

Содержание

1	Аналитическая часть	4
1.1	Анализ существующих решений	4
1.1.1	Российский рынок	4
1.1.2	Зарубежный рынок	6
1.2	Формализация задачи	8
1.2.1	Список серверов	8
1.2.2	Рейтинг серверов	8
1.2.3	Деление серверов по платформам	8
1.2.4	Добавление/изменение/удаление серверов	9
1.3	Описание типов пользователей	9
1.4	Формализация данных	10
1.5	Выбор модели базы данных	12
1.5.1	Реляционные базы данных	12
1.5.2	NoSQL базы данных	13
1.6	Вывод	14
	Литература	15

1 Аналитическая часть

В данном разделе будет проведен анализ существующих решений на русском и зарубежном рынках. Также будет произведена формализация задачи и данных, описание типов пользователей, а также обзор существующих баз данных.

1.1 Анализ существующих решений

Существует большое количество сайтов, которые предоставляют возможность найти сервера для той или иной онлайн игры.

1.1.1 Российский рынок

Одним из сайтов для поиска серверов для игры «Minecraft» [1] является «MinecraftRating» [2]. При этом интерфейс главной страницы представлен на рисунке 1.1. На нем предоставляется большое количество функциональностей, таких как:

- поиск серверов (с сортировкой по самым популярным, по версии, по количеству игроков);
- получение полной информации о каждом сервере;
- добавление нового сервера;
- для зарегистрированных пользователей — добавление в избранное.

При этом не имеется возможности посмотреть сервера для разных платформ.

Для компьютерной игры «Counter-Strike» [3] также существуют сайты с серверами. Примером такого сайта является «Сервера КС» [4]. На рисунке 1.2 представлен интерфейс главной страницы. Для данной игры рынок развит слабее, поэтому функциональностей куда меньше. При этом предоставляются следующие возможности:

- просмотр серверов (присутствует лишь список, поиск и сортировка невозможны);
- для зарегистрированных пользователей — добавление нового сервера, добавление в избранное.

Из-за отсутствия сортировок серверов, имеется возможность купить место в верху таблицы, чтобы пользователи замечали сначала проплаченные сервера.

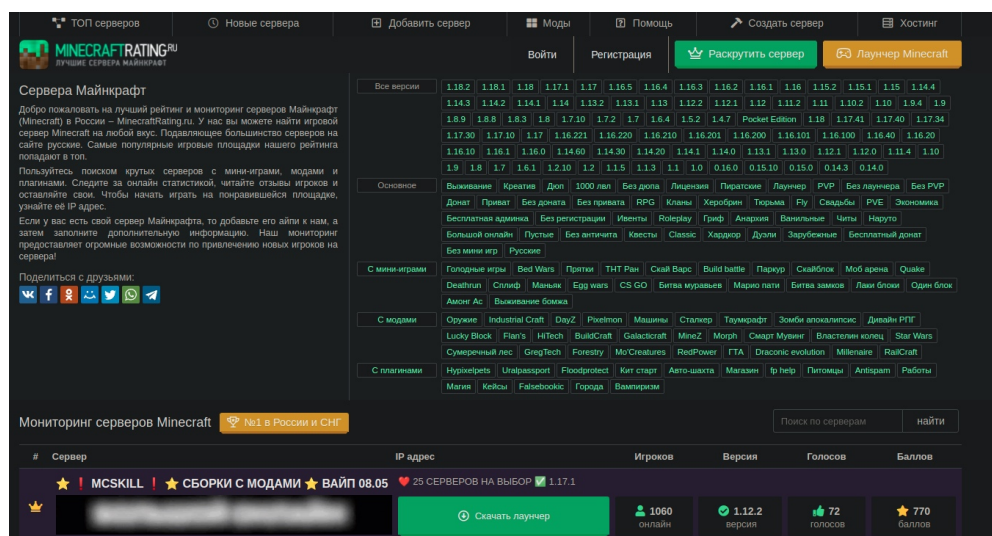


Рисунок 1.1 – Сайт для поиска серверов для игры «Minecraft» на русском рынке

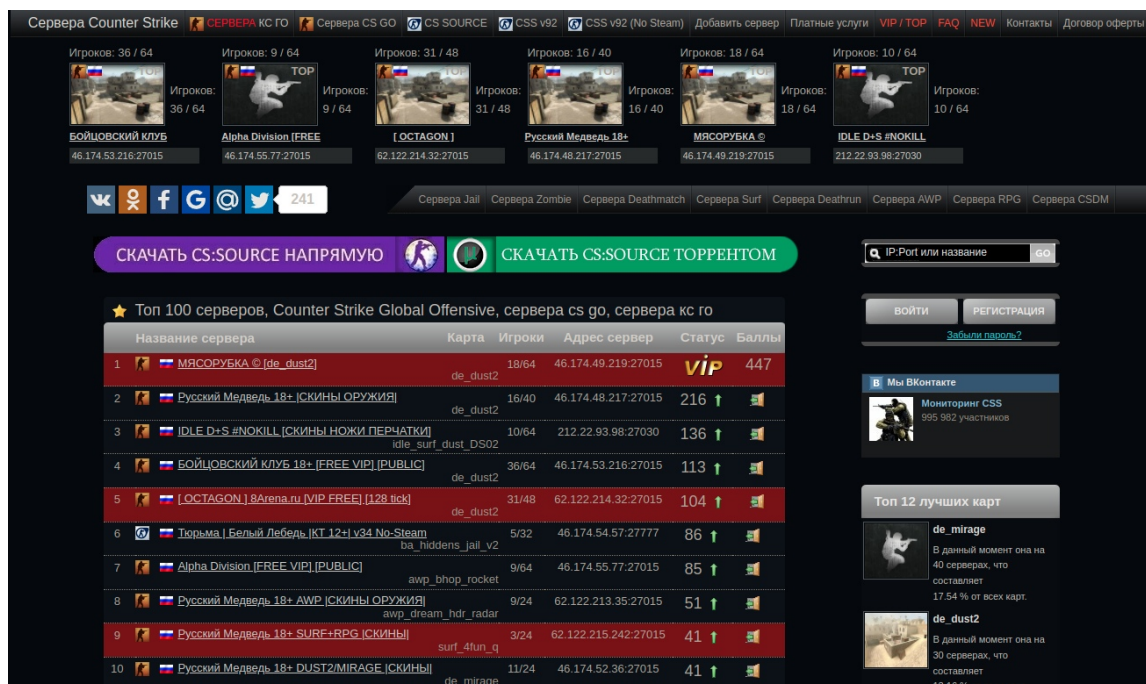


Рисунок 1.2 – Сайт для поиска серверов для игры «Counter-Strike» на русском рынке

1.1.2 Зарубежный рынок

Примером сайта для поиска серверов для игры «Minecraft» [1] является «Minecraft Servers» [5]. На рисунке 1.3 представлен интерфейс главной страницы. Зарубежный аналог обладает тем же самым функционалом, что и пример сайта с российского рынка. При этом поиск серверов для различных платформ также отсутствует.

Для игры «Counter-Strike» [3] зарубежный рынок развит сильнее. Так, примером является сайт «Game Tracker» [6]. Интерфейс главной страницы представлен на рисунке 1.4. Он обладает тем же самым функционалом, что и его российский аналог, а также:

- присутствует сортировка по различным параметрам;
- существует возможность выбора серверов для различных стран мира.

При этом возможность покупки приоритетного места в списке отсутствует.

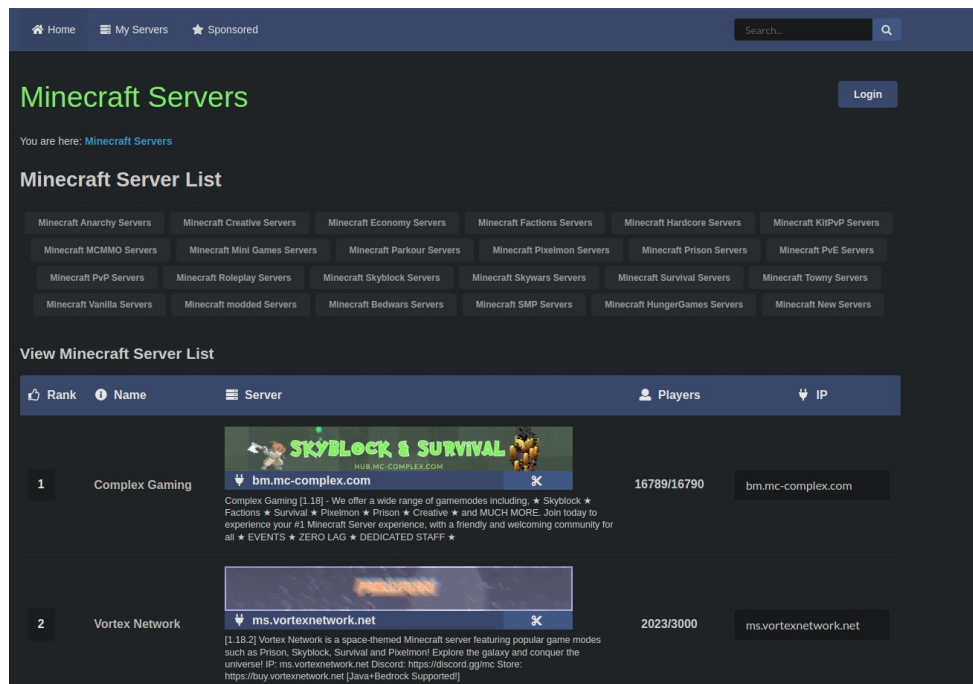


Рисунок 1.3 – Сайт для поиска серверов для игры «Minecraft» на зарубежном рынке

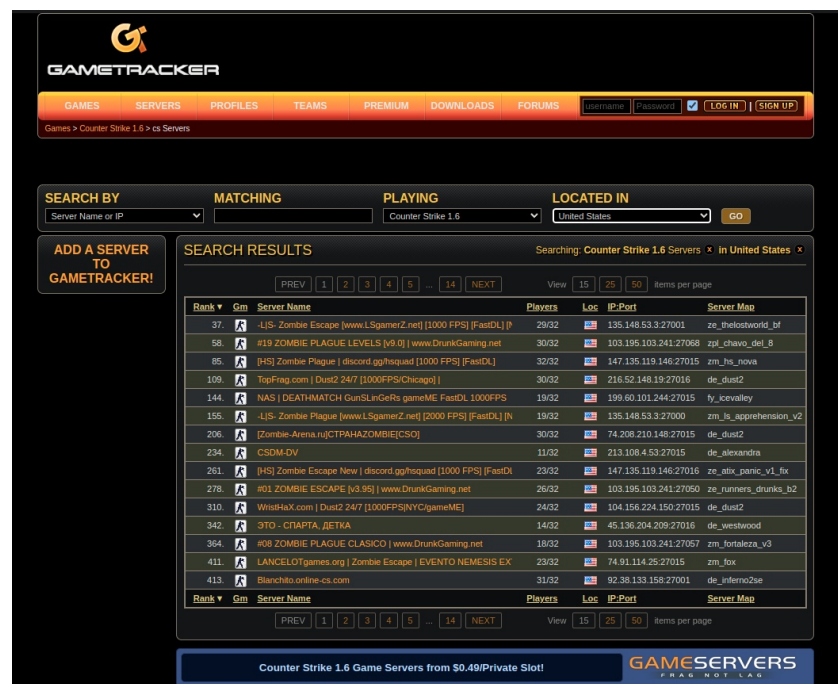


Рисунок 1.4 – Сайт для поиска серверов для игры «Counter-Strike» на зарубежном рынке

Вывод

Таким образом, российский и зарубежный рынки предоставляют множество различных сайтов для нахождения серверов для игр. Стоит отметить

то, что при этом сайты обладают рядом серьезных проблем.

1. Отсутствует возможность просмотреть списки серверов для разных платформ одной и той же игры.
2. Недостаточная функциональность для поиска необходимых серверов.
3. Слабая развитость рынка для некоторых игр.

1.2 Формализация задачи

Формализуем основные моменты разрабатываемого продукта.

1.2.1 Список серверов

Основным процессом является просмотр списка серверов. Список серверов представляет из себя таблицу из всех имеющихся серверов в базе данных. Информационные поля сервера указаны в колонках таблицы, а каждый отдельный сервер — строка этой таблицы.

1.2.2 Рейтинг серверов

Рейтинг сервера формируется из количества добавлений данного сервера в список избранных серверов отдельно взятого пользователя. Данный рейтинг выводится в качестве поля в таблице серверов, предоставляя возможность пользователям узнать наиболее популярный сервер. При этом, если пользователь сайта удалил данный сервер из своего списка избранных серверов, то рейтинг сервера понизится.

1.2.3 Деление серверов по платформам

Каждый сервер может находиться лишь на одной единственной платформе. Поэтому важно разделить список серверов на отдельные списки для каждой

платформы, чтобы пользователь мог выбрать именно те сервера, которые подходят для его рабочего устройства.

1.2.4 Добавление/изменение/удаление серверов

Данной возможностью наделен лишь администратор сайта. Должен быть предоставлен интерфейс для данного процесса. При добавлении/изменении должны быть добавлены ограничения на ввод информации, чтобы предотвратить ошибки в таблице серверов. При удалении должно быть реализовано подтверждение удаления сервера, чтобы предотвратить случайные нажатия.

1.3 Описание типов пользователей

В данной задаче должно быть выделено 3 типа пользователей (таблица 1.1).

Таблица 1.1 – Типы пользователей

Тип	Функциональность
Гость (неавторизованный пользователь)	<ul style="list-style-type: none">● Просмотр списка серверов● Выбор платформы для списка серверов● Сортировка списка серверов по доступным критериям● Регистрация● Авторизация
Авторизованный пользователь	<ul style="list-style-type: none">● Добавление сервера в список избранных серверов● Выбор платформы для списка серверов● Просмотр списка серверов● Сортировка списка серверов по доступным критериям

Администратор	<ul style="list-style-type: none"> • Добавление нового сервера на сайт • Удаление сервера с сайта • Изменение информации о сервере • Просмотр информации о хостинге для каждого сервера • Добавление сервера в список избранных серверов • Просмотр списка серверов • Выбор платформы для списка серверов • Сортировка списка серверов по доступным критериям
---------------	---

1.4 Формализация данных

В соответствии с задачей и типами пользователей, база данных должна содержать следующие данные (таблица 1.2).

Таблица 1.2 – Типы пользователей

Таблица	Данные
Сервер	<ul style="list-style-type: none"> • ID • Название • IP адрес • Версия игры • Рейтинг
Платформа	<ul style="list-style-type: none"> • ID • Название • Популярность • Стоимость
Хостинг	<ul style="list-style-type: none"> • ID • Название • Плата в месяц

Игрок	<ul style="list-style-type: none"> • ID • Никнейм • Сыграно часов на сервере • Дата последнего захода на сервер
Пользователь	<ul style="list-style-type: none"> • ID • Логин • Пароль • Статус

Также на рисунке 1.5 представлена ER-диаграмма [7] разрабатываемой системы.

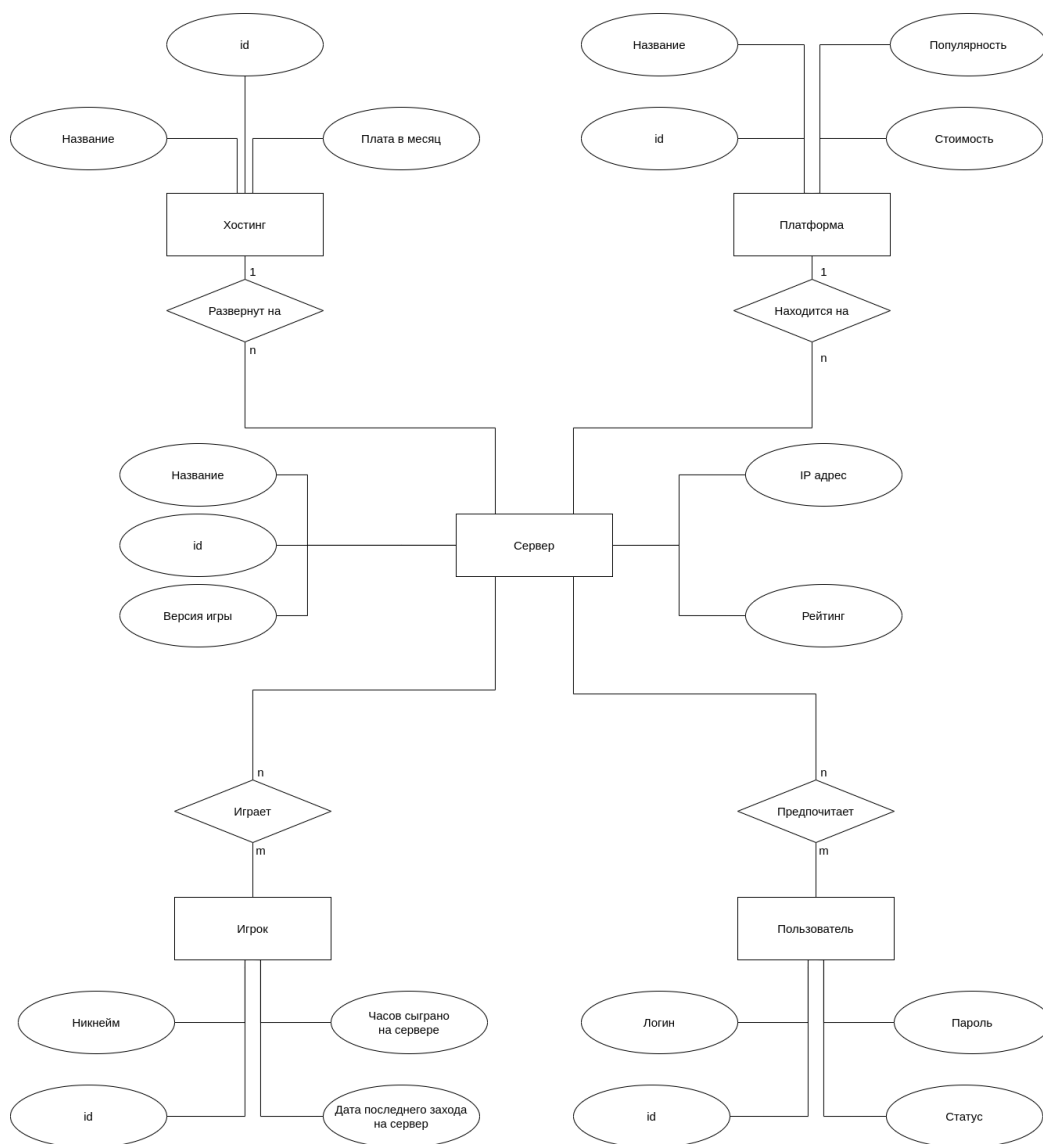


Рисунок 1.5 – ER-диаграмма

1.5 Выбор модели базы данных

База данных [8] — упорядоченный набор структурированной информации или данных, которые обычно хранятся в электронном виде в компьютерной системе. База данных обычно управляется системой управления базами данных (СУБД). Данные вместе с СУБД, а также приложения, которые с ними связаны, называются базой данных.

Данные в наиболее распространенных типах современных баз данных обычно хранятся в виде строк и столбцов формирующих таблицу [8]. Этими данными можно легко управлять, изменять, обновлять, контролировать и упорядочивать.

Базы данных делятся на две основные модели:

- реляционные базы данных (SQL базы данных);
- NoSQL базы данных.

1.5.1 Реляционные базы данных

Реляционная база данных [9] — совокупность отношений, которые содержат всю информацию, которая должна храниться в базе данных. Каждое отношение — двумерная таблица, в каждой строке которой хранится запись об объекте, а в каждом столбце — свойства данного объекта. При этом для каждой записи обязательно должен присутствовать атрибут — первичный ключ. Он однозначно идентифицирует запись в таблице базы данных.

В реляционных базах данных соблюдается принцип ACID:

- атомарность — для каждой транзакции либо выполняются все операции внутри нее, либо ни одной;
- согласованность — транзакция не может перевести базу данных в несогласованное состояние;
- изолированность — на результат транзакции не влияют другие транзакции;

- долговечность — любые изменения, которые были внесены транзакцией, остаются в базе данных при любой ситуации.

Реляционные базы данных обладают рядом преимуществ:

- использование стандартизированного языка запросов SQL [?];
- отображает всю информацию в простой и единой форме — отношении;
- основано на математической модели, которая строго описывает основные операции над данными.

Недостатками реляционные баз данных являются:

- медленный доступ к данным;
- вертикальная масштабируемость;
- трудоемкость разработки.

1.5.2 NoSQL базы данных

NoSQL база данных [10] — база данных, в которой в отличие от большинства традиционных систем баз данных не используется табличная схема строк и столбцов. В этих базах данных применяется модель хранения, оптимизированная под конкретные требования типа хранимых данных.

NoSQL базы данных делятся на 4 типа:

- коллекции — документы, упорядоченные по группам;
- ключ-значение — хэш-таблица, в которой по ключу находится значение;
- колоночная — хранит информацию в виде разреженной матрицы, строки и столбцы которой используются как ключи;
- графовые — сетевая база, использующие узлы и ребра для хранения данных.

NoSQL базы данных обладают следующими преимуществами:

- высокая скорость доступа к данным;
- горизонтальная масштабируемость;
- хранить информацию можно неструктурированной.

При этом NoSQL базы данных обладают рядом недостатков:

- привязка к одной СУБД, так как нет стандартизированного языка запросов;
- смягчение требований ACID.

Вывод

С учетом преимуществ и недостатков рассмотренных моделей баз данных и выполняемой задачи была выбрана реляционная модель базы данных.

1.6 Вывод

В данном разделе была проанализирована выполняемая задача — была проведена ее формализация, проведена формализация данных, описаны типы пользователей. Также были рассмотрены модели базы данных и выбрана реляционная модель.

Литература

- [1] Официальный сайт Minecraft [Электронный ресурс]. Режим доступа: <https://www.minecraft.net/ru-ru> (дата обращения: 5.05.2022).
- [2] Сервера Майнкрафт «MinecraftRating» [Электронный ресурс]. Режим доступа: <https://minecraftrating.ru/> (дата обращения: 5.05.2022).
- [3] Официальный сайт Counter-Strike [Электронный ресурс]. Режим доступа: <https://blog.counter-strike.net/> (дата обращения: 5.05.2022).
- [4] Сервера Counter-Strike «Сервера КС» [Электронный ресурс]. Режим доступа: <https://servera-csgo.ru/> (дата обращения: 5.05.2022).
- [5] Сервера Майнкрафт «Best Minecraft Servers» [Электронный ресурс]. Режим доступа: <https://best-minecraft-servers.co/> (дата обращения: 5.05.2022).
- [6] Сервера Counter-Strike «Game Tracker» [Электронный ресурс]. Режим доступа: <https://www.gametracker.com/search/cs/> (дата обращения: 5.05.2022).
- [7] Нотации модели сущность-связь (ER диаграммы) [Электронный ресурс]. Режим доступа: <https://pro-prof.com/archives/8126> (дата обращения: 5.05.2022).
- [8] Что такое база данных? [Электронный ресурс]. Режим доступа: <https://www.oracle.com/cis/database/what-is-database/> (дата обращения: 5.05.2022).
- [9] Что такое реляционная база данных? [Электронный ресурс]. Режим доступа: <https://aws.amazon.com/ru/relational-database/> (дата обращения: 5.05.2022).
- [10] Нереляционные данные и базы данных NoSQL [Электронный ресурс]. Режим доступа: <https://docs.microsoft.com/ru-ru/azure/architecture/data-guide/big-data/non-relational-data> (дата обращения: 5.05.2022).