

ВВЕДЕНИЕ

Мир компьютерных игр развивается с каждым годом. Всего за каких-то 30 лет простые 2D разработки превратились в полномасштабные проекты, которые поражают своей графической составляющей и невероятным вниманием к деталям. Важной частью этого бурного роста стало появление онлайн в компьютерных играх. Теперь появилась возможность сыграть в любимую игру вместе со своими друзьями со всего мира.

В связи с этим стали появляться специальные сайты, которые собирают информацию обо всех серверах, которые доступны для данной игры и предоставляют пользователю возможность подобрать идеально подходящий для себя сервер, исходя из доступной информации. При этом практически каждый сайт обладает своими проблемами, которые будут описаны позже.

Таким образом, целью данного курсового проекта является создание информационной системы для серверов игры. Для достижения данной цели необходимо решить следующие задачи:

- рассмотреть существующие решения;
- выбрать модель хранения данных;
- разработать базу данных;
- выделить роли пользователей;
- выбрать необходимый набор технологий для разработки;
- создать программный продукт, который решает поставленную цель.

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 типов 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.

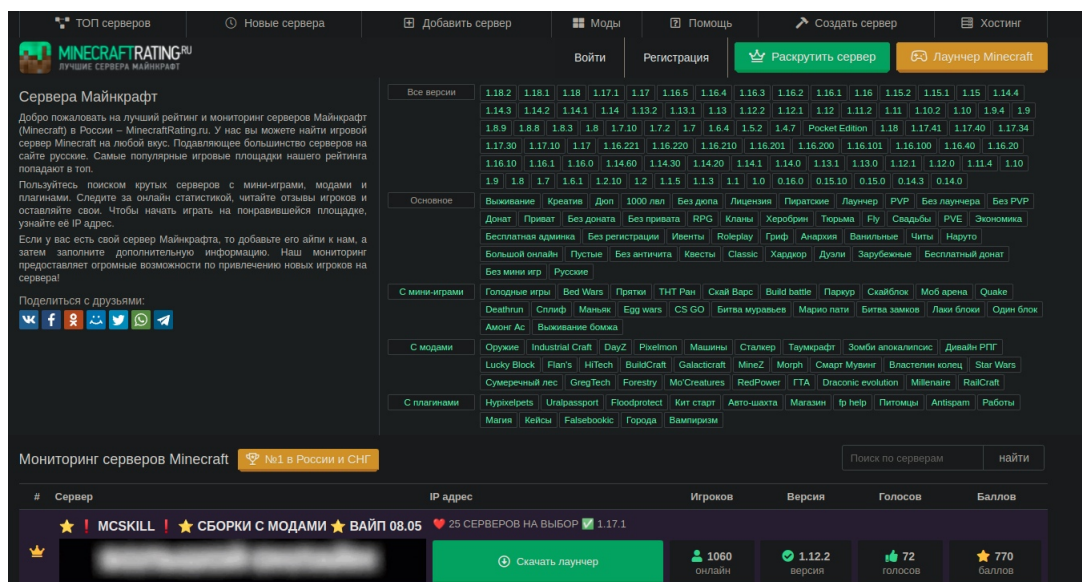


Рисунок 1.1 – Сайт для поиска серверов для игры «Minecraft» на русском рынке

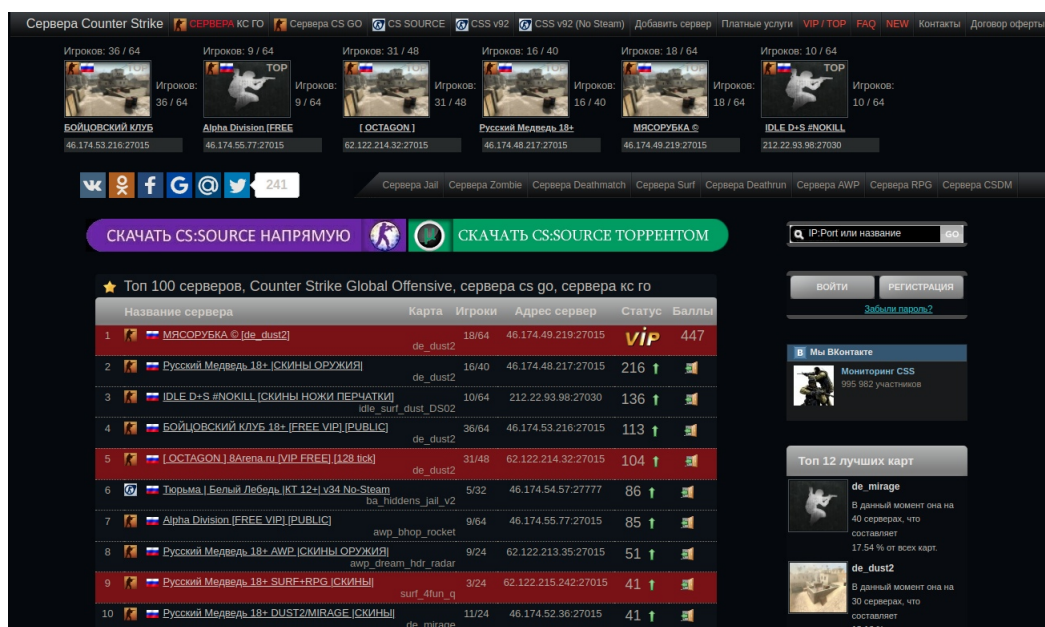


Рисунок 1.2 – Сайт для поиска серверов для игры «Counter-Strike» на русском рынке

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.

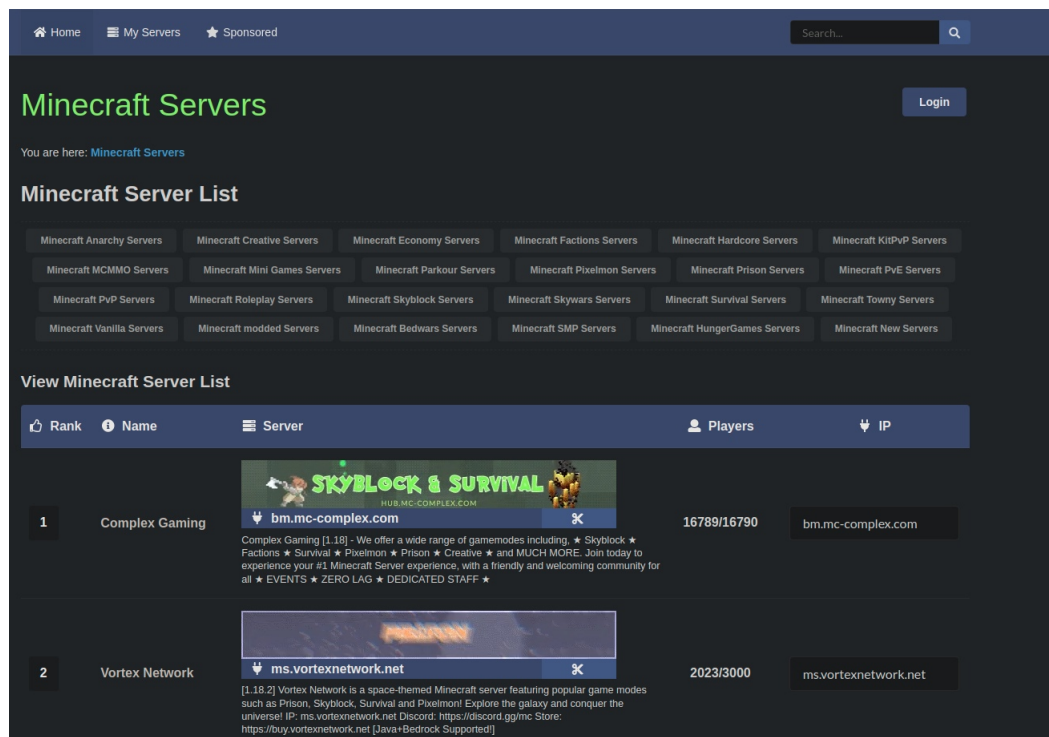


Рисунок 1.3 – Сайт для поиска серверов для игры «Minecraft» на зарубежном рынке

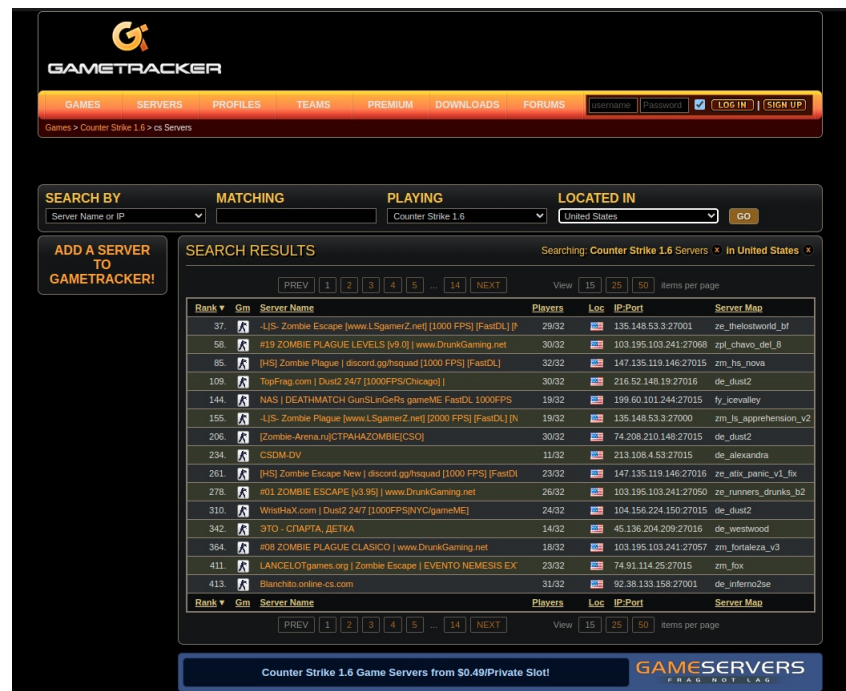


Рисунок 1.4 – Сайт для поиска серверов для игры «Counter-Strike» на зарубежном рынке

Вывод

В таблице 1.1 представлены результаты сравнения существующих решений. Выделенные критерии сравнения:

- К1 – просмотр списка серверов;
- К2 – сортировка списка серверов;
- К3 – просмотр серверов для разных платформ;
- К4 – добавление в избранное;
- К5 – получение подробной информации о сервере.

Таблица 1.1 – Сравнение существующих решений

Название	К1	К2	К3	К4	К5
«MinecraftRating»	+	+	-	+	+
«Сервера КС»	+	-	-	-	-
«Minecraft Servers»	+	+	-	+	-
«Game Tracker»	+	+	-	+	+

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

Должно быть разработано веб-приложение для поиска серверов для игры. При этом ПО должно содержать в себе следующие возможности:

- просмотр списка серверов с сортировкой по параметрам;
- авторизация и регистрация на сайте;
- просмотра списка избранных серверов;
- изменение рейтинга серверов;
- просмотр списка серверов для определенной платформы;
- добавление/изменение/удаление серверов администратором;
- изменение ролей пользователей администратором.

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.

Также таблица должна иметь возможность сортировки по полям сервера (названию, IP-адресу, версии игры, рейтингу сервера) и имени платформы, на которой запущен сервер.

1.2.2 Авторизация и регистрация

Должна быть введена возможность регистрации пользователя на сайте, чтобы открыть ему дополнительные возможности – просмотр информации о хостинге сервера, списка игроков сервера и добавления сервера в список избранного.

1.2.3 Список избранных серверов

Каждый зарегистрированный пользователь должен обладать возможностью добавить интересующий его сервер в список избранных. Это необходимо для того, чтобы пользователь не потерял интересующий его сервер и всегда имел быстрый доступ к информации о нем.

1.2.4 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.

1.2.5 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.

1.2.6 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 ввода, а также недопустить появления серверов с таким же названием или на том же IP-адресе. При удалении должно быть реализовано подтверждение удаления сервера, чтобы предотвратить случайные нажатия.

1.2.7 Изменение ролей зарегистрированных пользователей

Должна быть введена возможность изменения роли зарегистрированного пользователя. При этом администратор не может изменить собственную роль, а также администратору с никнеймом «admin» должно быть запрещено изменять роль, чтобы на сайте был всегда, как минимум, один администратор.

1.3 Описание типов пользователей

В задаче выделено 3 типа пользователей (таблица 1.2).

Таблица 1.2 – Типы пользователей

Тип	Функциональность
Гость (неавторизованный пользователь)	<ul style="list-style-type: none">• Просмотр списка серверов• Выбор платформы для списка серверов• Сортировка списка серверов• Регистрация• Авторизация
Авторизованный пользователь	<ul style="list-style-type: none">• Добавление сервера в список избранных• Выбор платформы для списка серверов• Просмотр списка серверов• Сортировка списка серверов• Просмотр информации о хостинге сервера, а также списка игроков сервера
Администратор	<ul style="list-style-type: none">• Добавление нового сервера на сайт• Удаление сервера с сайта• Изменение информации о сервере• Изменение ролей зарегистрированных пользователей• Добавление сервера в список избранных• Просмотр списка серверов• Выбор платформы для списка серверов• Сортировка списка серверов• Просмотр информации о хостинге сервера, а также списка игроков сервера

На рисунках 1.5-1.7 представлены Use-Case диаграммы выделенных типов пользователей.

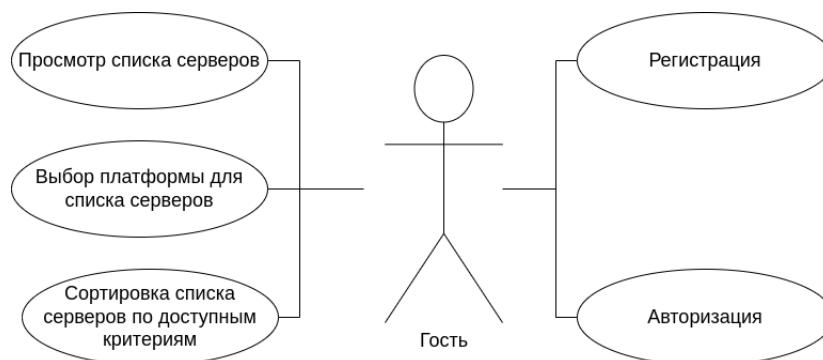


Рисунок 1.5 – Use-Case диаграмма для неавторизованного пользователя

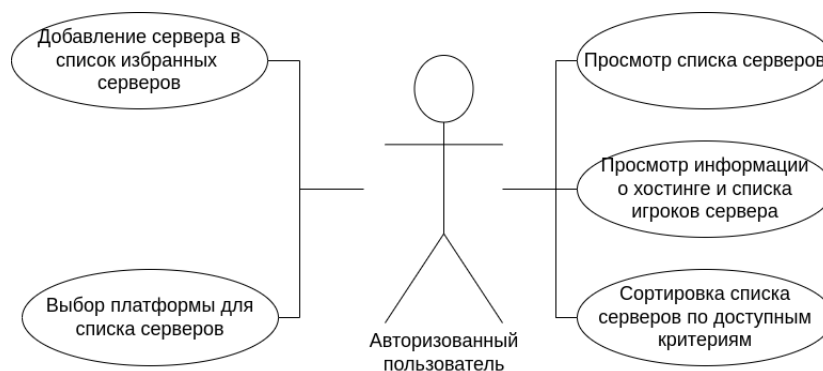


Рисунок 1.6 – Use-Case диаграмма для авторизованного пользователя

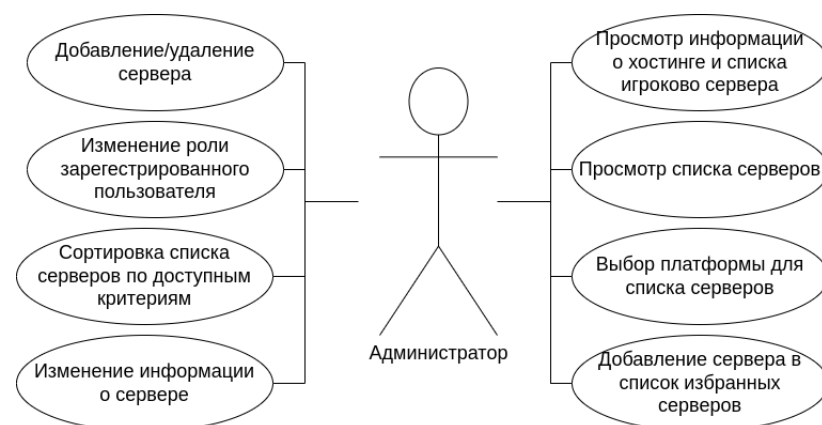


Рисунок 1.7 – Use-Case диаграмма для администратора

1.4 Формализация данных

В соответствии с задачей и типами пользователей, база данных должна содержать следующие модели (таблица 1.3):

- сервер;
- пользователь;
- платформа;
- игрок;
- хостинг.

Таблица 1.3 – Модели базы данных

Тип	Функциональность
Сервер	<ul style="list-style-type: none"> • ID • Название • IP-адрес • Версия игры • Рейтинг • ID платформы • ID хостинга
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 • Роль

Также на рисунке 1.8 представлена ER-диаграмма [7] разрабатываемой системы в нотации Чена.

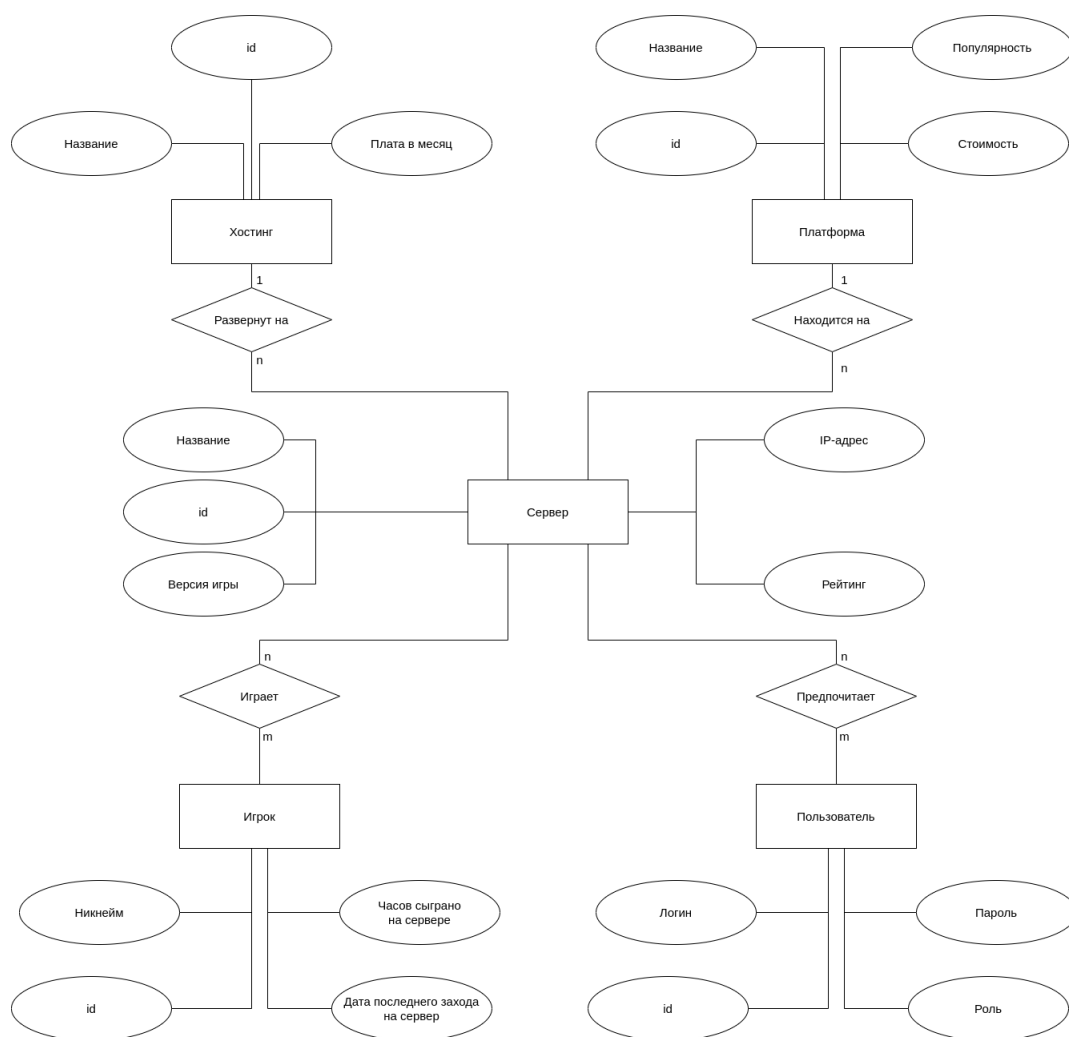


Рисунок 1.8 – ER-диаграмма в нотации Чена

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 управлять, изменять, обновлять, контролировать и упорядочивать.

Базы данных делятся на три модели организации данных:

- дореляционные;

- реляционные;
- постреляционные.

1.5.1 Дореляционные модели

Дореляционные модели базы данных [9] предоставляли доступ на уровне записей, которые располагались в виде древовидной структуры со связями предок-потомок. При этом взаимодействие с базой данных происходило с использованием языков программирования, которые были расширены функциями дореляционных СУБД.

Главными недостатками является то, что оптимизация доступа к данным со стороны системы отсутствует, а также древовидная структура является весьма трудоемкой.

1.5.2 Реляционные модели

Реляционная база данных [10] — 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.

Реляционные базы данных обладают несомненным преимуществом — благодаря стандартизированного языка запросов SQL, существует возможность подмены СУБД.

1.5.3 Постреляционные модели

В постреляционных моделях баз данных [11] не используется табличная схема строк и столбцов. В этих базах данных применяется модель хранения, оптимизированная под конкретные требования типа хранимых данных.

При этом они делятся на следующие основные категории:

- 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.

Данные модели используются для специфических задач, где явно подходит одна из приведенных выше категорий, что явно ускоряет работу программного продукта благодаря грамотной работе с данными.

Вывод

Для поставленной задачи разработки информационной системы наилучшим образом подходит реляционная модель хранения данных. Выбор обусловлен необходимостью хранить данные в виде структурированных таблиц, между которыми проведена связь.

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.

2 Конструкторская часть

В данном разделе приведено описание таблиц разрабатываемой базы данных, а также описаны функции, триггеры и роли базы данных.

2.1 Описание таблиц базы данных

Схема разрабатываемой базы данных приведена на рисунке 2.1.

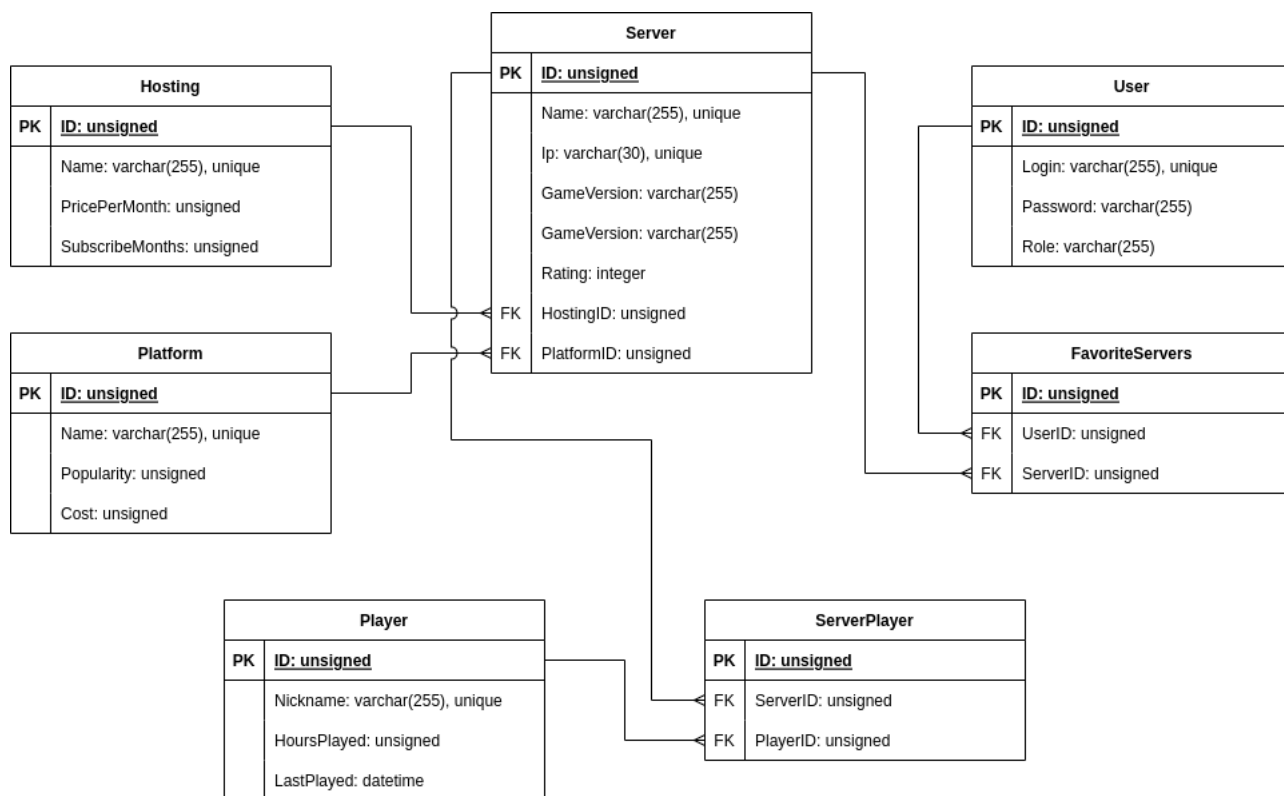


Рисунок 2.1 – Диаграмма базы данных

Реализуемая модель базы данных содержит 7 таблиц.

1. Таблица «Server» хранит информацию о серверах. Содержит следующие поля:

- ID – первичный ключ; тип данных – unsigned;
- Name – уникальное название сервера; тип данных – varchar(255);
- IP – уникальный IP-адрес сервера; тип данных – varchar(30);
- GameVersion – версия игры, для которой открыт данный сервер; тип данных – varchar(255);
- Rating – рейтинг сервера; тип данных – integer;

- HostingID – идентификатор хостинга; тип данных – unsigned;
 - PlatformID – идентификатор платформы; тип данных – unsigned;
2. Таблица «User» хранит информацию о пользователе сайта. Содержит следующие поля:
- ID – первичный ключ; тип данных – unsigned;
 - Login – уникальный логин на сайте; тип данных – varchar(255);
 - Password – пароль на сайте; тип данных – varchar(255);
 - Role – роль пользователя; тип данных – varchar(255);
3. Таблица «FavoriteServer» хранит информацию об избранных серверах игроков. Является связующей таблицей между таблицами «Server» и «User». Содержит следующие поля:
- ID – первичный ключ; тип данных – unsigned;
 - UserID – идентификатор пользователя; тип данных – unsigned;
 - ServerID – идентификатор сервера; тип данных – unsigned;
4. Таблица «Player» хранит информацию об игроке сервера. Содержит следующие поля:
- ID – первичный ключ; тип данных – unsigned;
 - Nickname – уникальный никнейм игрока; тип данных – varchar(255);
 - HoursPlayed – количество часов, сыгранных на серверах; тип данных – unsigned;
 - LastPlayed – дата крайнего раза захода на сервер; тип данных – unsigned;
5. Таблица «FavoriteServer» хранит информацию об игроках и на каких серверах они играют. Является связующей таблицей между таблицами «Server» и «Player». Содержит следующие поля:
- ID – первичный ключ; тип данных – unsigned;
 - PlayerID – идентификатор игрока; тип данных – unsigned;

- ServerID – идентификатор сервера; тип данных – unsigned;
6. Таблица «Hosting» хранит информацию о хостинге, на котором находится сервер. Содержит следующие поля:
- ID – первичный ключ; тип данных – unsigned;
 - Name – уникальное название хостинга; тип данных – varchar(255);
 - PricePerMonth – плата за размещение сервера на хостинге в месяц; тип данных – unsigned;
 - SubMonts – количество месяцев подписки сервера на хостинг; тип данных – unsigned;
7. Таблица «Platform» хранит информацию о платформах, на которых может быть запущена игра. Содержит следующие поля:
- ID – первичный ключ; тип данных – unsigned;
 - Name – уникальное название платформы; тип данных – varchar(255);
 - Popularity – популярность платформы; тип данных – unsigned;
 - Cost – стоимость платформы; тип данных – unsigned;

2.2 Функции базы данных

В проектируемой базе данных определены следующие функции.

1. **Функция выбора серверов.** Данная функция выбирает или все сервера из таблицы «Server», или только избранные сервера пользователя по его идентификатору. Схема алгоритма функции представлена на рисунке 2.2.
2. **Функция фильтрации серверов.** Данная функция фильтрует сервера из таблицы «Server», полученные после работы функции выбора серверов. Фильтрация происходит по названию сервера и/или идентификатору платформы. Схема алгоритма функции представлена на рисунке 2.5.
3. **Функция сортировки серверов.** Данная функция сортирует сервера из таблицы «Server», полученные после работы функции фильтрации серверов. Сортировка происходит по полям: название сервера, IP-адрес сервера, версия игры сервера, рейтинг сервера, названию платформы (все – по

возрастанию и по убыванию). Схема алгоритма функции представлена на рисунке 2.4.

4. **Функция парсинга серверов.** Данная функция является некой «прослойкой», которая возвращает все сервера из таблицы «Server», полученных после работы функции сортировки серверов. Схема алгоритма функции представлена на рисунке 2.3.

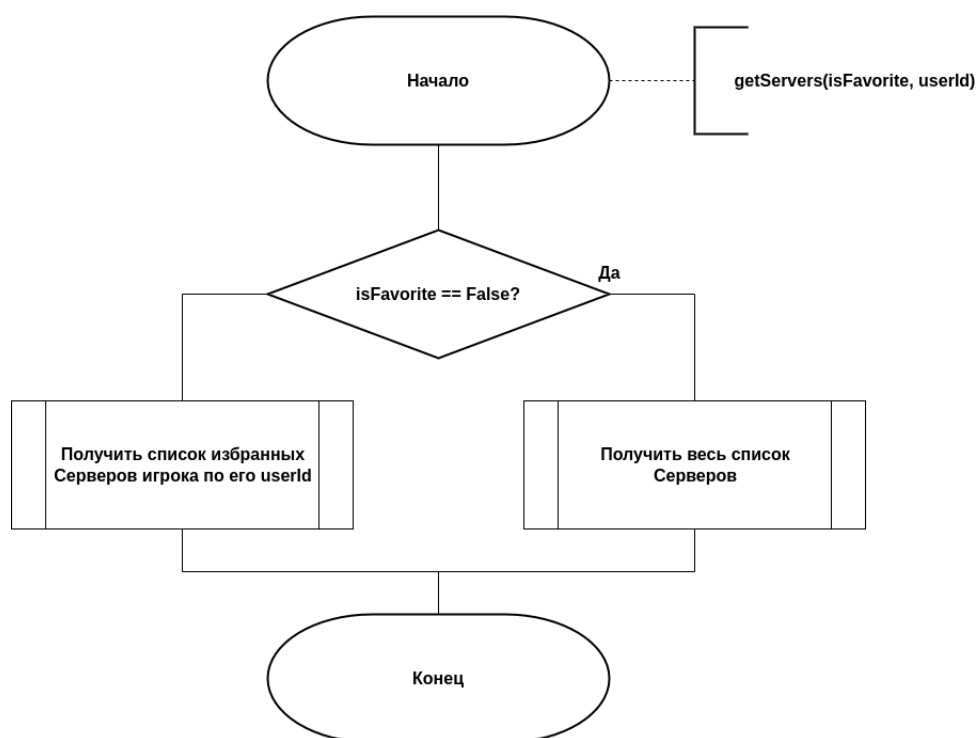


Рисунок 2.2 – Функция выбора серверов

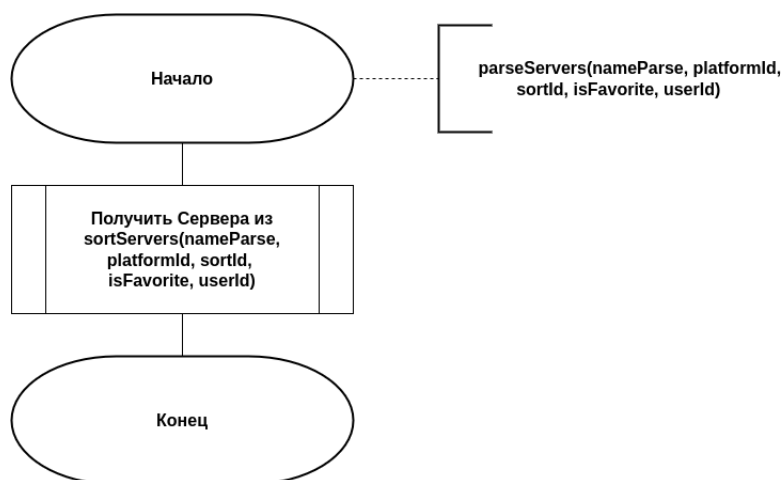


Рисунок 2.3 – Функция парсинга серверов

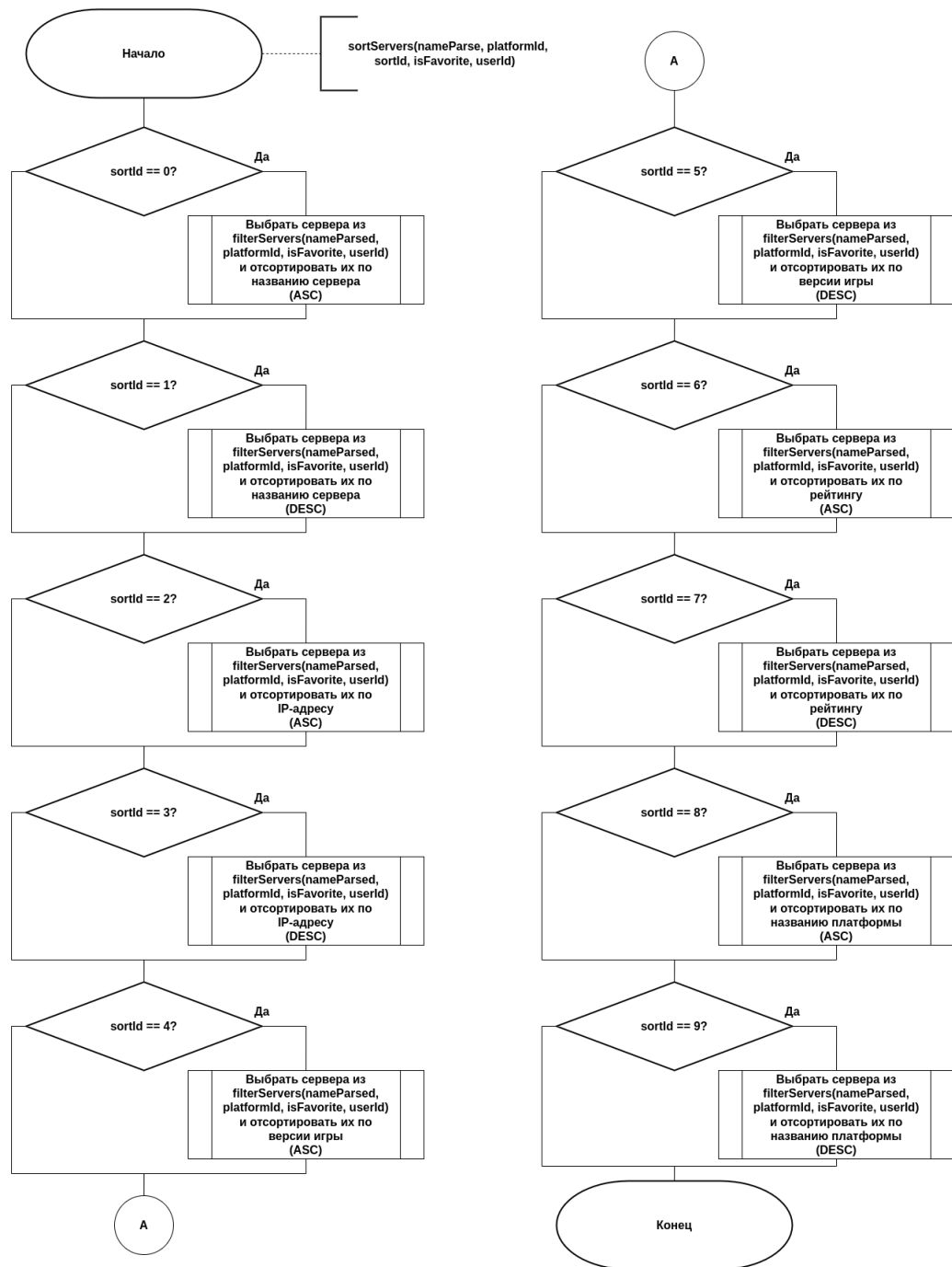


Рисунок 2.4 – Функция сортировки серверов

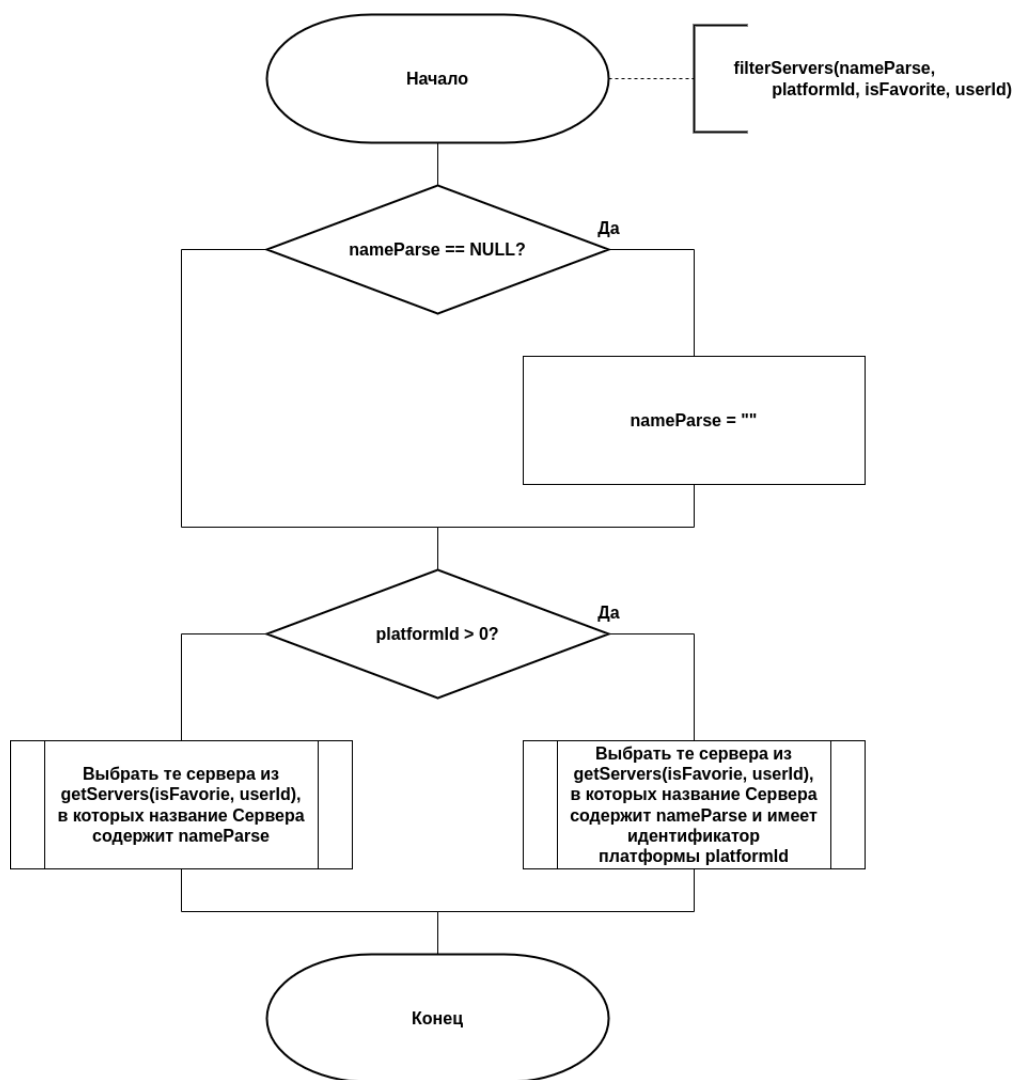


Рисунок 2.5 – Функция фильтрации серверов

2.3 Триггеры базы данных

В проектируемой базе данных определены следующие триггеры.

1. **Триггер установки роли пользователя.** Данный триггер срабатывает при операции «Insert» в таблицу «User» и вызывает функцию установки роли пользователя, которая выдает пользователю роль «User» или «Admin», если его логин «admin». Таким образом, в базе данных точно будет один администратор для управления сайтом. Схема алгоритма вызываемой функции представлена на рисунке 2.6.
2. **Триггеры изменения рейтинга сервера.** Один триггер срабатывает при операции «Insert» в таблицу «FavoriteServer» и вызывает функцию *увеличения* рейтинга добавленного сервера на единицу, а второй триггер –

при операции «Delete» из таблицы «FavoriteServer» и вызывает функцию *уменьшения* рейтинга удаляемого сервера на единицу. Схема алгоритма вызываемой функции представлена на рисунке 2.6.

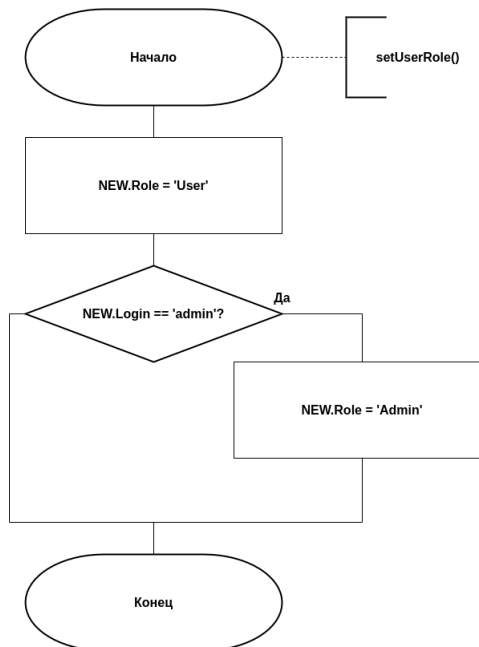


Рисунок 2.6 – Функция выдачи роли пользователю

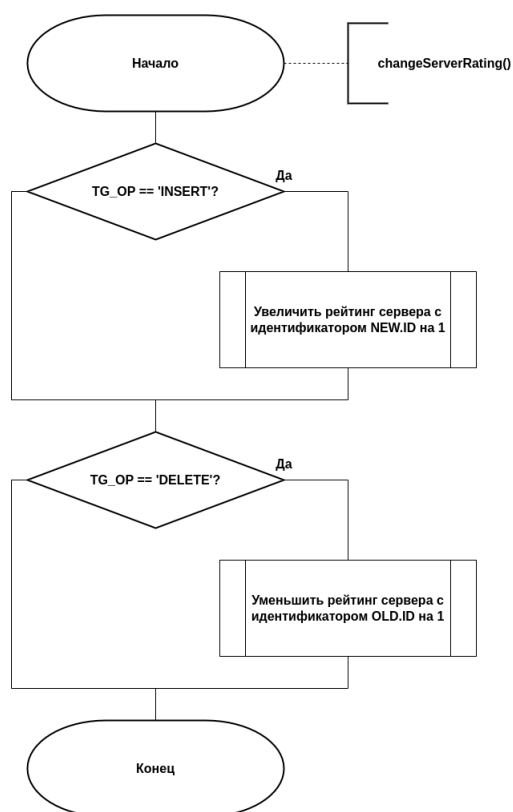


Рисунок 2.7 – Функция изменения рейтинга сервера

2.4 Роли базы данных

В проектируемой базе данных определены следующие роли.

1. **Неавторизованный пользователь.** Данная роль обладает возможность просмотра таблиц серверов и платформ, так как общий список серверов доступен всем видам пользователей. Также обладает возможностью добавлять в таблицу «User» для регистрации нового аккаунта.
2. **Авторизованный пользователь.** Данная роль обладает возможность просмотра таблиц серверов, платформ для списка серверов; просмотра таблиц «Hosting», «Player» и «ServerPlayer» для получения детальной информации о сервере; просмотра таблицы «FavoriteServer» для просмотра серверов из списка избранного. Также обладает возможностью добавлять и удаления в таблице «FavoriteServer» для добавления и удаления серверов из списка избранного.
3. **Администратор сайта.** Данная роль обладает возможность просмотра, добавления и удаления на все определенные в базе данных таблицы.

Вывод

В данном разделе были подробно описаны все поля всех таблиц проектируемой базы данных, ее функции, триггеры и роли.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

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

12. SQL [Электронный ресурс]. — URL: <https://blog.skillfactory.ru/glossary/sql/> (дата обращения: 5.05.2022).