

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	5
1. ОБЩИЙ РАЗДЕЛ	7
1.1. Исследование предметной области	7
1.2. Аналитический обзор существующих решений	7
1.3. Обоснование выбора стека технологий	7
1.3.1. Выбор языка программирования	7
1.3.2. Выбор системы управления базами данных	7
1.3.3. Выбор дополнительных средств разработки	7
2. СПЕЦИАЛЬНЫЙ РАЗДЕЛ	7
2.1. Требования к программному обеспечению	7
2.2. Проектирование и разработка структуры базы данных	7
2.3. Проектирование и разработка пользовательского интерфейса	7
2.4. Разработка алгоритмов	7
2.5. Разработка бизнес-логики	7
3. РЕАЛИЗАЦИЯ	7
3.1. Настройка технологического обеспечения	7
3.2. Инструкция использования	7
4. ТЕСТИРОВАНИЕ	7
4.1. Проведение тестирования и отладки	7
ЗАКЛЮЧЕНИЕ	7
СПИСОК ИСТОЧНИКОВ И ЛИТЕРАТУРЫ	7
ПРИЛОЖЕНИЕ 1. ИСХОДНЫЙ КОД	7

ВВЕДЕНИЕ

В современном динамичном мире, где информационные потоки неустанно нарастают, а требования к эффективности и организации личного и рабочего времени становятся все более строгими, неотъемлемым инструментом становится систематизация задач и планирование деятельности. Одним из ответов на вызовы современной жизни является разработка приложения для планирования и управления рабочими и личными задачами, что представляет собой актуальную и перспективную область информационных технологий.

Проблема, с которой сталкиваются люди, связана с неэффективным управлением временем и задачами в современном мире. Требования к организации повседневной деятельности становятся все более строгими, а поток информации неустанно растет. В таких условиях разработка инновационного приложения становится насущной необходимостью, направленной на облегчение этой задачи для пользователей.

Актуальность темы курсового проекта обусловлена наличием современных задач, которые требуют эффективных средств управления. Существующие приложения для планирования задач, несмотря на свою распространенность, не всегда полностью отвечают потребностям пользователей, оставляя пробелы в функциональности и удобстве использования.

Смотря на готовые решения, можно заметить, что они часто ограничены в функциональности, не предоставляют гибкости в настройке и не всегда соответствуют современным стандартам эффективного управления временем. В связи с этим возникает необходимость в разработке собственного приложения, которое бы удовлетворяло все требования пользователей.

Целью данного курсового проекта является разработка приложения для планирования и управления рабочими и личными задачами, способного поддерживать пользователя в организации своей повседневной деятельности. Проект ориентирован на обеспечение простоты использования, гибкости

настроек и максимальной функциональности для учёта разнообразных потребностей пользователей.

Были определены следующие задачи, которые необходимо решить в ходе курсового проектирования: исследовать предметную область; провести аналитический обзор существующих решений; определить стек используемых при разработке технологий; спроектировать и разработать структуру базы данных; спроектировать и разработать структуру приложения и алгоритмы работы его компонентов; реализовать приложение; провести тестирование и отладку.

В итоге курсового проектирования необходимо создать приложение, практическая значимость которого заключается в организации рабочего и личного времени пользователей, предоставляя простоту использования, гибкость настроек и максимальную функциональность. Его практическая значимость заключается не только в устранении трудностей, связанных с управлением задачами, но и в том, чтобы поддерживать пользователей на пути к повышению эффективности и достижению поставленных целей. Внимание к современным тенденциям в управлении временем и учет потребностей пользователей является ключевым элементом, обеспечивающим создание приложения, способного успешно интегрироваться в их повседневную жизнь.

1. ОБЩИЙ РАЗДЕЛ

1.1. Исследование предметной области

Без тщательного анализа задачи невозможно эффективно разработать приложение, отвечающее потребностям пользователей.

В современном обществе, где информационные потоки без устали нарастают, и требования к эффективности и организации времени становятся все более строгими, важность глубокого понимания задач и проблем, с которыми сталкиваются конечные пользователи, не может быть недооценена. Этот анализ представляет собой ключевой этап, в ходе которого выявляются уникальные потребности разнообразных категорий пользователей. Различные контексты использования, особенности повседневной деятельности и индивидуальные предпочтения требуют внимательного рассмотрения, чтобы создать инструмент, который действительно будет соответствовать реальным потребностям пользователей.

Целевая аудитория разрабатываемого приложения тесно связана с пользователями, стремящимися к повышению своей продуктивности и эффективности управления своим личным и рабочим временем. В современном мире, насыщенном информацией и постоянно меняющимися рабочими условиями, возникает острая необходимость в инструментах, способных помочь пользователям эффективнее распределять и контролировать свои задачи.

Целевая аудитория включает в себя широкий спектр пользователей, начиная от школьников, студентов и предпринимателей до заботливых домохозяек.

После тщательного анализа поставленной задачи, было принято решение представить основные компоненты системы в форме UML-диаграммы.

UML (англ. Unified Modeling Language — унифицированный язык моделирования) — язык графического описания для объектного моделирования в области разработки программного обеспечения, для моделирования бизнес-процессов, системного проектирования и отображения организационных структур [X].

На основе изучения предметной области были выделены роли, которые взаимодействуют с системой, каждая из них выполняет определенный набор функций. Результаты этого анализа отражены в виде диаграммы прецедентов, представленной на рисунке 1.1.

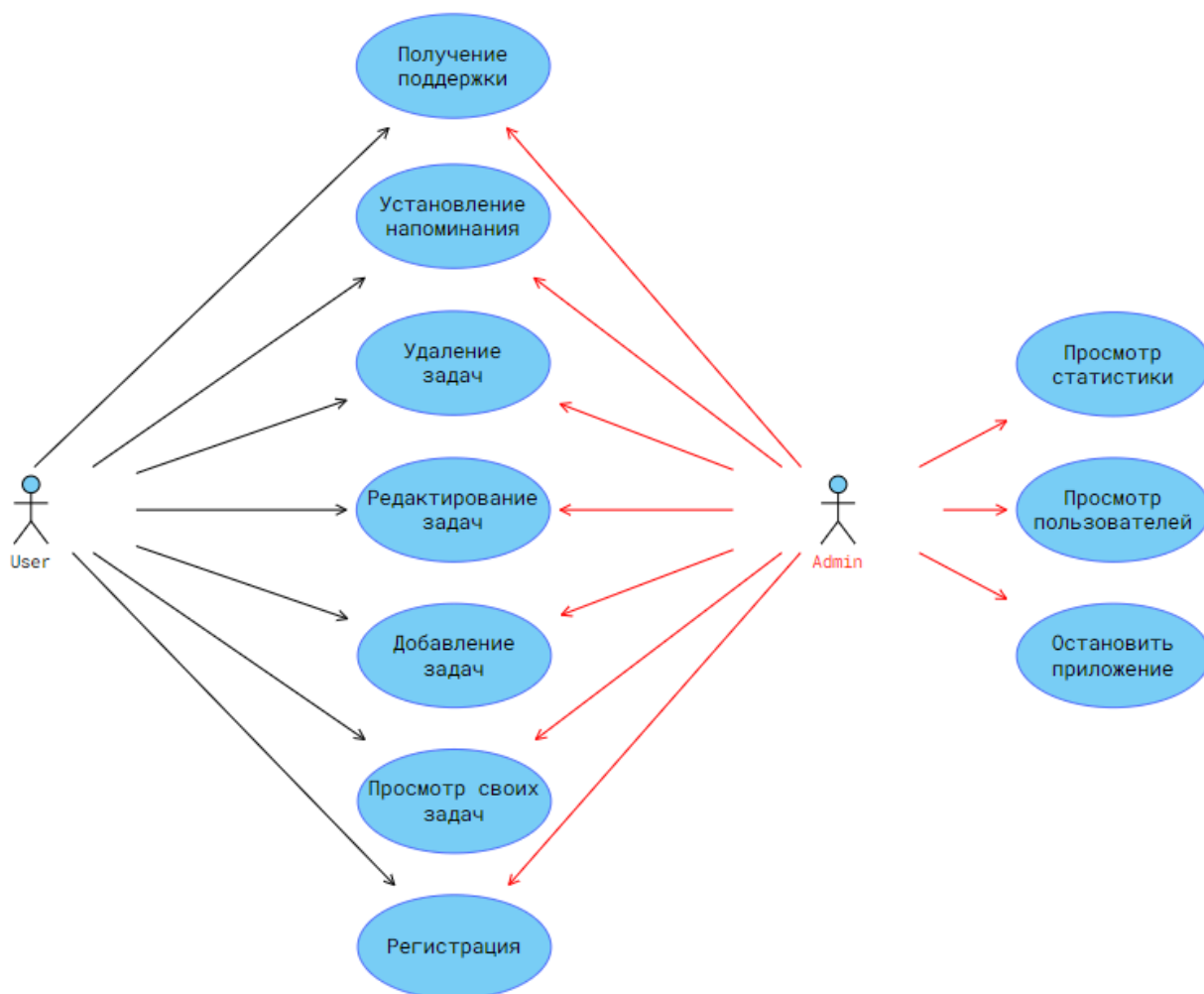


Рисунок 1.1 - UML-диаграмма вариантов использования

На диаграмме вариантов использования можно увидеть, что в системе будут присутствовать две роли: пользователь и администратор.

У пользователя доступны все функции разрабатываемого приложения, такие как: регистрация, просмотр своих задач, добавление задач, редактирование задач, удаление задач, установление напоминания, получение поддержки. У

администратора есть всё выше перечисленное и в дополнение - остановка приложения, просмотр пользователей приложения, просмотр статистики.

Для отображения последовательности действий в информационной системе, была разработана диаграмма последовательности, представленная на рисунке 1.2.

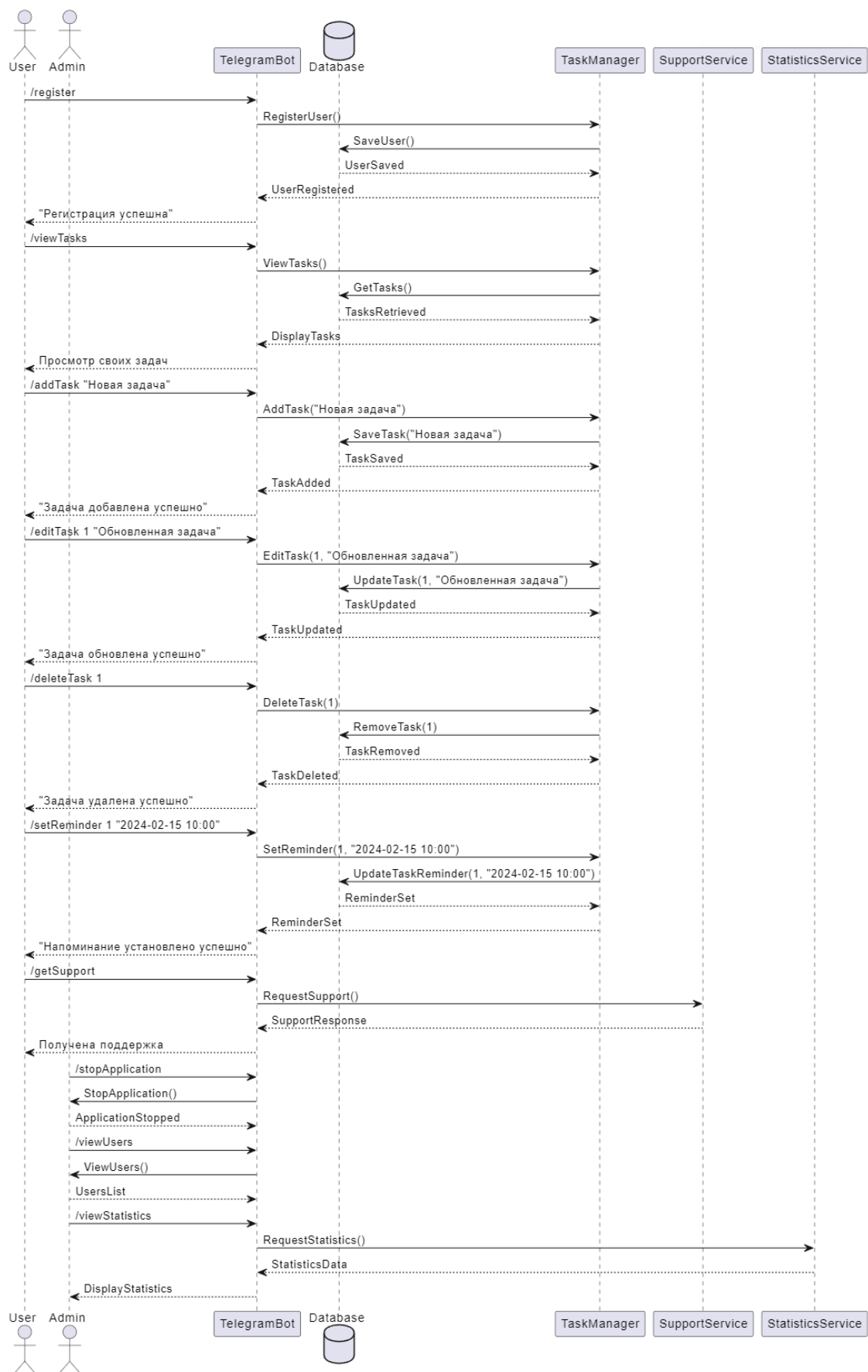


Рисунок 1.2 - UML-диаграмма последовательности

На данной диаграмме последовательности представлены действия, которые пользователь и администратор выполняют во взаимодействии с приложением управления задачами. Для использования приложения управления задачами пользователь должен следовать нескольким шагам. Сначала, он должен зарегистрироваться, отправив соответствующий запрос боту. После успешной регистрации пользователь может просматривать свои задачи, добавлять новые, редактировать, удалять, а также устанавливать напоминания для задач. В случае необходимости он может обратиться за поддержкой, отправив запрос боту. Все эти действия осуществляются с использованием механик Telegram API, что делает использование бота удобным и доступным для пользователя.

Представленные выше диаграммы отражают взаимодействие модулей системы в процессе выполнения основных функций. При регистрации пользователя система обращается к модулю управления задачами для сохранения данных в базе, после чего возвращает подтверждение. Запрос на просмотр задач активирует обращение к модулю для получения списка из базы, который затем предоставляется пользователю. Аналогично, операции добавления, редактирования, удаления задач и другие взаимодействия с базой данных и компонентами системы выполняются для обеспечения функциональности приложения.

Подводя итоги данного раздела, можно утверждать, что были выявлены ключевые аспекты, необходимые для разработки приложения, соответствующего потребностям пользователей. Внимание к уникальным запросам различных пользовательских категорий, особенностям контекстов использования и индивидуальным предпочтениям является неотъемлемой частью процесса. Результатом этого анализа стала UML-диаграмма прецедентов, отражающая взаимодействие пользователей и администратора с приложением, а также диаграмма последовательности для визуализации последовательности действий в системе. Полученная информация формирует основу для разработки приложения, способного эффективно удовлетворять

разнообразные потребности пользователей и обеспечивать эффективное управление задачами.

1.2. Аналитический обзор существующих решений

Сегодня можно найти много инструментов и решений для управления своим временем и задачами.

Проведем анализ двух существующих решений, предназначенных для планирования и управления задачами.

TickTick — сервис для создания и организации списков задач (Рисунок 1.3).

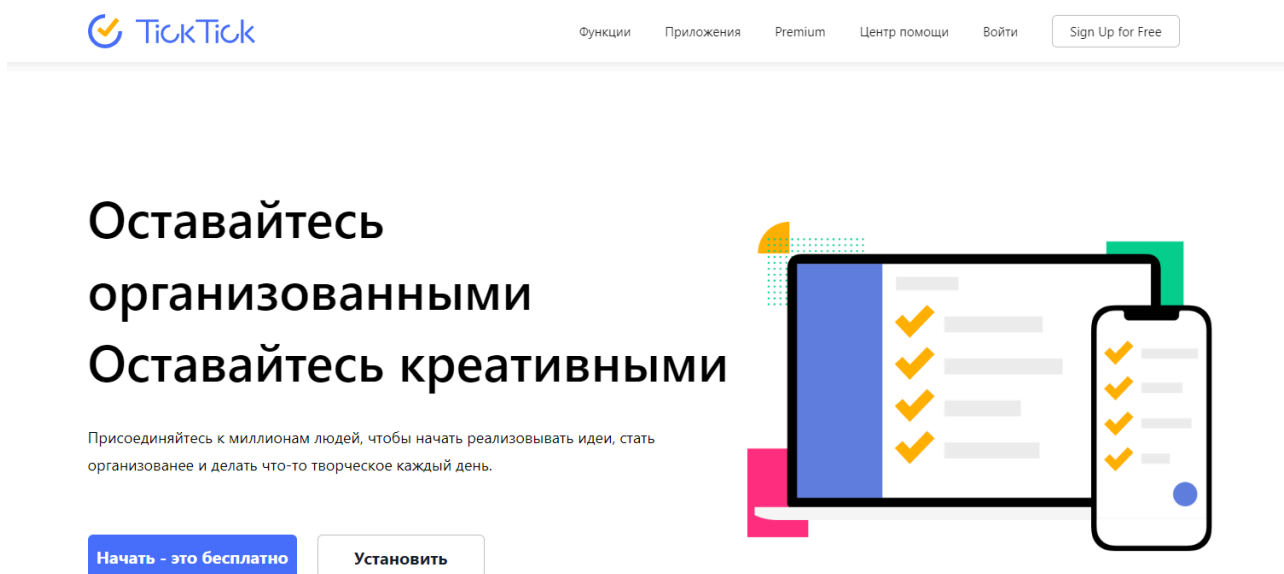


Рисунок 1.3 - Главный экран проекта TickTick

TickTick предоставляет обширный набор функций для управления задачами, включая создание подзадач, установку напоминаний и интеграцию с календарем, что делает его полезным инструментом для эффективного планирования времени. Многоплатформенность приложения обеспечивает синхронизацию данных между устройствами, а интуитивный интерфейс и гибкие настройки облегчают использование. Однако ограниченные бесплатные функции и необходимость в платной подписке для доступа к продвинутым возможностям могут ограничить опыт для некоторых пользователей. Возможны также проблемы с синхронизацией.

Следующая платформа для рассмотрения - Trello. Это платформа для управления проектами и задачами, основанная на концепции карточек и досок. (Рисунок 1.5).

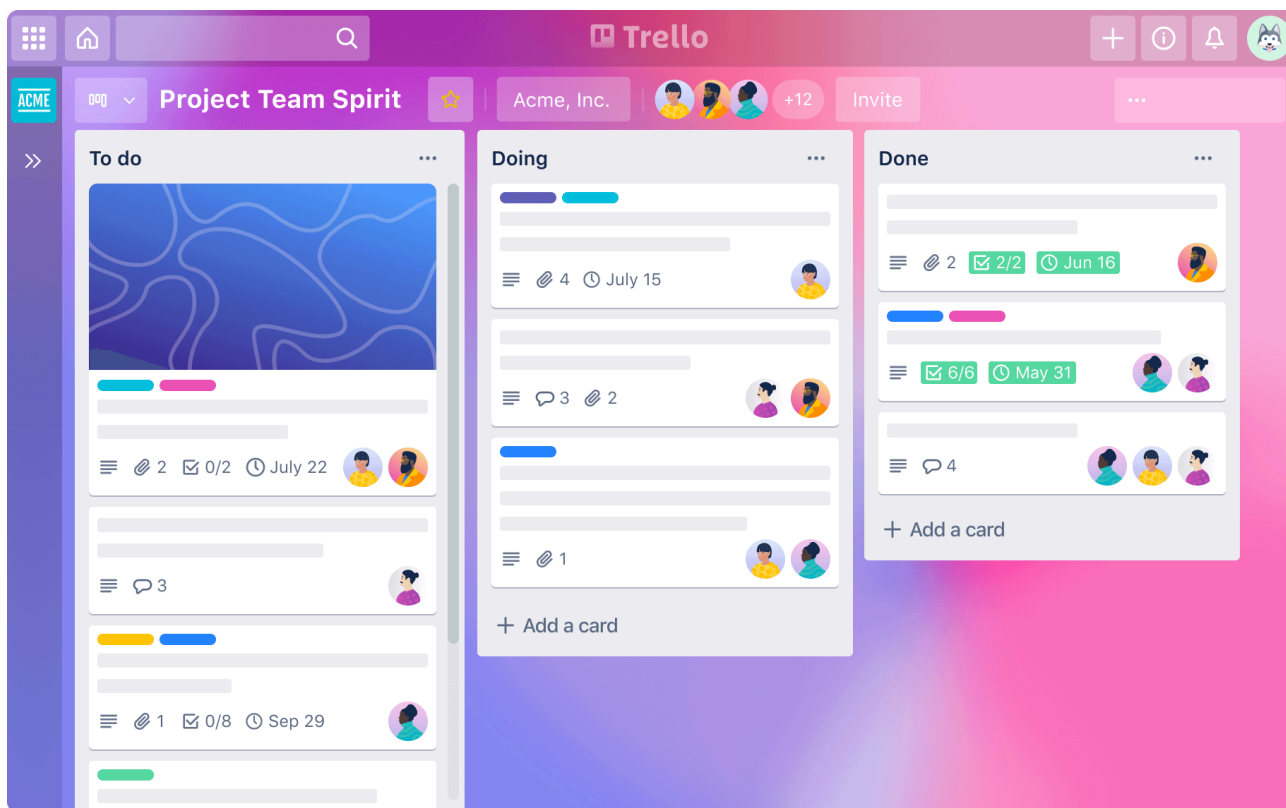


Рисунок 1.4 - Пример проекта Trello

Его преимущества включают интуитивный интерфейс, где пользователи могут создавать задачи в виде карточек, организовывать их на досках и следить за ходом выполнения проектов. Интеграция с различными приложениями и сервисами, а также наличие бесплатной версии, делают Trello привлекательным для широкого круга пользователей. Однако у Trello также есть свои недостатки. Ограниченные возможности в бесплатной версии могут ограничивать пользователя, а более расширенные функции доступны только в платных планах. Кроме того, в сравнении с TickTick, Trello предоставляет меньше инструментов для управления временем и создания задач. Как к минусам Trello для пользователей в России можно добавить, что в некоторых случаях могут возникнуть проблемы с доступом, так как сервис может быть заблокирован. Это может создать неудобства для пользователей, пытающихся использовать Trello в России.

Чтобы выявить достоинства и недостатки выше рассмотренных инвестиционных решений по сравнению друг с другом, был проведен сравнительный анализ этих продуктов, результаты которого представлены в таблице 1.1.

Таблица 1.1 – Сравнительный анализ существующих решений

Функциональные особенности	TickTick	Trello
Создание задач и подзадач	Да	Да
Установка напоминаний	Да	В ограниченной степени
Интеграция с календарем	Да	В ограниченной степени
Многоплатформенность	Да (веб, мобильные устройства, десктоп)	Да (веб, мобильные устройства, десктоп)
Интуитивный интерфейс	Да	Да
Гибкие настройки	Да	Ограничены в бесплатной версии
Бесплатная версия	Да (с ограничениями)	Да (с ограничениями)
Платные возможности	Да (расширенные функции)	Да (расширенные функции)
Проблемы с синхронизацией	Иногда возникают	В ограниченных случаях
Интеграция с другими приложениями	В ограниченной степени	В ограниченной степени
Доступность в России	Да	Могут возникнуть проблемы из-за блокировки

Оба рассмотренных решения, TickTick и Trello, предоставляют разнообразные инструменты для управления задачами, но выбор между ними зависит от конкретных потребностей пользователя. TickTick обеспечивает более

обширный функционал, включая управление временем, однако для полного доступа могут потребоваться платные подписки. Trello привлекателен своей интуитивной структурой карточек и досок, но его ограниченные возможности в бесплатной версии и региональные ограничения в России могут быть ограничивающими для целевой аудитории. Разработка собственного решения может быть рассмотрена для учета конкретных требований и избежания ограничений существующих платформ.

1.3. Обоснование выбора стека технологий

1.3.1. Выбор языка программирования

1.3.2. Выбор системы управления базами данных

1.3.3. Выбор дополнительных средств разработки

2. СПЕЦИАЛЬНЫЙ РАЗДЕЛ

2.1. Требования к программному обеспечению

2.2. Проектирование и разработка структуры базы данных

2.3. Проектирование и разработка пользовательского интерфейса

2.4. Разработка алгоритмов

2.5. Разработка бизнес-логики

3. РЕАЛИЗАЦИЯ

3.1. Настройка технологического обеспечения

3.2. Инструкция использования

4. ТЕСТИРОВАНИЕ

4.1. Проведение тестирования и отладки

ЗАКЛЮЧЕНИЕ

СПИСОК ИСТОЧНИКОВ И ЛИТЕРАТУРЫ

Используемая литература

Интернет-источники

ПРИЛОЖЕНИЕ 1. ИСХОДНЫЙ КОД