Министерство образования Республики Беларусь

Учреждения образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНФОРМАТИКИ

И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра информатики

К защите допустить:

Заведующая кафедрой информатики

\_\_\_\_\_\_\_\_\_\_\_\_ Сиротко

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе

на тему

**АГРЕГАТОР ДАННЫХ И ПРОСМОТРЩИК ИГРОВОЙ**

**ИНФОРМАЦИИ**

БГУИР

БГУИР КП 1-40 40 01 113 ПЗ

Студент: Нгуен Б. К. М.

Руководитель: Давыдчик А. В.

Минск 2023

Оглавление

**Элементы оглавления не найдены.**

**ВВЕДЕНИЕ**

Не для кого не секрет, что в современном мире игровая индустрия занимает достаточно важное место в жизни людей. Профессиональным, а также простым игрокам всегда будет интересно посмотреть информацию о своей любимой игре: новости, интересующие категории контента, собственная статистика игры и так далее. Однако, не каждая игра имеет свою платформу для подачи такой нужной информации или же платформа не обладает инструментом, которые бы обрабатывал данные, необходимые пользователю.

В рамках курсового проекта будет разрабатываться проект под названием «Агрегатор данных и просмотрщик игровой информации». Этот проект направлен на создание веб-приложения, которое позволит пользователям получать доступ к разнообразным статистическим данным и информации о конкретной игре, включая, но не ограничиваясь, статистикой оружия, списком игроков, и многим другим.

Информация о видеоиграх становится всё более важной, как для профессиональных игроков, так и для обычных энтузиастов. Проект "Агрегатор данных и просмотрщик игровой информации" предоставит пользователю удобный и понятный интерфейс для просмотра статистики и анализа данных о выбранной игре. Кроме того, он обеспечит возможность проводить собственные исследования и анализ данных игровой статистики.

В рамках данной курсовой работы, мы подробно рассмотрим процесс разработки веб-приложения и создания базы данных с использованием MySQL, которая будет служить основой для хранения и обработки игровой информации. Мы также рассмотрим технологический стек, используемый в проекте, и обсудим методы сбора и анализа данных.

Целью этой работы является представление обзора процесса разработки агрегатора данных и просмотрщика игровой информации, а также анализ практической применимости баз данных MySQL в области игровой статистики.

**2 АНАЛИЗ РЫНКА ИГРОВЫХ СТАТИСТИЧЕСКИХ САЙТОВ**

На данный момент на рынке существуют ряд достаточно популярных игровых статистических сайтов для таких известных игр как Dota 2 –Dotabuff.com и Spectral.gg, League of legends – OP.gg и тд. Такого рода сайты предоставляют пользователям более обширную информацию о игре, включая данные обо всех официальных киберспортивных команд и их игроков. Также сайты обладают функцией, которая предоставляет пользователям анализ их игр для того, чтобы игроки могли улучшить свой игровой опыт. Такой подход и пользуется популярностью среди игроков.

Хороший пример этого явления – это компьютерная игра DOTA 2. Для оценки навыков игры используется численное значение, называемое очками рейтинга или MMR. Учитывая высокий порог входа в игру и множество нюансов в процессе игры, многие игроки имеют низкий MMR, даже имея довольно большое число отыгранных матчей.

**2.1 Концептуальная модель программного продукта**

Проанализировав рынок игровых статистических сайтов, было выбрано разработка аналога Dotabuff.org для игры Dota2, с целью улучшения перфоманса и качества просмотра контента.

Среди возможных функций, которыми бы хотели воспользоваться игроки:

1 Статистический анализ. Как было описано в параграфе 2 – подробные данные и анализ своих игр позволят игрокам улучшить свой игровой опыт, а также разрабатывать свои собственные стратегии.

2 Сравнение с другими игроками. Такая функция всегда вызывает интерес и усиливает конкуренцию между игроками, мотивирую их становиться лучше.

3 Новости и обновления. Игрокам также интересно последующее: обновления игры, какие события произойдут, какие турниры буду и другие новости в мире CS2.

4 Рейтинг и лидерство. Рейтинговая таблица стимулирует становиться лучше, а также позволяет игрокам знать на какой позиции находятся их любимая команда, либо игроки.

//Нужно будет схему сделать

**2.2 Функциональная карта ПП**

Функциональная карта отображает общую концепцию проекта.

Простой набор функций проекта:

1 Система авторизации. В приложении должна быть разработана система авторизации/регистрации, которая предоставляет пользователям возможность войти с использованием адреса электронной почты, либо использование входа аккаунтов социальных сетей, таких как Google, Facebook.

2 Редактирование. В своем кабинете необходимо реализовать функцию редактирования профиля, позволяющим изменять свои персональные данные и загружать фотографии профиля.

3 Фильтрация и поиск данных. Это могут быть данные об игроках, командах, либо фильтрация таких параметров как названия оружий и многое другое.

4 Перевод текстов. Интеграция функциональности текстов для пользователей, которые не владеют и не понимают английский язык, чтобы обеспечить им информацию на удобном для пользователей языке.

5 Навигация между разделами. Удобное переключения между интересующими разделами информации, чтобы пользователи могли найти быстро нужные данные.

6 Обратная связь. Отзыв пользователей, которые помогут узнать недостатки и положительные черты сайта.

7 Собственные данные. Просмотр собственной статистики своей игры.

Функциональная карта программного продукта наглядно изображен на рисунке 1.

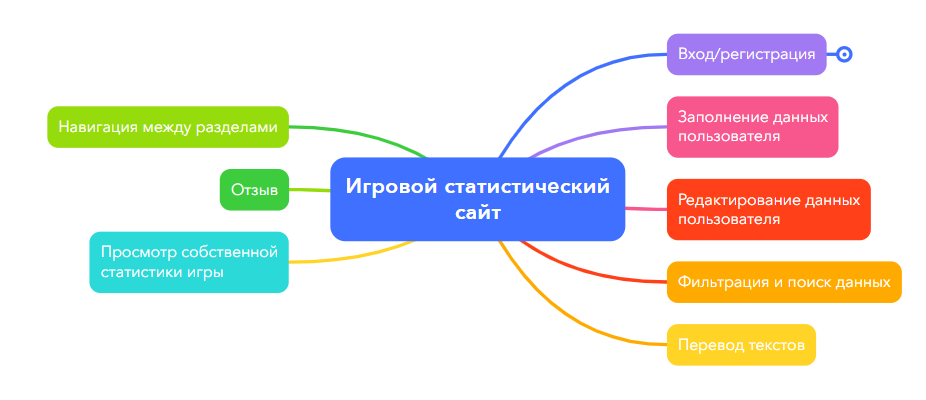


Рисунок 2.1 – Функциональная карта программного продукта

**2.3 Нефункциональные требования**

Удобство в использовании:

– интерфейс должен быть интуитивно понятным и легким в использовании для пользователей.

Локализация:

– система должна поддерживать несколько языков для удовлетворения потребностей разноязычных пользователей.

Требования к безопасности:

– система не должна позволять доступ к данным пользователям, которые не имеют доступ на их просмотр;

– необходимо создавать резервные копии данных.

**3 ВЫБОР ТЕХНОЛОГИЧЕСКОГО СТЕКА**

При разработке проекта "Агрегатор данных и просмотрщик игровой информации" были приняты важные решения, связанные с выбором технологий, которые определят архитектуру и функциональность веб-приложения. Ниже предоставлено объяснение причин выбора системы управления базами данных, языка программирования и фреймворка для разработки.

**3.1 Система управленияданных: MySQL**

Для хранения и управления данными агрегатора игровой информации был выбран MySQL в качестве СУБД (системы управления базами данных).

MySQL – это реляционная система управления базами данных (RDBMS), что означает, что записи хранятся не в одном хранилище, а в нескольких, отдельных и высоко систематизированных таблицах.

5 ключевых преимуществ MySQL:

1 Открытый исходный код и совместимость. Это означает, что третьим лицам дозволено изменять и настраивать исходный код. Совместимость же означает, что СУБД совместима с широким спектром систем, языков программирования, моделей баз данных, что позволяет нам интегрировать его в наше веб-приложение, созданное на Python.. MySQL также имеет широкие возможности для проектирования баз данных и моделирования данных (например, концептуальные модели данных или логические модели данных). Это делает ее простым и практичным вариантом для многих организаций, одновременно снижая опасения «привязки» к системе.

2 Популярность и стабильность: MySQL давно зарекомендовала себя как надежная СУБД с большим сообществом разработчиков и обширным опытом использования в разнообразных проектах.

3 Производительность. MySQL предлагает высокую производительность и хорошую оптимизацию запросов, что важно для эффективной обработки игровой статистики и данных.

4 Масштабируемость. По мере увеличения объемов данных и нагрузки на пользователей хранилище БД должно масштабироваться. Оно должно быть способным справляться с дополнительной нагрузкой без падения производительности. MySQL можно масштабировать различными способами, обычно с помощью репликации, кластеризации или шардинга (или их комбинации). СУБД MySQL способна поддерживать и обрабатывать очень большие базы данных, хотя это, скорее всего, скажется на скорости.

5. Безопасность. Это всегда важный момент для компаний, поскольку им необходимо защитить конфиденциальные данные и защититься от кибератак. MySQL предлагает шифрование с помощью протокола Secure Sockets Layer (SSL), маскировку данных (data masking), плагины аутентификации и другие уровни безопасности для защиты целостности данных. Пакет MySQL Enterprise также включает защиту брандмауэра от кибератак. [1]

Однако, полный выбор СУБД будет зависеть от приоритетов, ресурсов и амбиций организации.

**3.2 Язык программирования: Python**

В качестве языка программирования для создания веб-приложения был выбран Python. Этот выбор обусловлен несколькими факторами:

1 Простота и читаемость. Python известен своей читаемостью и простотой синтаксиса, что делает его идеальным выбором для быстрой разработки.

2 Множество библиотек и фреймворков. Python обладает обширной экосистемой библиотек и фреймворков, что позволяет нам легко интегрировать разнообразные инструменты и решения в приложение.

3 Поддержка Django. Для создания приложение было решено использрвать фреймворк Django, который построен на Python. Django предоставляет множество готовых компонентов и инструментов для создания веб-приложений, что сильно ускоряет разработку.

4 Остальные достоинства. Мощность и быстрота, хорошее сочетание с другими языками, доступность на любых ОС, наличие открытого исходного кода. [2]

Эти факторы и стали причиной почему выбирается такой язык программирования Python.

3. Фреймворк: Django

Django – это бесплатный полнофункциональный фреймворк для создания веб-приложений. Написан на Python, имеет открытый исходный код. Если говорить по-простому, Django – это набор готовых компонентов, которые помогут вам быстро создавать веб-сайты с прагматичной и чистой архитектурой.

При создании веб-сайтов разработчикам часто требуются одни и те же компоненты. Например для таких задач, как аутентификация пользователей, загрузка файлов и внедрение систем CMS. Django предоставляет готовые решения, позволяющие не изобретать заново велосипед при создании каждого нового сайта.

Выбор именно данного фреймворка основана на следующих факторах:

1 Фреймворк для перфекционистов. Благодаря простому синтаксису программы, написанные на Django, чисты, читабельны и хорошо структурированы. Это помогает разработчикам воплощать свои идеи в жизнь и быстрее выполнять требования.

2 Скорость. Django был создан, чтобы помочь создавать программное обеспечение как можно быстрее. Философия Don’t Repeat Yourself (DRY), на которой построен Django, побуждает разработчиков повторно использовать написанный ими код для других проектов. Это существенно сокращает время, необходимое для разработки продукта.

3 Масштабируемость. Компоненты Django разделены, то есть их можно добавлять или удалять по мере необходимости, как элементы Lego. В зависимости от требований вашего продукта, масштаб проекта может быть увеличен или уменьшен путем изменения количества и сложности компонентов Django.

4 Безопасность. Django защищает приложения по умолчанию. Он устраняет некоторые из наиболее распространенных ошибок безопасности, связанных с Python. Кроме того, он защищает приложения от межсайтового скриптинга, подделки запросов и SQL-инъекций.

5 Работа с большинством баз данных. Объектно-реляционное отображение (ORM) Django совместимо с рядом популярных баз данных. Однако его ключевой особенностью является то, что он позволяет разработчикам работать с несколькими базами данных одновременно. Более того, Django позволяет переходить из одной базы данных в другую и выполнять стандартные операции без необходимости написания большого количества дополнительного кода. [2]

Помимо этих фактором существует еще множество других преимуществ, по которой следует выбрать данный фреймворк для создания собственного приложения.