключевые проблемы, связанные с конфиденциальностью данных в существующих решениях.
* Разработать прототип интерфейса планировщика задач, учитывая требования целевой аудитории.
* Создать минимально жизнеспособный продукт (MVP) TaskCraft и протестировать его на реальных пользователях.

Составить план дальнейшего развития и улучшения продукта на основе полученной обратной связи.

1. Целевая аудитория

Для определения целевой аудитории использовалась методика 5W Марка Шеррингтона. Данный метод позволяет сегментировать рынок и выявить основные характеристики потенциальных потребителей.

а) Что? (What?)

TaskCraft – это планировщик задач, предоставляющий пользователям следующие возможности:

Простое и быстрое добавление задач.

Интуитивно понятный интерфейс.

Напоминания о важных делах.

Установка приоритетов для задач.

Минимум отвлекающих элементов и рекламы.

Возможность работы оффлайн.

Поддержка темного режима.

Группировка задач по проектам или категориям.

Синхронизация задач между устройствами.

б) Кто? (Who?)

Основные характеристики пользователей:

Группы пользователей: офисные работники, студенты, фрилансеры.

Устройства: большинство пользователей используют Android-устройства, реже – iOS и компьютеры/ноутбуки.

в) Почему? (Why?)

Мотивация и потребности пользователей:

Необходимость в простом и удобном инструменте для управления задачами.

Повышение личной эффективности и продуктивности.

Снижение когнитивной нагрузки благодаря минималистичному интерфейсу.

Обеспечение конфиденциальности данных (многие пользователи не согласны на сбор данных о их активности).

г) Когда? (When?)

Ситуации использования TaskCraft:

Ежедневное отслеживание всех дел.

Регулярные проверки списка задач.

Периодическое обновление статуса задач.

Планирование задач утром и вечером.

Управление проектами и задачами на протяжении дня.

д) Где? (Where?)

Основные точки контакта с клиентом:

TaskCraft используется в различных условиях: дома, на работе, в учебных заведениях и других местах.

Приложение должно быть доступно как онлайн, так и оффлайн.

Выводы:

Пользователи ценят простоту и интуитивность интерфейса TaskCraft.

Ключевыми функциями являются напоминания, возможность установки приоритетов и работа оффлайн.

Пользователи предпочитают избегать перегруженных интерфейсов с множеством ненужных функций и рекламой.

TaskCraft должен обеспечивать конфиденциальность данных, так как большинство пользователей не согласны на сбор данных.

Приложение должно быть доступно на различных устройствах, включая Android, iOS и компьютеры.

Эти выводы позволяют создать продукт, который максимально удовлетворяет потребности и ожидания целевой аудитории, обеспечивая при этом простоту и удобство использования.

1. Определение проблемы

Определение потребностей и болей целевой аудитории для планировщика задач TaskCraft является важным этапом разработки продукта. Для этого используются различные методы маркетинговых исследований, которые позволяют выявить основные проблемы клиентов.

Методы определения проблем

Опросы и интервью:

Проведение опросов и интервью с пользователями планировщиков задач помогает получить прямую обратную связь. Это позволяет узнать, какие функции они считают наиболее важными, а какие – лишними. Например, из нашей статистики следует, что пользователи ценят простоту и быстрый доступ к функциям, но недовольны перегруженностью интерфейсов и излишними рекламными уведомлениями.

Изучение конкурентов:

Анализ конкурентов, таких как Todoist, Microsoft To Do, Trello, Notion и Any.do, позволяет выявить их сильные и слабые стороны. На основании этого анализа можно определить, какие проблемы остаются нерешенными, и как TaskCraft может предложить лучшие решения. Например, многие пользователи отмечают, что существующие приложения перегружены функциями и не обеспечивают достаточного уровня конфиденциальности.

Тематические форумы:

Тематические форумы являются важным источником информации о проблемах пользователей. Здесь можно найти обсуждения о том, какие функции нужны в планировщике задач, а какие считаются ненужными или неудобными. Например, пользователи часто жалуются на сложные настройки и множество ненужных опций.

Социальные сети:

Социальные сети являются быстрым и надежным способом получения обратной связи. Пользователи активно делятся своими впечатлениями о планировщиках задач, что позволяет оперативно реагировать на их потребности. Например, многие пользователи отмечают необходимость в поддержке оффлайн-режима и минималистичном дизайне.

Отзывы:

Изучение отзывов на специализированных сайтах позволяет понять, с какими проблемами сталкиваются пользователи при использовании существующих приложений для управления задачами. Это дает возможность улучшить TaskCraft, учитывая реальные потребности аудитории.

Анализ поисковых запросов:

Анализ поисковых запросов через сервисы Wordstat и Google Trends позволяет выявить, что именно ищут пользователи, какие вопросы задают и какие проблемы пытаются решить. Например, пользователи часто ищут "простой планировщик задач" или "планировщик задач без рекламы".

Классификация проблем клиентов

Проблемы клиентов можно условно разделить на три основные категории:

Отсутствие ресурсов:

Время, деньги, умения. Например, пользователи, у которых нет времени на изучение сложных приложений, нуждаются в интуитивно понятном интерфейсе.

Неудачные попытки решения проблемы:

Пользователи, которые уже пробовали использовать другие планировщики задач, но остались недовольны их сложностью или перегруженностью функциями.

Недовольство своим состоянием:

Пользователи, которые хотят повысить свою продуктивность и организованность, но не могут найти подходящий инструмент.

Уровни «боли»

Технический уровень:

Наличие неудобств, вызванных конкретными проблемами, такими как медленная работа приложений или отсутствие оффлайн-режима. Вопрос: "Что создало эту проблему?"

Финансовый уровень:

Проблемы, связанные с финансовыми потерями или неэффективностью, такие как платные функции, которые не оправдывают своей стоимости. Вопрос: "Как это вредит?"

Личный интерес:

Личная мотивация клиента на решение проблемы, например, желание сэкономить время и улучшить свою продуктивность. Вопрос: "Зачем нужно устранить проблему?"

Выводы

Понимание проблем целевой аудитории позволяет более точно определить требования к TaskCraft и создать продукт, который будет максимально удовлетворять потребности пользователей. Использование различных методов маркетинговых исследований и анализ полученных данных помогает выявить основные проблемы клиентов и разработать планировщик задач, решающий эти проблемы наиболее эффективно.

1. Подходы к решению проблемы

Для решения проблем, выявленных в процессе анализа целевой аудитории и конкурентов, предлагается использовать многообразие подходов и методов. Это поможет найти наилучшие решения и обеспечить высокое качество конечного продукта TaskCraft.

Мозговые штурмы

Первым шагом является проведение мозговых штурмов с проектной командой. Это позволяет генерировать большое количество идей и предложений, которые затем можно тщательно анализировать и отбирать наиболее перспективные.

Интервьюирование заказчиков и пользователей

Интервью с заказчиками и пользователями дает возможность получить непосредственную обратную связь о текущих проблемах и потребностях. Это позволяет лучше понять, какие функции и особенности важны для целевой аудитории, и как они воспринимают существующие решения.

Методики ТРИЗ (Теория Решения Изобретательских Задач)

Использование методик ТРИЗ помогает систематически подходить к решению сложных задач и генерировать инновационные идеи. Это позволяет выявлять противоречия в существующих системах и предлагать оригинальные решения, которые могут значительно улучшить продукт.

Итеративный подход

Подготовка нескольких альтернативных решений на ранних стадиях разработки позволяет повысить вероятность принятия оптимального решения. На каждой итерации проводится анализ полученных данных и корректировка предложений. Это помогает лучше понять факторы, влияющие на принятие решения, и уточнять аналитические параметры, такие как критерии и их веса.

Структурированный метод анализа

Структурированный метод анализа позволяет оценивать каждое из предложенных решений по набору критериев. Это помогает объективно сравнивать различные варианты и выбирать наиболее подходящее решение. В случаях, когда результаты «взвешивания» нескольких решений оказываются близкими, проводится дополнительный анализ с привлечением экспертов и заинтересованных сторон.

Примеры решений для TaskCraft

Оптимизация интерфейса:

Разработка минималистичного интерфейса, который позволит пользователям быстро и легко добавлять и управлять задачами. Это включает в себя сокращение количества шагов для выполнения основных действий и устранение ненужных функций.

Поддержка оффлайн-режима:

Внедрение функции оффлайн-доступа, чтобы пользователи могли управлять своими задачами без необходимости постоянного интернет-соединения. Это особенно важно для тех, кто часто находится в местах с ограниченным доступом к интернету.

Конфиденциальность данных:

Обеспечение высокого уровня конфиденциальности данных пользователей. Это включает в себя минимизацию сбора данных и прозрачную политику конфиденциальности, что позволит пользователям доверять приложению и активно им пользоваться.

Автоматизация процессов:

Разработка автоматических напоминаний и уведомлений, чтобы пользователи не забывали о важных задачах и мероприятиях. Это поможет увеличить продуктивность и организацию времени.

Пример: Сокращение времени на управление задачами

Чтобы сократить время на управление задачами в TaskCraft, можно:

Оптимизация интерфейса:

Сокращение количества шагов для выполнения основных действий, таких как добавление или редактирование задач. Это поможет пользователям быстрее выполнять свои задачи без лишних действий.

Поддержка оффлайн-режима:

Внедрение функций, которые позволяют пользователям работать с задачами без необходимости постоянного интернет-соединения. Это сделает приложение более удобным для использования в любых условиях.

Конфиденциальность данных:

Обеспечение защиты данных пользователей, минимизация сбора данных и внедрение прозрачной политики конфиденциальности. Это позволит пользователям чувствовать себя защищенными при использовании приложения.

Разработка оптимальных решений требует понимания текущего состояния процессов и работы пользователей. Это позволяет не только устранить существующие проблемы, но и избежать создания новых. Путем тщательного анализа и итеративного подхода можно разработать продукт, который будет максимально соответствовать потребностям пользователей и обеспечивать высокую эффективность их работы.

1. Анализ аналогов

Прямые конкуренты – это аналогичное программное обеспечение на аналогичном рынке и работающее с той же целевой аудиторией, что и разрабатываемый продукт. Косвенные конкуренты – это похожее программное обеспечение с другими характеристиками или абсолютно другой продукт, но работающее с той же целевой аудиторией.

Шаги анализа конкурентов

Определить цели анализа конкурентов:

Выявить сильные и слабые стороны конкурентов.

Определить возможности для улучшения и дифференциации продукта TaskCraft.

Определить конкурентов:

Прямые конкуренты: Todoist, Microsoft To Do, Trello, Notion, Any.do.

Определить критерии анализа конкурентов:

Простота интерфейса.

Поддержка оффлайн.

Конфиденциальность данных.

Приступить к анализу (заполнение таблиц):

Провести анализ конкурентов по выбранным критериям и заполнить соответствующие таблицы.

Сопоставить полученные данные:

Сравнить данные по каждому из критериев для всех конкурентов и сделать выводы.

Сделать выводы:

Определить, какие функции и характеристики необходимо улучшить или добавить в TaskCraft для конкурентоспособности на рынке.

|  |  |  |  |
| --- | --- | --- | --- |
| Приложение | Простота интерфейса | Поддержка оффлайн | Конфиденциальность данных |
| Todoist | Перегруженный интерфейс | Требует интернет для полной функциональности | Собирает данные о пользователях |
| Microsoft To Do | Интуитивный интерфейс | Требует интернет для синхронизации | Интеграция с сервисами Microsoft может собирать данные |
| Trello | Перегруженный интерфейс | Требует интернет для синхронизации | Собирает данные о пользователях |
| Notion | Сложный интерфейс | Требует интернет для полной функциональности | Собирает данные о пользователях |
| Any.do | Простой интерфейс | Требует интернет для полной функциональности | Собирает данные о пользователях |

Выводы:

Todoist:

* Преимущества: Простое управление задачами, интеграция с календарем, поддержка тегов и меток.
* Недостатки: Перегруженный интерфейс, необходимость подключения к интернету для полной функциональности, сбор данных о пользователях.
* Microsoft To Do:
* Преимущества: Интуитивный интерфейс, синхронизация с другими сервисами Microsoft, поддержка списка задач, напоминания.
* Недостатки: Требуется интернет для синхронизации, возможен сбор данных через интеграции с сервисами Microsoft.

Trello:

* Преимущества: Визуальная организация задач с помощью карточек, гибкость в настройке досок, интеграции с другими инструментами.
* Недостатки: Перегруженный интерфейс, необходимость подключения к интернету для синхронизации, сбор данных о пользователях.

Notion:

* Преимущества: Все-в-одном инструмент для заметок и задач, мощные функции для организации и сотрудничества, настраиваемый интерфейс.
* Недостатки: Сложный интерфейс, необходимость подключения к интернету для полной функциональности, сбор данных о пользователях.

Any.do:

* Преимущества: Простой интерфейс, управление задачами и списками дел, интеграция с календарями, поддержка напоминаний и уведомлений.
* Недостатки: Требуется интернет для полной функциональности, сбор данных о пользователях.

На основе проведенного анализа можно сделать выводы о том, какие аспекты необходимо улучшить в TaskCraft для того, чтобы он был конкурентоспособен на рынке. Это включает в себя упрощение интерфейса, обеспечение оффлайн-доступа и повышение уровня конфиденциальности данных пользователей.

1. Календарный план проекта

Название проекта: TaskCraft

**Руководитель проекта**: Ростислав Коновалов (Тимлид)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название | Ответственный | Длительность | Дата начала | Дата окончания |
| Определение проблемы | Турабхан (Аналитик) | 7 дней | 08.03.2024 | 14.03.2024 |
| Выявление целевой аудитории | Турабхан (Аналитик) | 7 дней | 15.03.2024 | 21.03.2024 |
| Конкретизация проблемы | Турабхан (Аналитик) | 7 дней | 22.03.2024 | 28.03.2024 |
| Подходы к решению проблемы | Ростислав(Тимлид) | 7 дней | 29.03.2024 | 04.04.2024 |
| Анализ аналогов | Турабхан (Аналитик) | 7 дней | |  | | --- | | 05.04.2024 |  |  | | --- | |  | | 11.04.2024 |
| Определение платформы и стека для продукта | Ростислав (Тимлид) | 7 дней | 12.04.2024 | 18.04.2024 |
| Формулирование требований к MVP продукта | Турабхан (Аналитик) | 7 дней | 19.04.2024 | 25.04.2024 |
| Определение платформы и стека для MVP | Ростислав (Тимлид) | 7 дней | 26.04.2024 | 02.05.2024 |
| Формулировка цели | Турабхан (Аналитик) | 7 дней | 03.05.2024 | 09.05.2024 |
| Формулирование требований к продукту | Турабхан (Аналитик) | 7 дней | 10.05.2024 | 16.05.2024 |
| Определение задач | Турабхан (Аналитик) | 7 дней | 17.05.2024 | 23.05.2024 |
| Архитектура системы (компоненты, модули системы) | Макс, Вася (Программисты) | 7 дней | 24.05.2024 | 30.05.2024 |
| Разработка сценариев использования системы | Ростислав (Тимлид) | 7 дней | 31.05.2024 | 06.06.2024 |
| |  | | --- | | Прототипы интерфейсов |  |  | | --- | |  | | Женя (Дизайнер) | 7 дней | 07.06.2024 | 13.06.2024 |
| Дизайн-макеты | Женя (Дизайнер) | 0дней | 14.06.2024 | 14.06.2024 |
| Доработка приложения | Макс, Вася (Программисты) | 7 деней | 14.06.2024 | 20.06.2024 |

1. Сценарии использования

Акторы

Пользователь: Человек, использующий приложение для планирования и управления задачами.

Основные сценарии использования

Сценарий 1: Создание и управление задачами

Цель: Пользователь хочет создать новую задачу, установить приоритет и напоминание, а также управлять существующими задачами.

Начало: Пользователь открывает приложение TaskCraft.

Шаг 1: Пользователь выбирает опцию "Создать задачу".

Шаг 2: Пользователь вводит название задачи, описание и устанавливает приоритет.

Шаг 3: Пользователь устанавливает дату и время напоминания.

Шаг 4: Пользователь сохраняет задачу.

Шаг 5: Пользователь просматривает список задач и выбирает задачу для редактирования.

Шаг 6: Пользователь вносит изменения в задачу и сохраняет их.

Шаг 7: Пользователь отмечает задачу как выполненную.

Конец: Пользователь завершает сеанс работы с приложением.

Системные ответы:

Приложение сохраняет новую задачу и отображает ее в списке задач.

Приложение отправляет уведомление в установленное время напоминания.

Приложение сохраняет изменения в задаче и обновляет список задач.

Приложение отмечает задачу как выполненную и перемещает ее в архив.

Сценарий 2: Настройки приложения

Цель: Пользователь хочет изменить настройки приложения, такие как тема интерфейса или уведомления.

Начало: Пользователь открывает приложение TaskCraft.

Шаг 1: Пользователь выбирает опцию "Настройки".

Шаг 2: Пользователь выбирает раздел "Тема" и меняет тему интерфейса (например, на темную).

Шаг 3: Пользователь выбирает раздел "Уведомления" и настраивает параметры уведомлений.

Шаг 4: Пользователь сохраняет изменения.

Конец: Пользователь завершает сеанс работы с приложением.

1. Требования к продукту и к MVP

Требования к продукту TaskCraft систематизируются несколькими способами, включая требования клиентов, функциональные, нефункциональные и производные требования.

Требования клиентов

Клиенты TaskCraft – это пользователи, которые выполняют основные функции системного проектирования и взаимодействуют с системой. Главная цель системы – помочь пользователям эффективно управлять своими задачами и временем.

Функциональные требования

Функциональные требования объясняют, что должно быть сделано. Они идентифицируют задачи или действия, которые должны быть выполнены.

Создание задач:

Пользователь должен иметь возможность создать новую задачу, указав название, описание, приоритет и дату/время напоминания.

Редактирование задач:

Пользователь должен иметь возможность редактировать существующие задачи, внося изменения в их параметры.

Удаление задач:

Пользователь должен иметь возможность удалять задачи из списка.

Отметка задач как выполненных:

Пользователь должен иметь возможность отмечать задачи как выполненные.

Уведомления и напоминания:

Система должна отправлять уведомления и напоминания в указанное пользователем время.

Настройка приложения:

Пользователь должен иметь возможность изменять настройки приложения, включая тему интерфейса и параметры уведомлений.

Поддержка оффлайн-режима:

Система должна поддерживать работу в оффлайн-режиме, сохраняя изменения локально и синхронизируя их при подключении к интернету.

Нефункциональные требования

Нефункциональные требования определяют свойства, которые система должна демонстрировать, или ограничения, которые она должна соблюдать, не относящиеся к поведению системы.

Производительность:

Приложение должно обеспечивать быстрый отклик на действия пользователя, время выполнения любой операции не должно превышать 1 секунды.

Удобство сопровождения:

Код должен быть хорошо документирован и структурирован для облегчения сопровождения и обновления.

Расширяемость:

Архитектура системы должна позволять легко добавлять новые функции и интеграции.

Надежность:

Система должна быть устойчивой к сбоям и обеспечивать сохранность данных при неожиданных отключениях.

Конфиденциальность данных:

Данные пользователей должны быть защищены от несанкционированного доступа, минимизация сбора данных и прозрачная политика конфиденциальности.

Производные требования

Производные требования подразумеваются или преобразуются из высокоуровневых требований.

Хранение данных:

Данные о задачах должны храниться в локальной базе данных на устройстве пользователя.

Формат данных:

Данные о задачах должны храниться в формате, обеспечивающем их целостность и возможность восстановления после сбоя.

Обновления и синхронизация:

При наличии подключения к интернету система должна синхронизировать данные с основным хранилищем (если доступно), обеспечивая актуальность информации на всех устройствах пользователя.

Пример декомпозиции требований

Бизнес-требование:

Пользователь должен иметь возможность эффективно управлять своими задачами и временем.

Пользовательские требования:

Пользователь должен иметь возможность создать, редактировать, удалять задачи и отмечать их как выполненные.

Пользователь должен получать уведомления и напоминания о задачах.

Функциональные требования:

Система должна поддерживать создание, редактирование, удаление и отметку задач как выполненных.

Система должна отправлять уведомления и напоминания.

Нефункциональные требования:

Система должна обеспечивать быстрый отклик на действия пользователя.

Данные о задачах должны храниться в защищенной базе данных на устройстве пользователя.

Система должна поддерживать оффлайн-режим и синхронизацию данных при подключении к интернету.

Этот структурированный подход к определению требований позволяет создать продукт, который максимально удовлетворяет потребности пользователей и обеспечивает высокую эффективность их работы.

1. Стек для разработки

Все чаще можно наблюдать ситуацию, когда именно скорость запуска программного проекта является решающим фактором успеха. Долго и напряженно делать что-то ненужное хуже, чем быстро выпустить решение, полезное для клиентов и собрать обратную связь для следующего рывка.

Быстрый запуск возможен только при использовании максимума готовых составных элементов, фреймворков и библиотек с адекватным сроком поддержки (с учетом ожидаемого времени жизни вашего продукта).

К сожалению, алгоритмические и архитектурные знания востребованы все реже. Чаще приветствуется опыт решения похожих задач и практика использования комплексов инструментов и сильных сторон облачных провайдеров.

Не нужно ограничивать себя и все делать на одной технологии и только одном, ЛУЧШЕМ, языке программирования – огнестрел в руках ребенка эффективнее изучения боевых искусств самураем всю сознательную и бессознательную жизнь. Перфекционизм и идеализация убивают программные системы. Выбирайте верное оружие для каждой задачи в проекте. Берите готовое и сосредоточьте оставшиеся усилия на нестандартном.

Выбранные технологии должны быть простыми, понятными и доступными большинству в команде. Посмотрите, на чем пишут подобный проект в основном и возьмите этот стек технологий себе. Не автоматизируйте хостинг хаскелем и не пишите веб-сайты на C++. Если ваш проект «взлетит» и начнет развиваться, только тогда можно рассмотреть возможность переписывания его небольших частей на более сложных технологиях и языках программирования. Но обычно до этой стадии программные проекты либо не доходят, либо доходят через несколько лет.

Фреймворк позволяет на порядки ускорить запуск программного проекта. Обязательно проверьте наличие хорошей документации и уточните срок поддержки фреймворка, чтобы не случилась беда с полным переписыванием через 2-3 года. Это очень частый кейс.

Технология и язык(и) программирования не являются определяющими факторами успеха. Сосредоточьтесь на выстраивании самых необходимых процессов и внедрении ключевых практик разработки и проектирования. Правильные процессы, как правило, гарантированно приводят к «правильному» результату. Связавшись с «сектантами чистой технологии» или «попав на науку», вы потратите кучу времени впустую и можете так ничего и не запустить. Помните, 2-3 месяца это максимум для запуска в бой очередной итерации программного проекта.

Стек для разработки TaskCraft:

Язык программирования: Dart

Этот выбор позволяет нам использовать фреймворк Flutter для разработки кроссплатформенных приложений с высокопроизводительным и отзывчивым интерфейсом. Flutter широко применяется для разработки мобильных приложений и имеет отличную поддержку и документацию, что ускоряет процесс разработки и обеспечивает высокое качество конечного продукта.

При разработке TaskCraft мы сосредотачиваемся на простоте, удобстве и доступности выбранных технологий для нашей команды, что позволяет нам быстро адаптироваться и реагировать на изменения требований и обратную связь пользователей.

1. Прототипирование

Процесс создания прототипа TaskCraft включает в себя несколько ключевых шагов, направленных на разработку и тестирование начальных версий продукта с целью получения обратной связи и внесения улучшений.

Шаги создания прототипа

Определение начальных требований:

На этом этапе собираются основные функциональные и нефункциональные требования к TaskCraft. Это включает пожелания заказчиков, анализ потребностей целевой аудитории, а также внутренние технические и бизнес-требования.

Разработка первого варианта прототипа:

Создается начальный прототип интерфейса TaskCraft. Этот прототип фокусируется на пользовательском интерфейсе системы и включает основные экраны, такие как создание задач, управление задачами, настройка приложения и уведомления.

Этап изучения прототипа заказчиком и конечным пользователем:

Прототип предоставляется заказчику и конечным пользователям для тестирования. На этом этапе важно собрать мнения и отзывы о удобстве использования, функциональности и визуальной привлекательности прототипа.

Переработка прототипа с учетом полученных замечаний и предложений:

На основе обратной связи вносятся изменения и дополнения в прототип. Улучшаются элементы интерфейса и функциональность приложения. Этот процесс может повторяться несколько раз до достижения удовлетворительного результата.

Качества эффективного прототипа

Одноразовость:

Прототипы предназначены для донесения идеи до заинтересованных лиц. После получения необходимой информации и обратной связи прототип может быть отвергнут или переработан.

Сфокусированность:

Прототипы должны быть сфокусированы на ключевых частях интерфейса и взаимодействия, которые приносят наибольшую пользу продукту и пользователям.

Экономия времени и ресурсов:

Прототипы позволяют сэкономить время и ресурсы, выявляя ошибки и проблемы на ранних этапах разработки.

Простота и дешевизна разработки:

Прототипы должны быть простыми и дешевыми в разработке, что позволяет быстро проверять идеи и улучшать пользовательский интерфейс.

Возможность обновления:

Прототипы должны легко обновляться для внесения изменений и улучшений на основе полученной обратной связи.

Процесс разработки прототипа для TaskCraft

Определение начальных требований:

Собираются и анализируются требования пользователей и заказчиков. Выявляются основные функции и взаимодействия, которые должны быть включены в первый прототип. Например:

Возможность создания задач с установкой приоритетов и напоминаний.

Возможность редактирования и удаления задач.

Поддержка оффлайн-режима.

Настройки интерфейса и уведомлений.

Разработка первого варианта прототипа:

Создается начальный прототип интерфейса TaskCraft с использованием фреймворка Flutter. Прототип включает:

Главный экран с отображением списка задач.

Экран создания задачи с полями для ввода названия, описания, приоритета и времени напоминания.

Экран настройки приложения с возможностью изменения темы интерфейса и параметров уведомлений.

Этап изучения прототипа заказчиком и конечным пользователем:

Прототип предоставляется заказчику и пользователям для тестирования. Собираются мнения и отзывы о:

Удобстве использования интерфейса.

Визуальной привлекательности дизайна.

Функциональности и логике работы приложения.

Переработка прототипа:

На основе полученной обратной связи вносятся изменения и улучшения в прототип. Это может включать:

Изменение расположения элементов интерфейса для улучшения юзабилити.

Добавление новых функций, предложенных пользователями.

Оптимизация производительности и устранение выявленных ошибок.

Примеры прототипов TaskCraft

Первый прототип:

Включает основные экраны и элементы управления.

Демонстрирует общую структуру и функциональность приложения.

Прототип прост в разработке и быстро модифицируется для тестирования различных идей и подходов.

Детализированный прототип:

После успешного тестирования первого прототипа создается более детализированный вариант.

Включает все функциональные элементы и улучшенный интерфейс.

Прототип приближен к окончательной версии продукта и используется для финального тестирования перед разработкой MVP.

Создание эффективного прототипа интерфейса является чрезвычайно важной задачей. Прототип должен хорошо выглядеть, быть максимально дешевым, максимально полным и легко обновляемым. Начальные прототипы должны быть простыми и быстро модифицируемыми, чтобы проверить несколько версий интерфейса и исправить ошибки. Поздние версии прототипов должны быть максимально похожими на конечную систему для точного тестирования и получения достоверной обратной связи.

1. Проектирование и разработка системы

На этапе проектирования и разработки системы TaskCraft формируется общая структура программного комплекса. Программный комплекс разбивается на небольшие части – программные модули (блоки). Для каждого программного модуля формулируются требования по реализуемым функциям и разрабатывается алгоритм, реализующий эти функции. В этом разделе приведем описание этих модулей, алгоритм работы приложения и схему взаимодействия программных модулей.

Описание программных модулей

Модуль создания и управления задачами

Функции: Создание, редактирование, удаление задач, установка приоритетов и напоминаний.

Алгоритм: Пользователь создает/редактирует задачу -> система сохраняет изменения в локальной базе данных -> при необходимости устанавливаются напоминания.

Модуль уведомлений

Функции: Уведомление пользователя о предстоящих задачах.

Алгоритм: Система проверяет дату и время задач -> при совпадении с текущим временем отправляется уведомление.

Модуль настроек

Функции: Изменение параметров интерфейса и уведомлений.

Алгоритм: Пользователь изменяет настройки -> система сохраняет изменения в локальной базе данных.

Модуль оффлайн-режима

Функции: Работа с задачами без подключения к интернету, синхронизация данных при подключении.

Алгоритм: Пользователь вносит изменения в оффлайн-режиме -> система сохраняет данные локально -> при подключении к интернету данные синхронизируются.

Алгоритм работы приложения

Запуск приложения:

Проверка наличия активной сессии пользователя.

Если сессия активна -> переход к главному экрану задач.

Если сессии нет -> переход к экрану аутентификации.

Создание задачи:

Пользователь открывает экран создания задачи.

Вводит название, описание, приоритет и время напоминания.

Сохраняет задачу -> система сохраняет данные в локальной базе данных.

Редактирование задачи:

Пользователь выбирает задачу для редактирования.

Вносит изменения и сохраняет.

Система обновляет данные в локальной базе данных.

Удаление задачи:

Пользователь выбирает задачу для удаления.

Подтверждает удаление.

Система удаляет задачу из локальной базы данных.

Уведомления:

Система проверяет задачи на наличие предстоящих напоминаний.

При наступлении времени напоминания -> отправляется уведомление пользователю.

Настройки:

Пользователь изменяет параметры интерфейса и уведомлений.

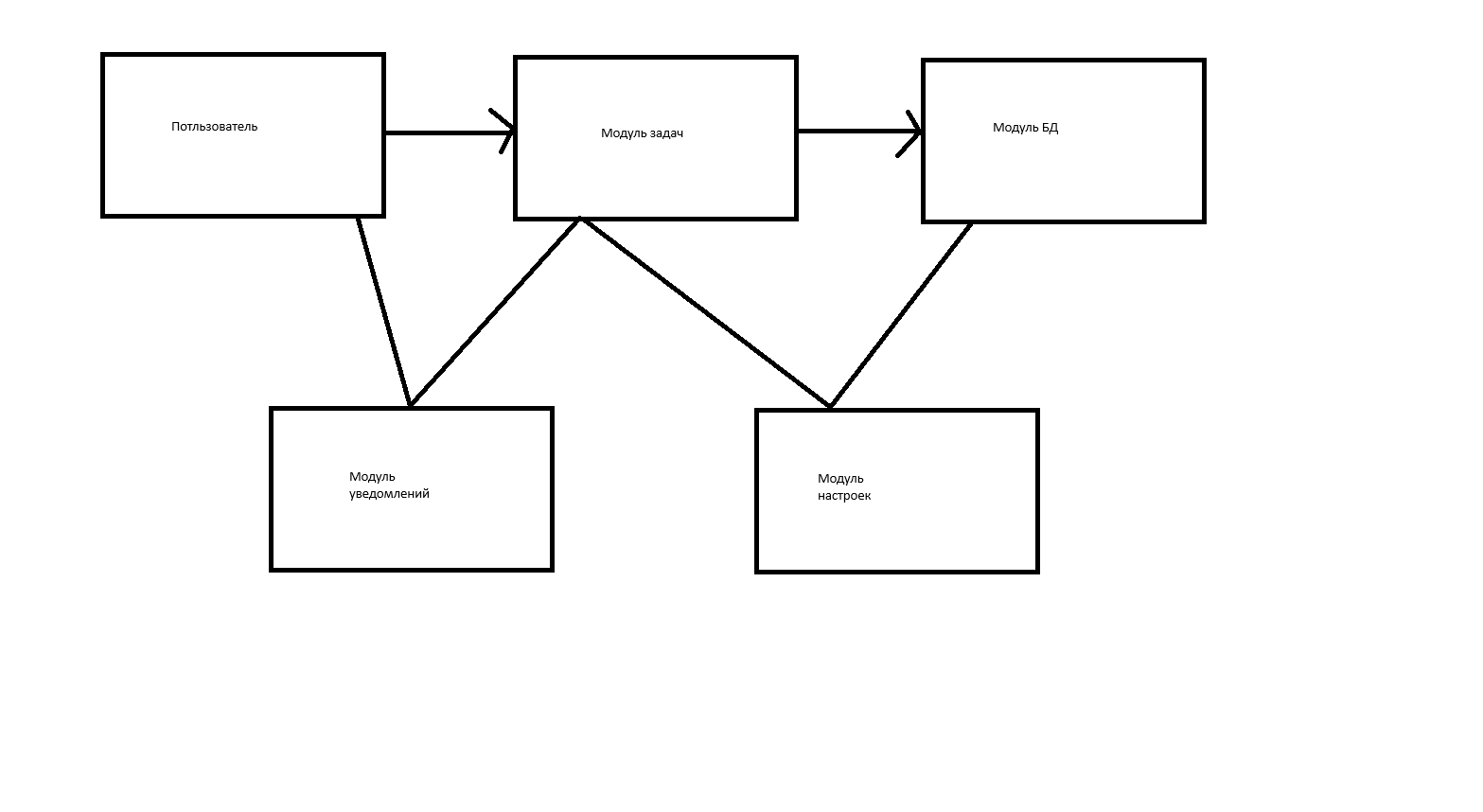
Система сохраняет изменения в локальной базе данных.

Оффлайн-режим:

Пользователь работает с задачами в оффлайн-режиме.

Система сохраняет данные локально.

При подключении к интернету данные синхронизируются с основным хранилищем.

Схема взаимодействия программных модулей:  


ЗАКЛЮЧЕНИЕ

В ходе разработки проекта TaskCraft была проведена значительная работа по определению проблем, потребностей целевой аудитории, анализу конкурентов и созданию эффективного прототипа. На основе собранных данных и проведенного анализа был разработан план проектирования и реализации системы, которая обеспечивает удобное и эффективное управление задачами для широкого круга пользователей.