

Оглавление

Введение	2
1 Аналитическая часть	3
1.1 Формализация задачи	3
1.2 Формализация данных	3
1.3 Формализация ролей	4
1.4 Use-case диаграмма	5
1.5 ER диаграмма	6
1.6 Анализ существующих решений	8
1.7 Описание существующих баз данных и систем управления базами данных	8
1.7.1 Классификация баз данных по способу хранения . . .	8
1.7.2 Выбор модели хранения данных для решения задачи	9
1.7.3 Обзор СУБД с построчным хранением	10
1.7.4 Выбор СУБД для решения задачи	11
Литература	13

Введение

В настоящее время количество частных инвесторов растет с каждым годом, так, согласно исследованиям [1] с начала 2021 года число физических лиц, имеющих брокерские счета на Московской бирже увеличилось на 6.2 миллиона человек и достигло 15 миллионов. В большинстве случаев выбор того или иного финансового инструмента у начинающего инвестора зависит от мнений авторитетных аналитиков. Как следствие, возникает необходимость ознакомления с большим количеством различных точек зрения. Ввиду отсутствия должного опыта пользователь вынужден посещать обширное количество интернет-ресурсов. Решением данной проблемы может послужить агрегация множества прогнозов в единое веб-приложение, которое предоставит информацию конечному пользователю в удобном и доступном виде.

Цель работы — реализовать программное обеспечение для добавления, хранения, редактирования и удаления данных о прогнозах на курсы акций.

Чтобы достигнуть поставленной цели, требуется решить следующие задачи:

- формализовать задание и определить необходимый функционал;
- проанализировать варианты представления данных и выбрать подходящий вариант для решения задачи;
- проанализировать системы управления базами данных и выбрать подходящую систему для хранения данных;
- спроектировать базу данных, описать ее сущности и связи;
- разработать программный интерфейс для работы с базой данных;
- реализовать программное обеспечение, которое позволит получить доступ к данным по средствам REST API [2].

1 Аналитическая часть

В данном разделе будет проанализирована поставленная задача и рассмотрены различные способы ее реализации.

1.1 Формализация задачи

В ходе выполнения курсовой работы должно быть спроектировано и реализовано Web-приложение, позволяющее пользователю ознакомиться с прогнозами курса акций различных аналитиков. Пользователь должен иметь возможность получения прогнозируемой цены одной из предложенных акций. Кроме того, необходимо разработать ролевую модель, на основе которой будет строиться взаимодействие с веб-приложением. Предусмотреть три роли: пользователь, аналитик и администратор. Также необходимо создать систему регистрации и авторизации пользователей и предусмотреть возможность назначения ролей администратором. Сделать возможным для аналитика добавление, редактирование и удаление прогнозов.

1.2 Формализация данных

База данных должна хранить информацию о:

- компаниях;
- прогнозах;
- финансовых показателях компаний;
- пользователях;
- действиях, совершаемых пользователями;
- группах пользователей;
- правах доступа пользователей.

Таблица 1.1: Категории данных и сведения о них

Категория	Сведения
Компания	ID компании, название, логотип, описание, тикер, ID показателей
Прогноз	ID прогноза, инвест-дом, дата публикации, дата обновления, дата истечения, целевая цена, прогноз, описание прогноза, ID компании
Финансовые показатели	ID показателей, цена, капитализация, объем прибыли до вычета налогов, годовая выручка
Пользователь	ID пользователя, имя пользователя в системе, пароль, электронная почта, полное имя
Действия, совершаемые пользователями	ID записи, ID пользователя, ID группы пользователя, информация об изменениях до и после
Группа пользователей	ID записи, ID прав доступа, ID пользователя
Права доступа	ID записи, права доступа

1.3 Формализация ролей

Должно быть выделено три категории пользователей.

Таблица 1.2: Роли пользователей и предоставляемый им функционал

Роль	Функционал
Пользователь	просмотр компаний и информации о них, просмотр прогнозов на выбранную компанию
Аналитик	добавление, просмотр, редактирование, удаление компаний, информации о них, а также прогнозов на выбранную компанию
Администратор	просмотр, добавление, редактирование, удаление выбранных пользователей и прав доступа

1.4 Use-case диаграмма

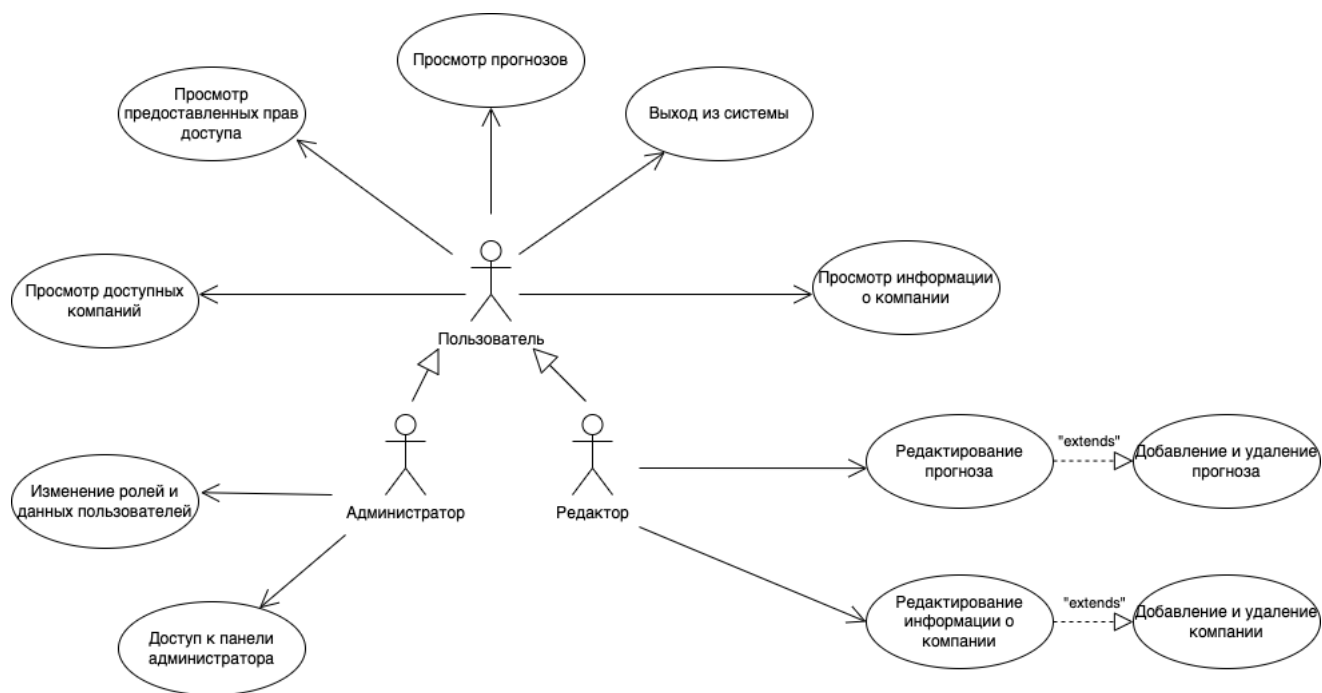


Рис. 1.1: Use-case диаграмма

1.5 ER диаграмма

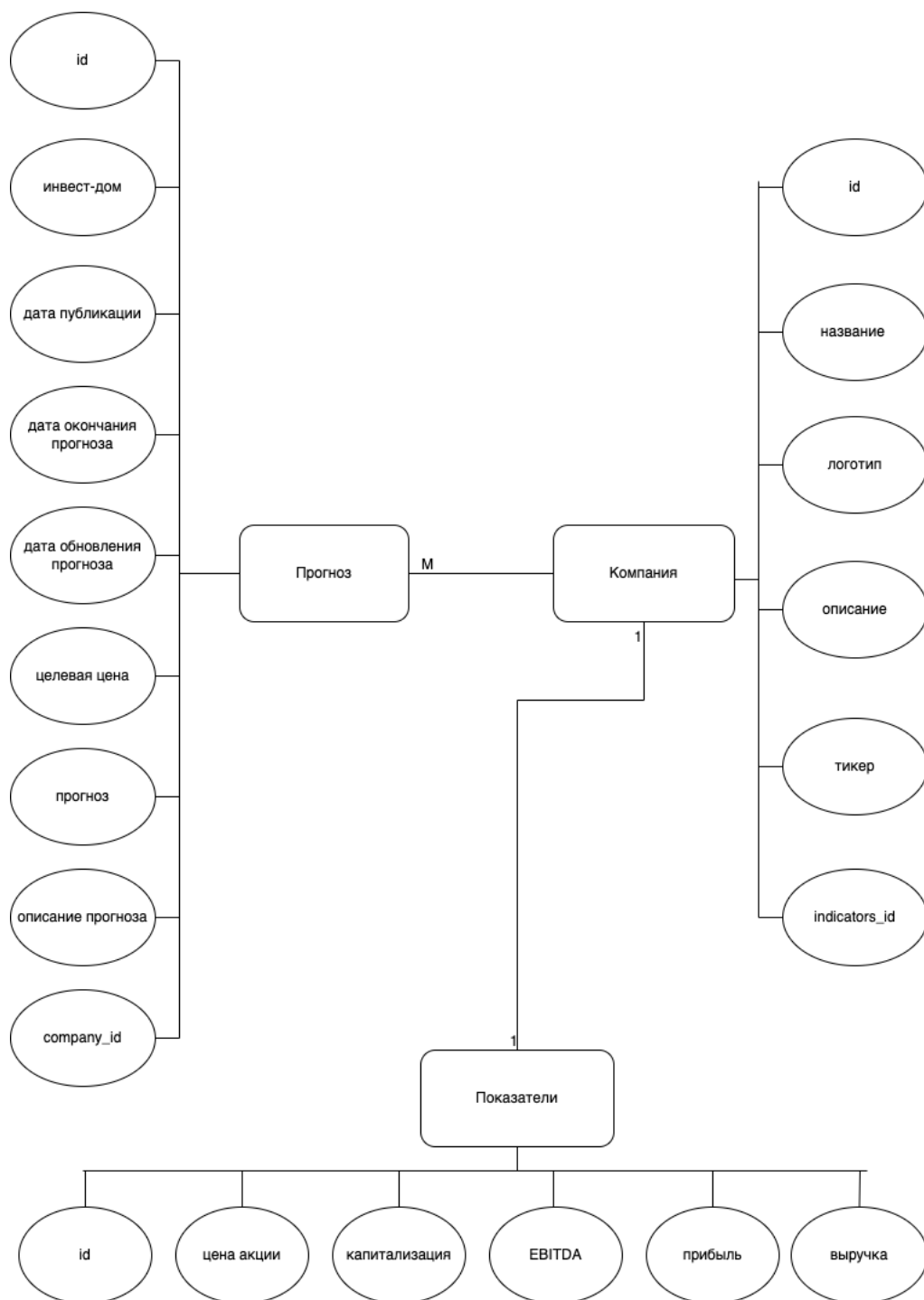


Рис. 1.2: ER диаграмма (1)

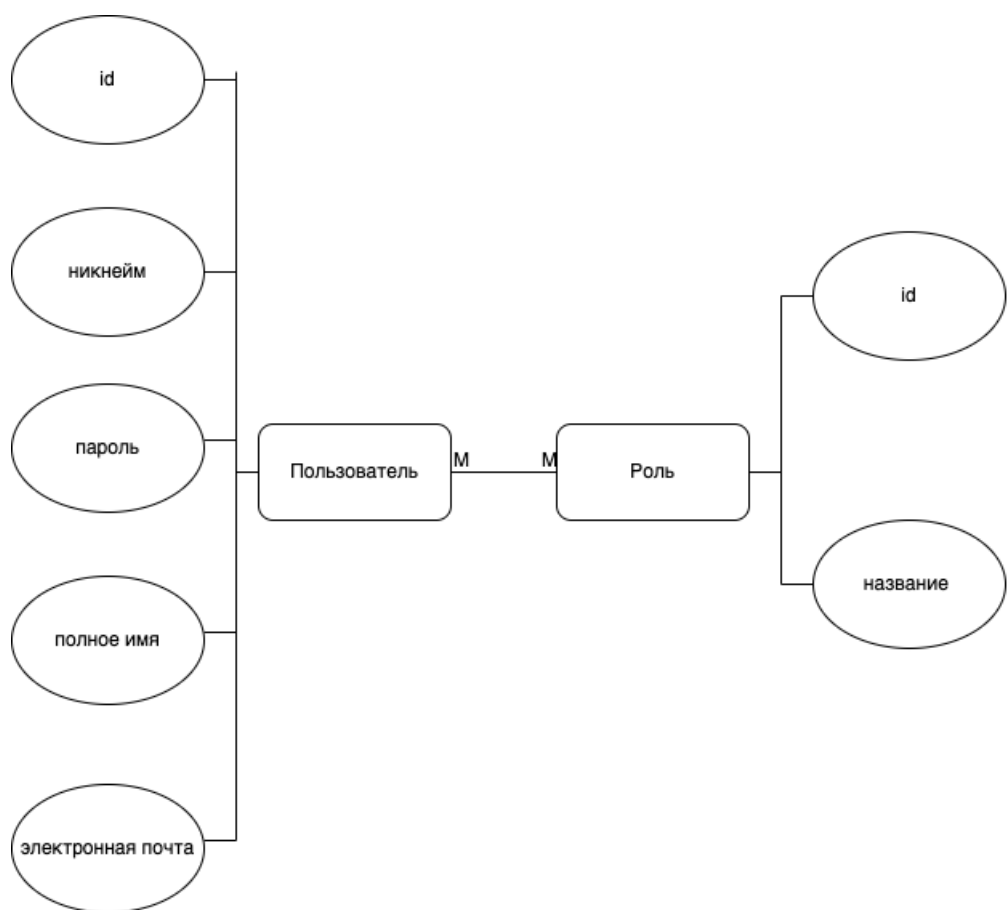


Рис. 1.3: ER диаграмма (2)

1.6 Анализ существующих решений

В качестве существующих решений для анализа выбраны сервисы Tinkoff Инвестиции, RBC Инвестиции, Investing.com, Finam. В таблице 1.3 представлен сравнительный анализ существующих решений.

Таблица 1.3: Анализ существующих решений

Сервис	Детализация	Аргументация	Субъективность	Платный доступ
Tinkoff Инвестиции	Да	Нет	Нет	Нет
RBC Инвестиции	Нет	Нет	Да	Нет
Investing.com	Да	Нет	Нет	Да
Finam	Да	Да	Да	Нет

1.7 Описание существующих баз данных и систем управления базами данных

В задаче разбора и хранения информации рабочей программы важную роль играет выбранная модель хранения данных. Для персистентного хранения данных используются базы данных. Для управления базами данных используются системы управления базами данных — СУБД. Система управления базами данных — это совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных.

1.7.1 Классификация баз данных по способу хранения

Базы данных, по способу хранения, делятся на две группы — строковые и колоночные. Каждый из этих типов служит для выполнения для определенного рода задач.

Строковые базы данных

Строковыми базами данных называются такие базы данных, записи которых в памяти представляются построчно. Строковые баз данных исполь-

зуются в транзакционных системах (англ. OLTP [3]). Для таких систем характерно большое количество коротких транзакций с операциями вставки, обновления и удаления данных - INSERT, UPDATE, DELETE.

Основной упор в системах OLTP делается на очень быструю обработку запросов, поддержание целостности данных в средах с множественным доступом и эффективность, которая измеряется количеством транзакций в секунду.

Схемой, используемой для хранения транзакционных баз данных, является модель сущностей, которая включает в себя запросы, обращающиеся к отдельным записям. Так же, в OLTP-системах есть подробные и текущие данные.

Колоночные базы данных

Колоночными базами данных называются базы данных, записи которых в памяти представляются по столбцам. Колоночные базы данных используются в аналитических системах (англ. OLAP [4]). OLAP характеризуется низким объемом транзакций, а запросы часто сложны и включают в себя агрегацию. Время отклика для таких систем является мерой эффективности.

OLAP-системы широко используются методами интеллектуального анализа данных. В таких базах есть агрегированные, исторические данные, хранящиеся в многомерных схемах.

1.7.2 Выбор модели хранения данных для решения задачи

Для решения задачи строчное хранение данных преобладает над колоночным хранением по нескольким причинам:

- задача предполагает постоянное добавление и изменение данных;
- задача предполагает быструю отзывчивость на запросы пользователя;

- задача не предполагает выполнения аналитических запросов;

1.7.3 Обзор СУБД с построчным хранением

В данном подразделе буду рассмотрены популярные построчные СУБД, которые могут быть использованы для реализации хранения в разрабатываемом программном продукте.

PostgreSQL

PostgreSQL [5] – это свободно распространяемая объектно-реляционная система управления базами данных, наиболее развитая из открытых СУБД в мире и являющаяся реальной альтернативой коммерческим базам данных [6].

PostgreSQL предоставляет транзакции со свойствами атомарности, согласованности, изоляции, долговечности (ACID [7]), автоматически обновляемые представления, материализованные представления, триггеры, внешние ключи и хранимые процедуры. Данная СУБД предназначена для обработки ряда рабочих нагрузок, от отдельных компьютеров до хранилищ данных или веб-сервисов с множеством одновременных пользователей.

Рассматриваемая СУБД управляет параллелизмом с помощью технологии управления многоверсионным параллелизмом (англ. MVCC [8]). Эта технология дает каждой транзакции «снимок» текущего состояния базы данных, позволяя вносить изменения, не затрагивая другие транзакции. Это в значительной степени устраняет необходимость в блокировках чтения (англ. read lock [9]) и гарантирует, что база данных поддерживает принципы ACID.

Oracle Database

Oracle Database [10] – объектно-реляционная система управления базами данных компании Oracle [11]. На данный момент, рассматриваемая СУБД является самой популярной в мире. [12]

Все транзакции Oracle Database соответствуют обладают свойствами

ACID, поддерживает триггеры, внешние ключи и хранимые процедуры. Данная СУБД подходит для разнообразных рабочих нагрузок и может использоваться практически в любых задачах. Особенностью Oracle Database является быстрая работа с большими массивами данных.

Oracle Database может использовать один или более методов параллелизма. Сюда входят механизмы блокировки для гарантии монопольного использования таблицы одной транзакцией, методы временных меток, которые разрешают сериализацию транзакций и планирование транзакций на основе проверки достоверности.

MySQL

MySQL [13] – свободная реляционная система управления базами данных. Разработку и поддержку MySQL осуществляет корпорация Oracle.

Рассматриваемая СУБД имеет два основных движка хранения данных: InnoDB [14] и myISAM [15]. Движок InnoDB полностью полностью совместим с принципами ACID, в отличие от движка myISAM. СУБД MySQL подходит для использования при разработке веб-приложений, что объясняется очень тесной интеграцией с популярными языками PHP [16] и Perl [17].

Реализация параллелизма в СУБД MySQL реализовано с помощью механизма блокировок, который обеспечивает одновременный доступ к данным.

1.7.4 Выбор СУБД для решения задачи

Для решения задачи была выбрана СУБД PostgreSQL, потому что данная СУБД является наиболее развитой из открытых СУБД, кроме того, PostgreSQL проста в развертывании.

Вывод

В данном разделе:

- рассмотрена структура рабочей программы дисциплины;
- проанализированы способы хранения информации для система и выбраны оптимальные способы для решения поставленной задачи;
- проведен анализ СУБД, используемых для решения задачи и также выбраны оптимальные информационные системы;
- формализованны данные, используемые в системе.

Литература

- [1] Исследование московской биржи. Режим доступа: <https://www.moex.com/n37253> (дата обращения: 07.06.2021).
- [2] What is a REST API? - Red Hat [Электронный ресурс]. Режим доступа: <https://www.redhat.com/en/topics/api/what-is-a-rest-api> (дата обращения: 07.06.2021).
- [3] What is OLTP? | IBM [Электронный ресурс]. Режим доступа: <https://www.ibm.com/cloud/learn/oltp> (дата обращения: 07.06.2021).
- [4] What is OLAP? | IBM [Электронный ресурс]. Режим доступа: <https://www.ibm.com/cloud/learn/olap> (дата обращения: 07.06.2021).
- [5] PostgreSQL: Документация. [Электронный ресурс]. Режим доступа: <https://postgrespro.ru/docs/postgresql/> (дата обращения: 07.06.2021).
- [6] PostgreSQL: вчера, сегодня, завтра [Электронный ресурс]. Режим доступа: <https://postgrespro.ru/blog/media/17768> (дата обращения: 07.06.2021).
- [7] Транзакции, ACID, CAP | GeekBrains [Электронный ресурс]. Режим доступа: https://gb.ru/posts/acid_cap_transactions (дата обращения: 07.06.2021).
- [8] Documentation: 12: 13.1. Introduction - PostgreSQL [Электронный ресурс]. Режим доступа: <https://www.postgresql.org/docs/12/mvcc-intro.html> (дата обращения: 07.06.2021).
- [9] Применение блокировок чтения/записи | IBM [Электронный ресурс]. Режим доступа: <https://www.ibm.com/docs/ru/aix/7.2?topic=programming-using-readwrite-locks> (дата обращения: 07.06.2021).
- [10] SQL Language | Oracle [Электронный ресурс]. Режим доступа: <https://www.oracle.com/database/technologies/appdev/sql.html> (дата обращения: 07.06.2021).

- [11] Oracle | Integrated Cloud Applications and Platform Services [Электронный ресурс]. Режим доступа: <https://www.oracle.com/index.html> (дата обращения: 07.06.2021).
- [12] DB-Engines Ranking [Электронный ресурс]. Режим доступа: <https://db-engines.com/en/ranking> (дата обращения: 07.06.2021).
- [13] MySQL Database Service is a fully managed database service to deploy cloud-native applications. [Электронный ресурс]. Режим доступа: <https://www.mysql.com/> (дата обращения: 07.06.2021).
- [14] MySQL Reference Manual 8.0: The InnoDB Storage Engine [Электронный ресурс]. Режим доступа: <https://dev.mysql.com/doc/refman/8.0/en/innodb-storage-engine.html> (дата обращения: 07.06.2021).
- [15] MySQL Reference Manual 16.2: The MyISAM Storage Engine [Электронный ресурс]. Режим доступа: <https://dev.mysql.com/doc/refman/8.0/en/myisam-storage-engine.html> (дата обращения: 07.06.2021).
- [16] PHP: Hypertext Preprocessor [Электронный ресурс]. Режим доступа: <https://www.php.net/> (дата обращения: 07.06.2021).
- [17] The Perl Programming Language [Электронный ресурс]. Режим доступа: <https://www.perl.org/> (дата обращения: 07.06.2021).