**1 Анализ и постановка задачи**

**1.1 Анализ предметной области**

В современном мире, где информационный поток постоянно увеличивается, важно иметь инструмент, который поможет организовать задачи и отслеживать выполнение. Приложения управления задачами представляют собой эффективный инструмент для управления задачами, который позволяет пользователям создавать, отслеживать и организовывать свои рабочие или личные дела.

Веб-приложения становятся неотъемлемой частью повседневной жизни, предоставляя пользователям доступ к различным услугам и информации. Разработка веб-приложений на сегодняшний день является одним из наиболее востребованных направлений в IT-индустрии. Данные технологии включают в себя создание программного обеспечения, способного работать в веб-браузере, с применением различных технологий, таких как HTML, CSS и JavaScript. Представленные языки программирования обеспечивают разработчикам необходимые инструменты для создания динамических и интерактивных веб-приложений.

Веб-приложение — это программное обеспечение, которое работает через интернет и доступно пользователям через веб-браузер. Веб-приложение сочетает в себе элементы клиентской и серверной частей, позволяя взаимодействовать с пользователями, обрабатывать данные и предоставлять различные услуги или функциональность. Веб-приложения могут варьироваться от простых форм и интерактивных страниц до сложных систем управления контентом и онлайн-сервисов.

Веб-разработка — это процесс создания, проектирования и поддержания веб-сайтов и веб-приложений. Данный процесс включает в себя как frontend-разработку, отвечающую за пользовательский интерфейс и взаимодействие с пользователем, так и backend-разработку, обеспечивающую серверную логику и управление данными.

Фронтенд и бэкенд разработка играют ключевую роль в создании современных веб-приложений, обеспечивая пользовательский интерфейс и логику работы приложения соответственно. Фронтенд отвечает за то, как приложение выглядит и взаимодействует с пользователем, в то время как бэкенд обеспечивает работу приложения на сервере, управление данными и бизнес-логику. Важно подчеркнуть, что фронтенд и бэкенд разработка тесно взаимосвязаны и требуют согласованной работы для создания полноценного веб-приложения.

С развитием технологий веб-разработки, веб-приложения становятся все

более мощными и интерактивными. Современные веб-технологии позволяют создавать сложные приложения с динамическим контентом, анимациями, мгновенной загрузкой данных и другими продвинутыми функциями, делая пользовательский опыт более удобным и привлекательным.

Современные веб-разработчики стремятся создавать веб-приложения не только функциональными, но и обладающими привлекательным и удобным пользовательским интерфейсом. Использование современных методик разработки, таких как Responsive Web Design, позволяет создавать приложения, которые корректно отображаются на различных устройствах, включая компьютеры, планшеты и смартфоны, обеспечивая при этом единое и удобное пользовательское взаимодействие.

Безопасность веб-приложений становится все более актуальной темой в условиях увеличения киберугроз и утечек данных. Защита от SQL-инъекций, кросс-сайтового скриптинга, обеспечение безопасности сессий и другие меры становятся необходимыми для обеспечения конфиденциальности и целостности информации.

Веб-приложения обладают рядом преимуществ перед традиционными настольными приложениями, таких как мобильность, доступность из любой точки мира, простота масштабирования и обновлений, а также возможность работы на различных операционных системах без необходимости установки дополнительных программ на устройство пользователя.

Веб-приложения играют ключевую роль в современном цифровом мире, обеспечивая пользователей удобными и доступными инструментами для выполнения различных задач. Способность работать через интернет и взаимодействовать с пользователями в реальном времени делает технологию особенно привлекательной как для бизнеса, так и для конечных пользователей. В конечном счете, веб-приложения не только улучшают производительность и эффективность, но и открывают новые горизонты для взаимодействия и сотрудничества. Значение веб-приложений в повседневной жизни и бизнес-процессах продолжает расти, что позволяет являться важной частью будущего цифровой экономики. [9]

**1.2 Анализ аналогов и прототипов**

В настоящее время существует большое разнообразие информационных ресурсов, выступающих в качестве аналогов разрабатываемому веб-приложению. Существует несколько причин, почему веб-приложения сейчас активно используются:

доступность – веб-приложения доступны из любого места, где есть интернет, что обеспечивает пользователю гибкость в работе и использовании сервисов;

мультиплатформенность – веб-приложения могут работать на различных

устройствах и операционных системах без необходимости установки дополнительных приложений, что делает приложения удобными для пользовате-

лей;

простота распространения – веб-приложения легко распространять через ссылки, что облегчает использование и распространение среди пользователей;

совместимость – хорошо разработанные веб-приложения обеспечивают высокую совместимость с различными браузерами, что делает приложения доступными для широкой аудитории.

Для эффективного планирования программного продукта необходимо провести анализ существующих аналогов, чтобы определить уже имеющиеся разработки и применение в реальной практике.

Первым аналогом является веб-приложение «Todoist». Данное приложение предоставляет возможность отслеживать свою деятельность, контролировать выполнение заданий и создавать заметки.

Интерфейс веб-приложения «Todoist» представлен на рисунке 1.1.

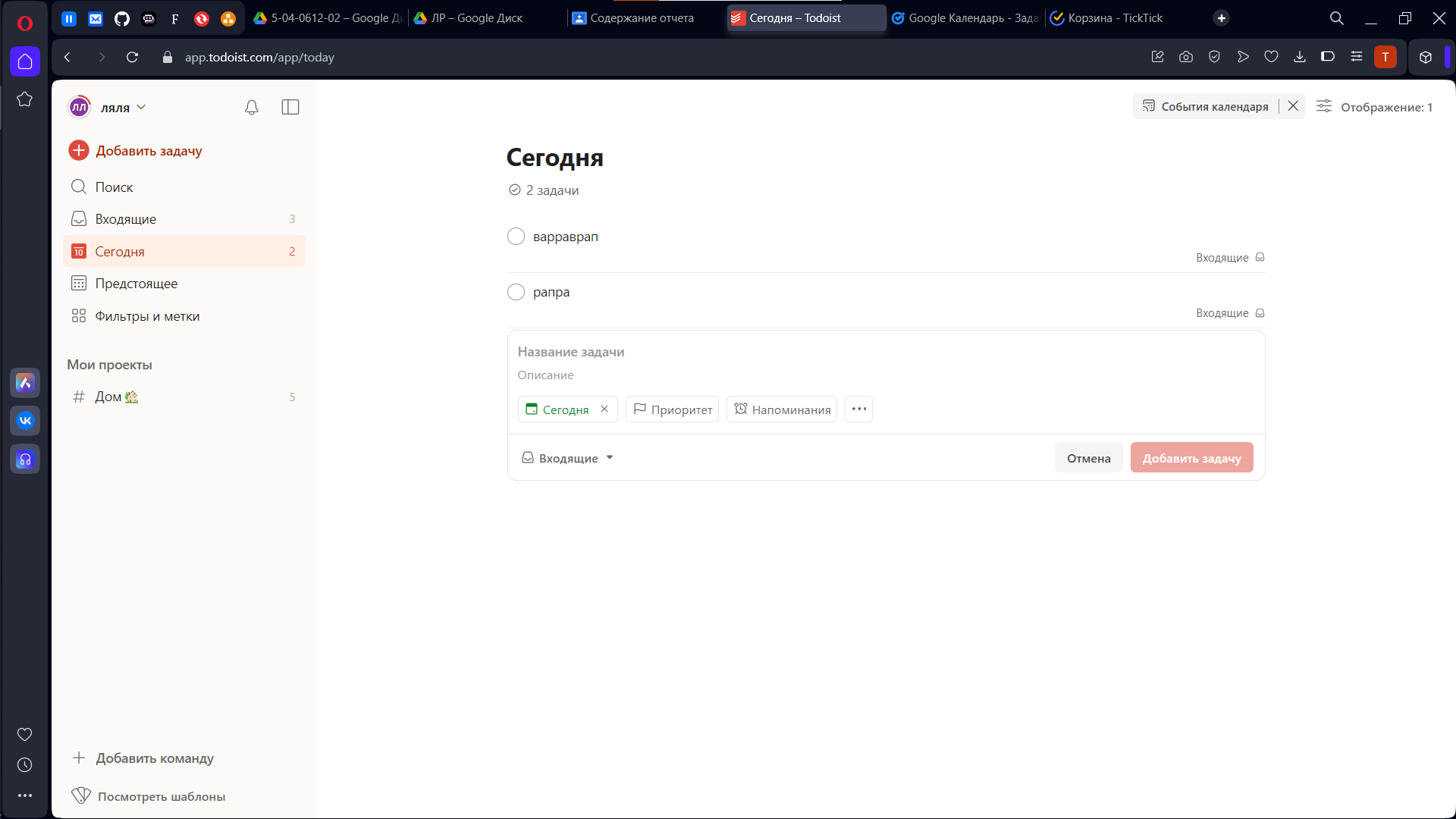


Рисунок 1.1 – Веб-приложение «Todoist»

Источник: интернет-ресурс

Достоинства:

мультиплатформенность;

оффлайн-режим;

уведомления и напоминания.

Недостатки:

ограниченные функции в бесплатной версии;

сложность интерфейса;

отсутствие просмотра выполненных задач.

Вторым аналогом является веб-приложение «Google Календарь». Приложение «Google Календарь» предоставляет возможность свои задачи в заметки, отслеживать выполнение, добавлять задания в календарь.

Интерфейс веб-приложения «Google Календарь» представлен на рисунке 1.2.

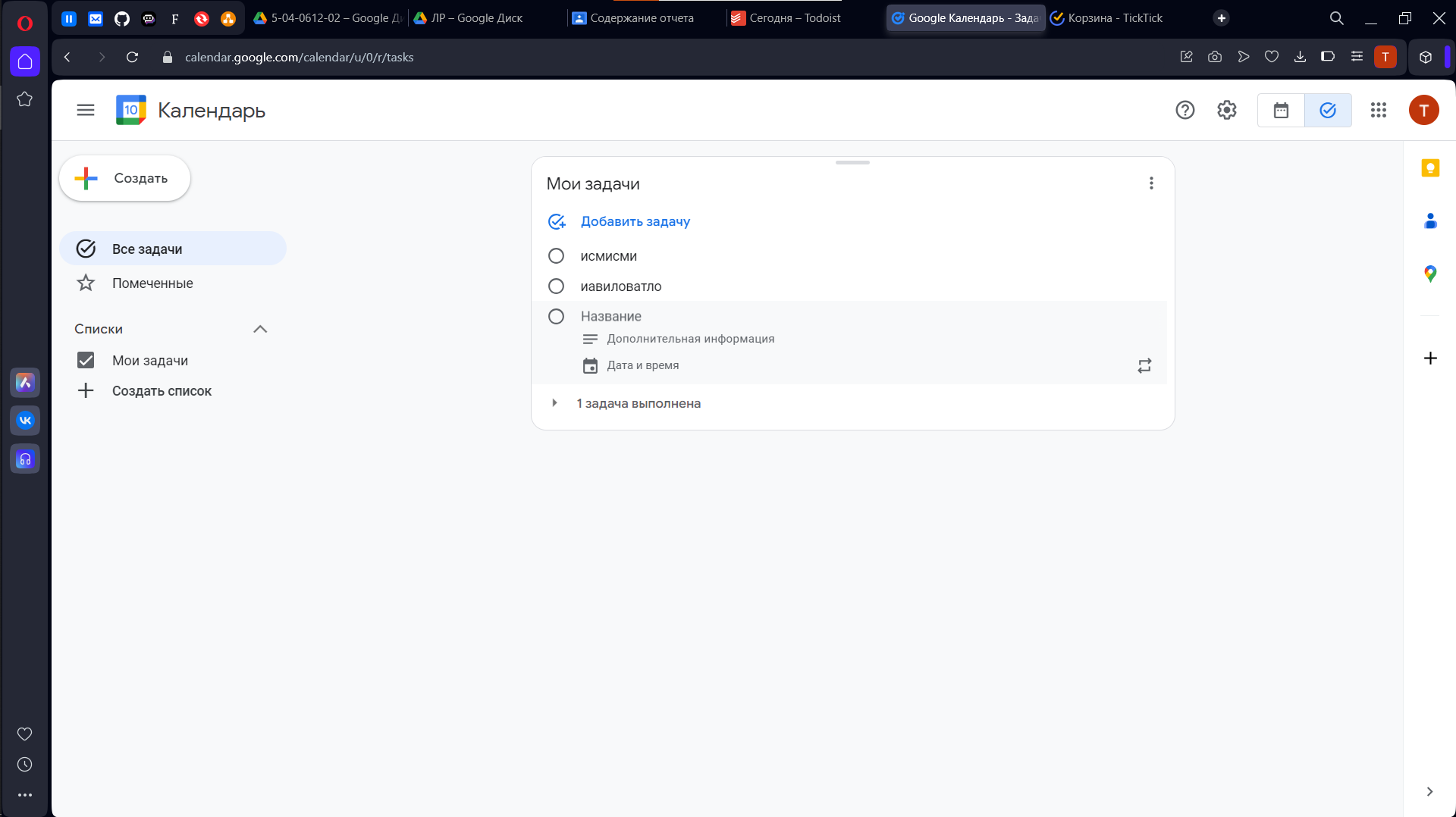


Рисунок 1.2 – Веб-приложение «Google Календарь»

Источник: интернет-ресурс

Достоинства:

мультиплатформенность;

совместное использование;

гибкие настройки.

Недостатки:

сложность интерфейса;

отсутствует возможность просмотреть выполненные задачи.

Третьим аналогом является веб-приложение «TickTick». Данное приложение предоставляет возможность управления задачами и планирования времени, а так же способствует увеличению продуктивности.

Интерфейс веб-приложения «TickTick» представлен на рисунке 1.3.

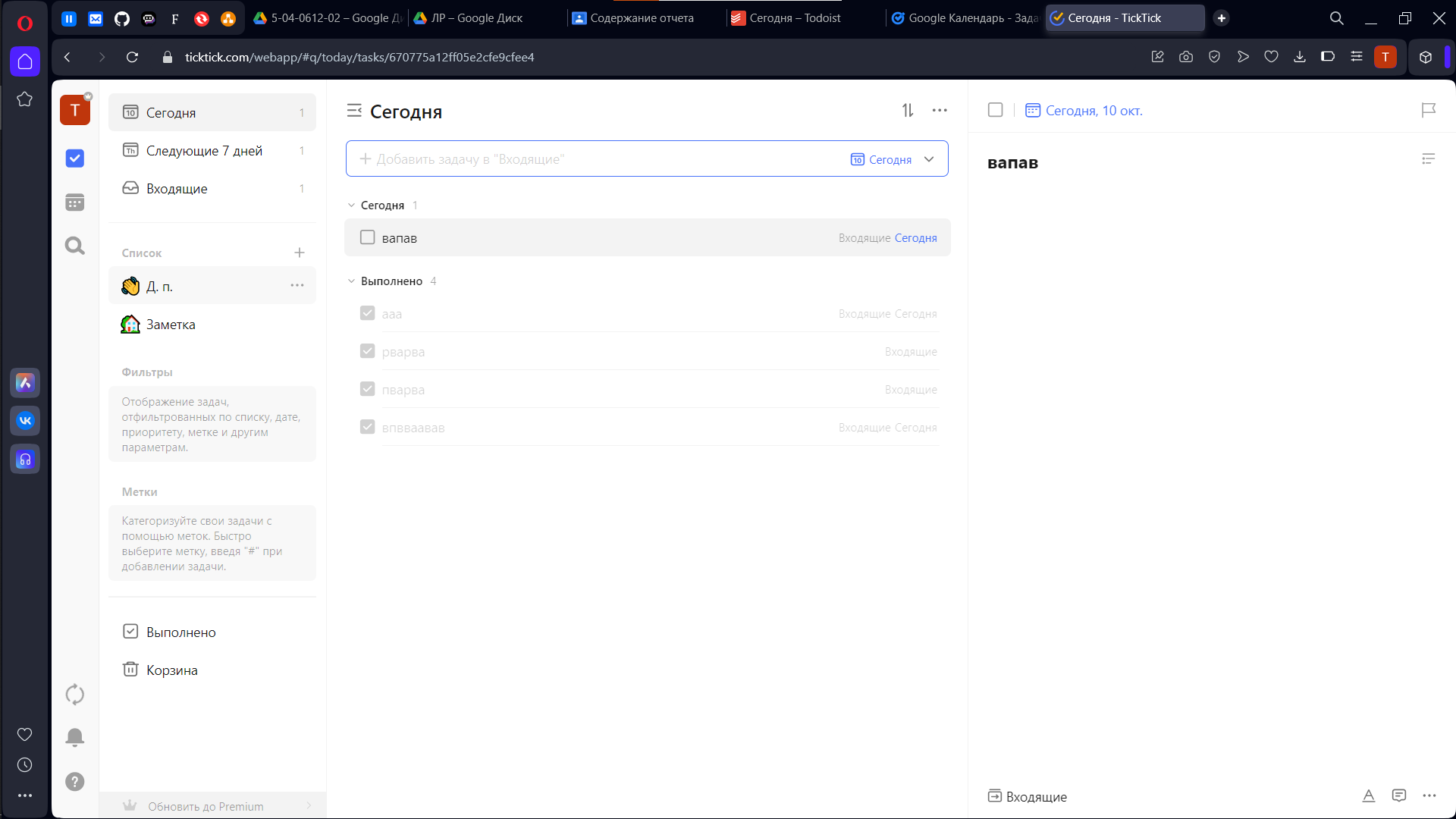


Рисунок 1.3 – Веб-приложение «TickTick»

Источник: интернет-ресурс

Достоинства:

кроссплатформенность;

планирование по времени.

уведомления и напоминания.

Недостатки:

сложность интерфейса;

ограниченная функциональность в бесплатной версии.

Одними из основных недостатков в рассмотренных аналогах стали сложность интерфейса и ограничения в функциональности в бесплатной версии. В ходе анализа аналогов были выявлены достоинства и недостатки, которые будут учтены в процессе разработки проекта.

**1.3 Требования к ПО**

Основная цель проекта заключается в разработке веб-приложения «TaskMaster», которое предоставит пользователям возможность создавать заметки о задачах, отслеживать выполнение дел, отмечать задачи как выполненные, удалять и просматривать завершенные задания за день. Главным фокусом при разработке этого приложения является создание простого и интуитивно понятного интерфейса для пользователей.

Разрабатываемый проект должен включать в себя следующие функции:

ввод заданий;

добавление заданий;

удаление заданий из блока ожидания;

отметка выполнения задания;

просмотр выполненных заданий;

удаление выполненных заданий.

В результате исследования предметной области и анализа требований, к веб-приложению были выделены следующие требования:

Приятный интерфейс: программный продукт должен представлять пользователю современный и приятный интерфейс;

Простой интерфейс: программный продукт должен иметь интуитивно понятный интерфейс, который позволит пользователям легко управлять своими задачами;

Удобное добавление и удаление задач: пользователи должны иметь возможность легко добавлять новые задачи в список, а также удалять или редактировать уже существующие задачи без лишних сложностей;

Просмотр выполненных задач: важным функционалом будет возможность просматривать список задач, что поможет пользователям отслеживать свою продуктивность и достижения;

Хранение данных: программный продукт должен обеспечивать надежное хранение данных о задачах пользователя, чтобы информация была

доступна и сохранена даже после выхода из приложения.

Приложение будет оптимизировано для использования на компьютерных версиях браузеров.

«TaskMaster» будет сочетать стильный и современный дизайн с простым и понятным интерфейсом, чтобы обеспечить удобство и эффективность использования для пользователей. [9]

Полный список требований к проекту представлен в ПРИЛОЖЕНИИ А.

**1.4 Выбор и обоснование выбора средств реализации проекта**

Для разработки проекта «TaskMaster» будет использоваться интегрированная среда разработки Visual Studio Code, которая предоставляет широкие возможности для написания и отладки кода на языках HTML, CSS и JavaScript. Эта среда позволяет эффективно организовывать процесс разработки благодаря разнообразным расширениям и инструментам.

Браузер Opera будет использоваться для тестирования и просмотра разрабатываемого веб-приложения. Opera обеспечит возможность проверить, как приложение будет отображаться и взаимодействовать с пользователем в реальном окружении браузера.

Выбор языков разметки и программирования HTML, CSS и JavaScript обусловлен ключевой ролью данных языков в веб-разработке. HTML используется для создания структуры страницы, CSS — для стилизации элементов и создания привлекательного дизайна, а JavaScript — для добавления интерактивности и функциональности.

Для создания прототипа веб-приложения будет использоваться инструмент Figma. Figma предоставляет возможность детально проработать дизайн интерфейса, создать макеты экранов и провести пользовательские тестирования перед фактической разработкой.

Для хранения данных приложения будет использоваться механизм LocalStorage, предоставляемый браузером. LocalStorage позволит сохранять данные на стороне клиента, что обеспечит быстрый доступ к информации и сохранение задач даже при перезагрузке страницы или закрытии браузера.

Для проектирования базы данных разрабатываемого приложения будет использоваться AllFusion Process Modeller. Данная программа позволит наглядно представить структуру работы базы данных.

Для проектирования функционала и работы приложения будет использоваться сервис draw.io. С помощью данного сервиса можно представить различные диаграммы: диаграмма классов, прецедентов, последовательности и т.д.

Весь процесс разработки веб-приложения «TaskMaster» будет ориентирован на создание удобного и интуитивно понятного интерфейса, который позволит пользователям легко управлять своими задачами. Использование LocalStorage упростит процесс хранения данных, не требуя дополнительных серверов или баз данных, что сделает приложение более легким в поддержке и развертывании.