УЧРЕЖДЕНИЕ ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ

«ИНСТИТУТ ИТ И БИЗНЕС-АДМИНИСТРИРОВАНИЯ»

Кафедра ИТ и бизнес-администрирования

ДИПЛОМНАЯ РАБОТА

**«ДОМАШНЯЯ БУХГАЛТЕРИЯ»**

Слушателя *Куликовой Ларисы Александровны*

осваивающего образовательную программу переподготовки руководящих работников и специалистов, имеющих высшее образование, по специальности

*1-40 01 72 Проектирование программного обеспечения информационных систем группа 2.22*

Руководитель дипломной работы: старший преподаватель кафедры информационных систем и технологий факультета информационных технологий БГТУ,

А.В.Бурмакова

Допущена к защите.

«\_\_»\_\_\_\_\_\_\_\_\_\_20\_\_г.

Заведующий кафедрой ИТ и

бизнес-администрирования

кандидат технических наук

Г.А.Сачок

\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Минск, 2024

Аннотация

Пояснительная записка дипломного проекта содержит 70 страниц, 4 главы, 10 рисунков, 10 таблиц, 10 использованных источников и 10 приложений.

Phython, Django, MySQL, Pythonanywhere……

Цель проекта: Разработка веб-приложения "Домашняя бухгалтерия", предназначенного для автоматизации учета и управления личными финансами. Приложение должно обеспечить пользователям удобный и интуитивно понятный интерфейс для ведения учета доходов и расходов, анализа финансовых данных и генерации отчетов, что позволит повысить эффективность управления личными финансами и улучшить финансовую дисциплину.

Основные результаты: В результате решения поставленных задач было разработано веб-приложение "Домашняя бухгалтерия", которое обеспечивает пользователям удобный интерфейс для учета доходов и расходов, анализа финансовых данных и генерации отчетов. Это позволяет повысить эффективность управления личными финансами и улучшить финансовую дисциплину.

Рекомендации по внедрению: Рекомендуется внедрение разработанного веб-приложения для использования в личных и семейных финансовых планированиях, а также для малых предприятий, нуждающихся в простом и эффективном инструменте для управления финансами.

Оглавление

[**Введение** 3](#_Toc179186524)

[**Глава 1. Теоретико-методологические основы исследования** 5](#_Toc179186525)

[**1.1** **Сущность и актуальность исследуемой проблемы, задачи исследования.** 5](#_Toc179186526)

[**1.2. Обзор аналогичных программных продуктов для контроля финансов.** 5](#_Toc179186527)

[**1.3 Выбор программной платформы для создания Web-приложения.** 11](#_Toc179186528)

[**1.3.1 Язык программирования Python.** 12](#_Toc179186529)

[**1.3.2 Фреймворк Django** 13](#_Toc179186530)

[**1.3.3 Библиотека Pandas** 14](#_Toc179186531)

[**1.3.4 Облачная платформа PythonAnywere** 15](#_Toc179186532)

[**1.3.5 Облачная платформа GitHub.** 16](#_Toc179186533)

[**1.3.6 Система управления базами данных (СУБД) MySQL.** 16](#_Toc179186534)

[**1.3.7 Язык программирования SQL (Structured Query Language).** 17](#_Toc179186535)

[1.4 Выводы по разделу 18](#_Toc179186536)

# **Введение**

В современном мире существует множество программных решений для управления финансами. Однако большинство из них, несмотря на кажущуюся простоту и интуитивность интерфейса, требуют от пользователя значительных усилий для освоения. Среднестатистический работающий гражданин, имеющий доход, часто не располагает временем и желанием на изучение инструкций и документации, предпочитая приложения, которые понятны и удобны в использовании с первого взгляда.

Существуют и другие программы, которые настолько просты, что интерфейс понятен, но явно не хватает некоторых простых отчетов, необходимых для анализа целесообразности расходов.

Цель данного проекта — разработка веб-приложения "Домашняя бухгалтерия", предназначенного для автоматизации учета и управления личными финансами. Приложение должно обеспечить пользователям удобный и интуитивно понятный интерфейс для ведения учета доходов и расходов, анализа финансовых данных и генерации простых и понятных отчетов. Приложение направлено на повышение эффективности управления личными финансами, улучшение финансовой дисциплины и устранение раздражающих факторов, которые могут возникнуть при использовании других приложений.

С ростом цифровых технологий люди ожидают, что приложение можно будет использовать когда угодно и где угодно, с любого устройства, независимо от наличия подключения к сети. Поэтому в дальнейшем в приложение будут включены модули десктопного приложения и приложения для системы Android, помимо основного модуля веб-приложения.

Для реализации проекта будут использованы следующие языки программирования и технологии:

• Python и фреймворк Django для создания основного модуля веб-приложения "Домашняя бухгалтерия" с базой данных MySQL.

• PyQt для создания кроссплатформенного десктопного приложения с локальной базой данных SQLite.

• Java (с библиотеками Room, Retrofit, WorkManager и базой данных SQLite) для создания приложения на платформе Android.

Также важным аспектом являются финансовые затраты, которые может понести пользователь при оплате за использование приложения. Продукт должен быть доступен максимальному количеству пользователей, возможно даже с предоставлением отдельных функций на безвозмездной основе.

# **Глава 1. Теоретико-методологические основы исследования**

# **1.1 Сущность и актуальность исследуемой проблемы, задачи исследования.**

В связи с тем, что существующие приложения для индивидуального контроля финансов либо сложны в освоении, либо слишком просты для получения эффективных результатов, возникла потребность в создании простого и функционального приложения для управления финансами.

Основные задачи данного проекта:

1. Провести анализ существующих решений для управления домашними финансами.

2. Разработать архитектуру системы.

3. Реализовать функционал веб-приложения для учета доходов и расходов.

4. Обеспечить безопасность данных пользователей.

5. Провести тестирование системы.

# **1.2. Обзор аналогичных программных продуктов для контроля финансов.**

* Домашняя бухгалтерия, разработана Компанией Keepsoft.

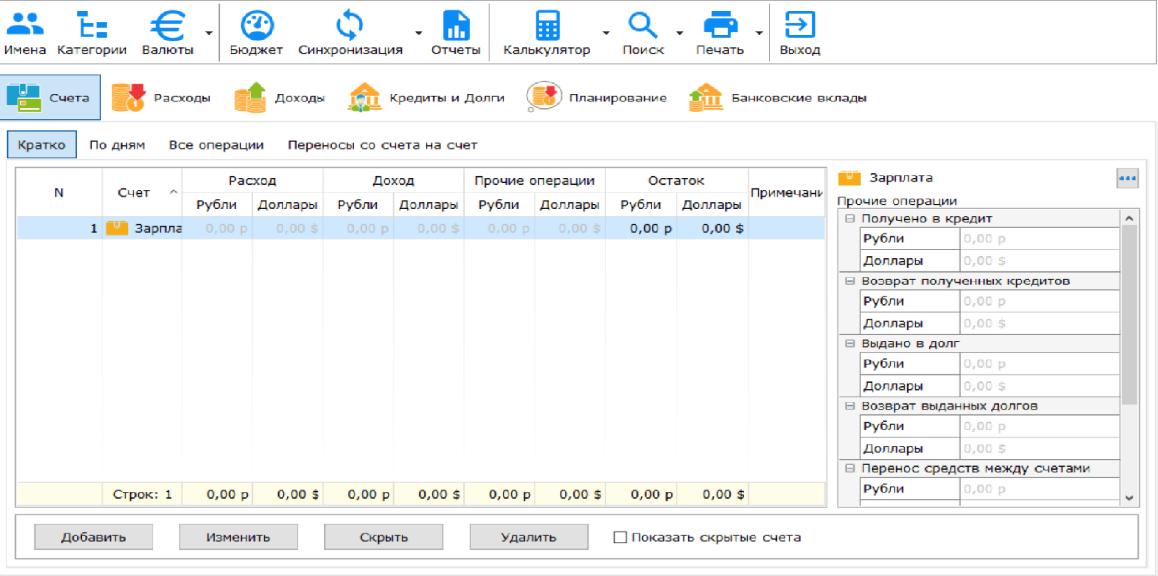


Рисунок 1.1.Интерфейс приложения "Домашняя бухгалтерия"

Плюсы:

Многофункциональность: Программа позволяет вести учет доходов и расходов, создавать бюджеты и анализировать финансовые данные.

Поддержка нескольких валют: Возможность вести учет в разных валютах одновременно.

Доступность на разных платформах: Программа доступна для Windows, Android и iOS.

Минусы:

Сложный интерфейс: Несмотря на многофункциональность, интерфейс программы может быть сложным для новых пользователей.

Отсутствие некоторых отчетов: Для получения детализированных отчетов, приходиться формировать из за каждый месяц.

Десктопное приложение.

* Экономка Онлайн – домашняя бухгалтерия в облаке, неизвестный разработчик.

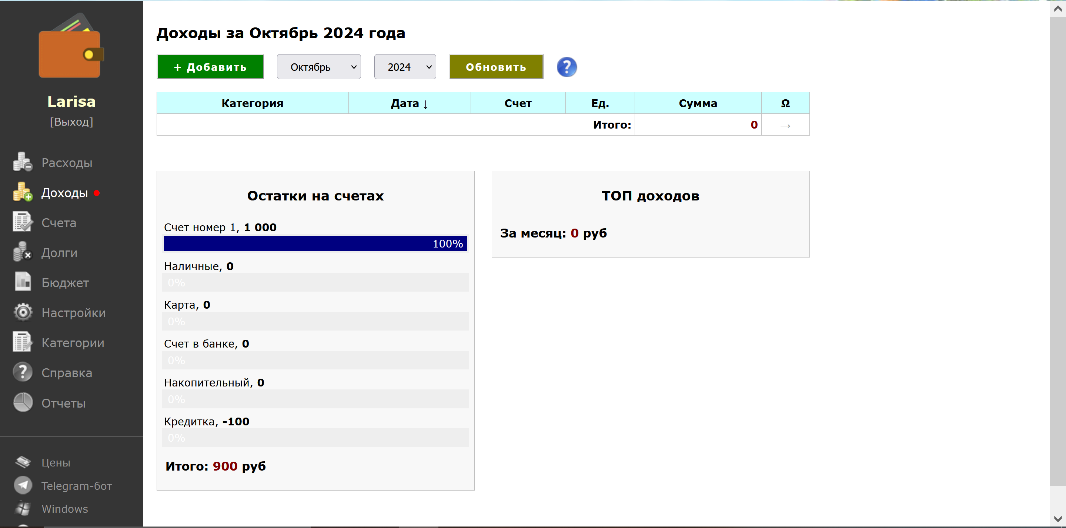


Рисунок 1.2 Интерфейс приложения Экономка

Плюсы:

Простой интерфейс: Легкость в использовании и доступность для пользователей с минимальными техническими навыками.

Облачное хранение: Данные хранятся в облаке, что обеспечивает доступ с любого устройства.

Минусы:

Отсутствие обработки ошибок: При регистрации нового пользователя некорректный e-mail не обрабатывается, что требует перезагрузки формы регистрации.

Ограниченные возможности ввода данных: Недостаточно полей для ввода детализированной информации, например, для учета расходов на автомобиль.

Отсутствие автоматического обновления курсов валют: Пользователю приходится вручную искать и вводить курсы валют.

Неудобство создания категорий: Нет возможности создать новую категорию дохода или расхода прямо со страницы ввода данных.

Непонятные графики: Некоторые графики расходов и доходов могут быть сложны для понимания.

* Mint.

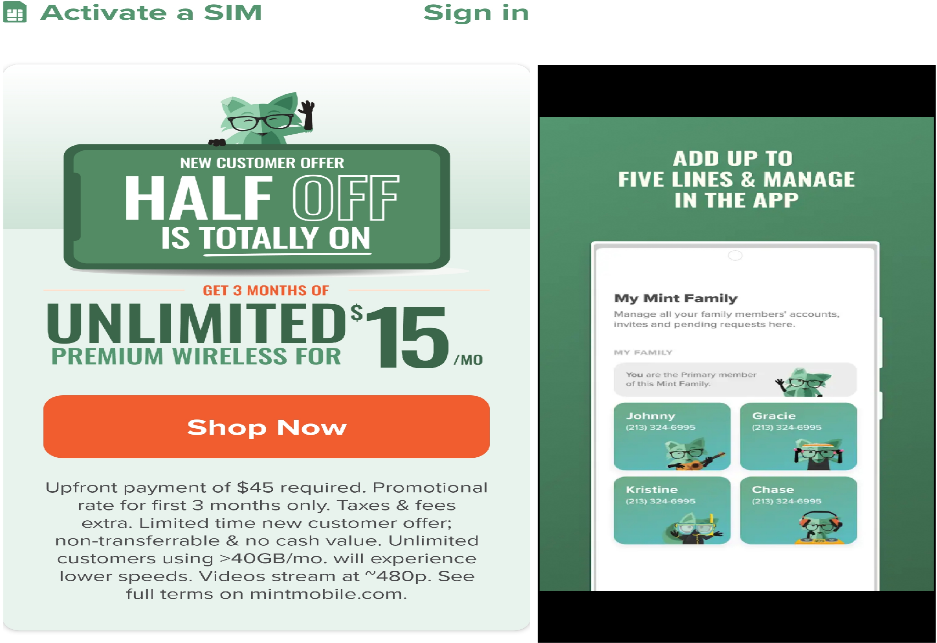


Рисунок 1.3 Приложение Mint

Плюсы:

Удобный интерфейс, автоматическое обновление данных, возможность планирования бюджета.

Минусы:

Ограниченные возможности для кастомизации, реклама в бесплатной версии.

* PocketSmith.

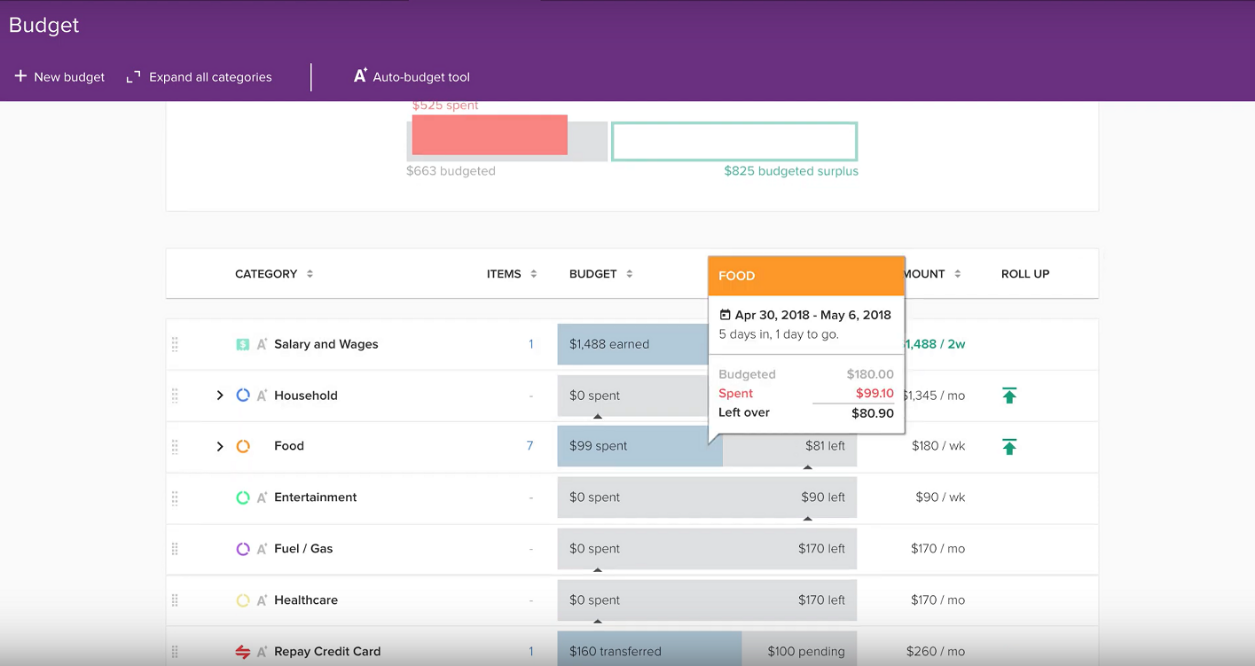


Рисунок 1.4 Пример интерфейса программа PocketSmith (вкладка бюджет)

Плюсы:

Поддержка нескольких валют, возможность прогнозирования денежных потоков, удобный интерфейс.

Минусы:

Высокая стоимость подписки для доступа ко всем функциям.

Нужно обучиться основным функциям



Рисунок 1.5. Учебник для программы PocketSmith

* Monefy.



Рисунок 1.6 Интерфейс приложения Monefy

Плюсы:

Простота использования. Интуитивно понятный и удобный интерфейс.

Быстрое добавление транзакций.

Наглядные графики и диаграммы для простого анализа финансов.

Возможность работы с разными валютами

Поддержка паролей для защиты данных

Минусы:

Некоторые полезные функции доступны только в платной версии.

Пользователи жалуются на отсутствие учебника.

Пользователи отмечают, что существуют проблемы с синхронизацией данных, если возникает потребность заменить устройство, на котором был установлен продукт на другое устройство.

Нет возможности создавать гибкие отчеты.

# **1.3 Выбор программной платформы для создания Web-приложения.**

Основные технологии указаны в Таблице 1.1 «Инструменты, технологии, библиотеки»

Таблица 1.1 «Инструменты, технологии, библиотеки»

|  |  |  |
| --- | --- | --- |
| Тип | Наименование | Назначение |
| Язык программирования | Python | Язык программирования для обработки и анализа данных |
| Фреймворк | Django | Для разработки Web-приложения на языке программирования Python |
| Библиотека | Pandas | Для обработки и анализа структурированных данных, построения графиков |
| Облачная платформа | PythonAnywere | Для запуска и редактирование web-приложений. Для хранения базы данных MySQL |
| Облачная платформа | GitHub | Для хранения и автоматической загрузки, измененного кода на облачный сервис PythonAnywhere |
| Система управления базами данных (СУБД) | MySQL | Для создания связанных таблиц и управления данными из этих таблиц |
| Язык программирования | SQL | Для управления и обработки данных в таблицах базы данных, а также для создания и управления базами данных и таблицами. |

Подробное описание каждой из выбранных технологий в подпунктах ниже по тексту.

# **1.3.1 Язык программирования Python.**

Python — это высокоуровневый язык программирования, мощный инструмент для создания программ самого разнообразного назначения, известный своей простотой и читаемостью. Python поддерживает несколько парадигм программирования, включая объектно-ориентированное, процедурное и функциональное программирование.

**Преимущества языка Python:**

• Динамическая типизация: В Python не нужно заранее объявлять тип переменной, что очень удобно при разработке.

• Хорошая поддержка модульности: Вы можете легко написать свой модуль и использовать его в других программах.

• Встроенная поддержка Unicode в строках: В Python необязательно писать всё на английском языке, в программах вполне может использоваться ваш родной язык.

• Поддержка объектно-ориентированного программирования: Реализация ООП в Python является одной из самых понятных.

• Автоматическая сборка мусора: Отсутствие утечек памяти.

• Интеграция с C/C++: Если возможностей Python недостаточно, можно интегрировать код на C/C++.

• Понятный и лаконичный синтаксис: Способствует ясному отображению кода. Удобная система функций позволяет при грамотном подходе создавать код, в котором будет легко разобраться другому человеку в случае необходимости. Также вы сможете научиться читать программы и модули, написанные другими людьми.

• Огромное количество модулей: Как входящих в стандартную поставку Python, так и сторонних. В некоторых случаях для написания программы достаточно лишь найти подходящие модули и правильно их скомбинировать. Таким образом, вы можете думать о составлении программы на более высоком уровне, работая с уже готовыми элементами, выполняющими различные действия.

• Кроссплатформенность: Программа, написанная на Python, будет функционировать одинаково вне зависимости от операционной системы. Отличия возникают лишь в редких случаях, и их легко заранее предусмотреть благодаря наличию подробной документации.

**Основные применения Python:**

• Веб-разработка: Используется для создания веб-приложений с помощью фреймворков, таких как Django и Flask.

• Научные исследования и анализ данных: Широко применяется в научных вычислениях, анализе данных и машинном обучении благодаря библиотекам, таким как NumPy, Pandas и TensorFlow.

• Автоматизация и скрипты: Python часто используется для написания скриптов для автоматизации задач.

• Разработка игр: С помощью библиотек, таких как Pygame, можно создавать игры.

• Образование: Python популярен в образовательных учреждениях благодаря своей простоте и доступности.

# **1.3.2 Фреймворк Django**

Django — это высокоуровневый веб-фреймворк на языке программирования Python, который используется для создания быстрых и безопасных веб-приложений. Он был разработан для упрощения процесса веб-разработки, позволяя разработчикам сосредоточиться на написании уникальных функций, а не на рутинных задачах.

**Преимущества Django:**

• Быстрая разработка: Позволяет быстро переходить от идеи к готовому продукту благодаря встроенным инструментам и шаблонам.

• Безопасность: Фреймворк включает в себя механизмы защиты от распространенных угроз, таких как SQL-инъекции и межсайтовый скриптинг (XSS).

• Масштабируемость: Django подходит для проектов любого размера, от небольших сайтов до крупных веб-приложений с высокой нагрузкой.

• Административный интерфейс: Встроенная админ-панель позволяет легко управлять контентом и данными приложения.

• Сообщество и документация: Django имеет большое сообщество разработчиков и обширную документацию, что облегчает обучение и решение возникающих проблем.

**Основные применения Django:**

Django в основном используется для создания различных веб-приложений и сайтов, от блогов и новостных порталов до сложных корпоративных систем. С помощью Django REST Framework можно легко создавать RESTful API для взаимодействия с клиентскими приложениями.

# **1.3.3 Библиотека Pandas**

Pandas — это мощная библиотека с открытым исходным кодом на языке Python, предназначенная для манипуляций с данными и анализа. Она предоставляет высокоуровневые структуры данных, такие как DataFrame и Series, которые позволяют эффективно работать со структурированными данными.

**Основные преимущества Pandas:**

• DataFrame и Series позволяют легко манипулировать табличными данными.

• Pandas хорошо интегрируется с NumPy, Matplotlib и другими библиотеками для научных вычислений и визуализации данных.

• Поддержка различных операций с данными, включая фильтрацию, группировку, агрегацию и преобразование данных.

• Поддержка множества форматов данных, таких как CSV, Excel, SQL и другие.

• Встроенные инструменты для работы с пропущенными значениями.

**Основные применения Pandas:**

• Используется для анализа и обработки больших объемов данных в различных областях, включая финансы, науку и маркетинг.

• Широко применяется в научных исследованиях для обработки экспериментальных данных.

• Подготовка данных для моделей машинного обучения.

• Создание графиков и диаграмм для представления данных.

# **1.3.4 Облачная платформа PythonAnywere**

PythonAnywhere — это облачная платформа для разработки и хостинга Python-приложений. Она позволяет писать, запускать и управлять Python-кодом прямо в веб-браузере.Это удобный инструмент для разработчиков, который позволяет быстро и эффективно разрабатывать и развертывать Python-приложения в облаке.**Основные предназначения:**• Возможность писать и редактировать код в веб-редакторе или запускать консольные сессии из любого современного браузера.• Платформа поддерживает различные веб-фреймворки, такие как Django и Flask, что позволяет легко развертывать веб-приложения.• Встроенная поддержка баз данных MySQL и SQLite.• Возможность синхронизации кода с GitHub для автоматического обновления приложений.**Основные преимущества:**• Можно работать с любого устройства, имеющего доступ к интернету.• Легко настраивать и управлять проектами без необходимости установки и настройки локальной среды разработки.• Подходит для небольших проектов и тестирования.• Поддержка Jupyter Notebook для анализа данных (платная версия).• Возможность масштабирования ресурсов в зависимости от потребностей проекта.

# **1.3.5 Облачная платформа GitHub.**

GitHub — это облачная платформа для хостинга и совместной разработки проектов с использованием системы контроля версий Git. Она предоставляет разработчикам инструменты для хранения, управления и совместной работы над кодом. Это мощный инструмент для разработчиков, который упрощает процесс разработки и управления проектами.

**Основное предназначение GitHub:**

• GitHub позволяет хранить исходный код проектов в репозиториях, обеспечивая доступ к ним из любой точки мира.

• Платформа поддерживает совместную работу над проектами, включая создание веток, слияние изменений и обсуждение кода.

• GitHub предоставляет инструменты для управления задачами, отслеживания ошибок и автоматизации рабочих процессов.

**Основные преимущества GitHub:**

• Облачное хранение позволяет работать с проектами из любой точки мира.

• Поддержка интеграции с различными инструментами для CI/CD, такими как Jenkins и Travis CI.

• GitHub является крупнейшей платформой для open-source проектов, что позволяет легко находить и участвовать в разработке открытого кода.

• Платформа обеспечивает высокий уровень безопасности данных и поддерживает управление доступом.

• Обширная документация и большое сообщество разработчиков облегчают обучение и решение возникающих проблем.

# **1.3.6 Система управления базами данных (СУБД) MySQL.**

MySQL — это реляционная система управления базами данных (СУБД) с открытым исходным кодом, основанная на языке SQL. Является одной из самых популярных СУБД благодаря своей надежности, производительности и простоте использования.

**Основное предназначение MySQL:**

• MySQL используется для хранения данных в структурированном виде, организованном в таблицы.

• Обеспечивает инструменты для добавления, обновления, удаления и извлечения данных.

• Поддерживает транзакции и ограничения ссылочной целостности для обеспечения надежности данных.

**Основные преимущества MySQL:**

• MySQL обеспечивает высокую скорость работы и производительность, что делает его подходящим для приложений с высокой нагрузкой.

• Поддерживает масштабирование от небольших приложений до крупных корпоративных систем.

• Легко изучается и используется благодаря понятному синтаксису и хорошей документации.

• Обеспечивает поддержку транзакций для обеспечения целостности данных.

• Работает на различных операционных системах, включая Windows, Linux и macOS.

• Распространяется под свободной лицензией GNU GPL, что позволяет свободно использовать, изменять и распространять программу.

# **1.3.7 Язык программирования SQL (Structured Query Language).**

SQL (Structured Query Language) — это язык программирования, предназначенный для управления и обработки данных в реляционных базах данных. Он был разработан для выполнения различных операций с данными, таких как создание, чтение, обновление и удаление данных. Является основным инструментом для работы с данными в реляционных базах данных, обеспечивая мощные и гибкие возможности для управления и обработки данных.**Основное предназначение SQL:**• SQL используется для добавления, изменения, удаления и извлечения данных из баз данных.• SQL позволяет создавать и изменять структуры данных, такие как таблицы, индексы и представления.• SQL поддерживает механизмы для обеспечения целостности и согласованности данных, такие как ограничения и транзакции.**Основное преимущество SQL:**• SQL является стандартным языком для работы с реляционными базами данных, что делает его универсальным и широко применимым в различных системах управления базами данных.

1.4 Выводы по разделу

В данном разделе были поставлены цель дипломного проекта и основные задачи. Проведен анализ пяти аналогичных программных продуктов, определены их плюсы и минусы, такие как недостаточность полезных отчетов, сложный интерфейс, высокая стоимость подписки, ограниченные возможности бесплатных версий, отсутствие автоматического обновления, отсутствие обработки ошибок и ограниченные возможности ввода данных.

В результате обзора технических средств для реализации данного проекта были выбраны следующие инструменты:

• Язык программирования Python для написания кода приложения.

• Фреймворк Django для удобства разработки веб-приложения на языке Python.

• Библиотека Pandas для обработки данных таблиц и составления графиков.

• Облачная платформа PythonAnywhere для развертывания приложения.

• Облачная платформа GitHub для синхронизации измененных данных между средой разработки и облачной платформой.

• Система управления базами данных MySQL для формирования баз данных, таблиц и управления данными с помощью языка SQL.

Глава 2. В разработке

**Список использованных источников**

ПРИЛОЖЕНИЕ А

Общая структура компонентов системы