



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет имени  
Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ «Информатика и системы управления»

---

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

---

# РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

## *К КУРСОВОЙ РАБОТЕ НА ТЕМУ:*

*Реализация базы данных для хранения информации об  
экологических инцидентах*

---

---

---

---

Студент ИУ7-66Б  
(Группа)

\_\_\_\_\_  
(Подпись, дата)

Е.А. Мазур  
(И.О. Фамилия)

Руководитель курсовой работы

\_\_\_\_\_  
(Подпись, дата)

А.С. Кострицкий  
(И.О. Фамилия)

2022 г.

# Содержание

<b>Введение</b>	<b>3</b>
<b>1 Аналитический раздел</b>	<b>4</b>
1.1 Обзор существующих решений . . . . .	4
1.2 Формализация задачи . . . . .	4
1.3 Анализ баз данных по способу хранения . . . . .	6
1.4 Вывод . . . . .	7
<b>2 Конструкторский раздел</b>	<b>8</b>
<b>3 Технологический раздел</b>	<b>9</b>
3.1 Обзор СУБД с построчным хранением . . . . .	9
<b>Литература</b>	<b>10</b>

# Введение

Разлив дизельного топлива в Норильске[1], сибирские пожары[2], катастрофа на Камчатке[3], повлекшая за собой массовую гибель морских животных – последние годы подобные инциденты потрясают экологов и всех небезразличных жителей планеты. Чаще всего первоисточником информации о происшествиях являются местные жители, которые, обнаружив проблему, начинают звонить журналистам, писать экологическим организациям, публиковать информацию в социальных сетях[4].

Необходим единый сервис для размещения информации об экологических инцидентах. Такая система не только позволила бы оперативно оповещать специальные службы и экологическое сообщество о происшествиях, но и повысила бы экологическую осведомленность населения. Сформированная таким образом база данных экологических инцидентов могла бы использоваться исследователями для анализа.

Цель данной работы – разработать базу данных для хранения данных об экологических инцидентах.

Для достижения данной цели, необходимо решить следующие задачи:

- формализовать задачу и сформулировать требования к разрабатываемому ПО;
- проанализировать существующие СУБД и выбрать подходящую для решения задачи систему;
- спроектировать базу данных, описать ее сущности и связи;
- реализовать интерфейс доступа к базе данных;
- реализовать ПО для работы пользователей с базой данных.

# 1 Аналитический раздел

В данном разделе будут рассмотрены существующие решения, формализована решаемая задача, выбрана СУБД для ее решения.

## 1.1 Обзор существующих решений

В 2020 году WWF России представил проект, который призван помочь жителям нашей страны оперативно сообщать об авариях и инцидентах [5]. Идея была размещена в рамках форума "Сильные идеи для нового времени" [7]. С помощью специальной формы на сайте каждый сможет оперативно передать фото с места аварии, координаты инцидента и снабдить свое сообщение комментарием. По информации с места событий будут запрашиваться оперативные данные космического мониторинга. Однако на момент написания курсовой работы нет сведений о судьбе данного проекта.

WWF России и Fairy, бренд компании Procter&Gamble, запустили национальную программу общественного мониторинга аварийных экологических ситуаций. Программа позволяет своевременно отслеживать и оповещать дежурные службы об инцидентах, аналогичных катастрофам на побережье Авачинской бухты на Камчатке, разливе в Норильске или аварии на продуктопроводе на реке Оби [6]. В рамках данного проекта у пользователей есть возможность получить информацию об инцидентах в с помощью интерактивной карты. Однако пользователи не могут загружать информацию об инцидентах в систему. Также данная программа узконаправлена – данные собираются только об инцидентах в нефтегазовом секторе.

## 1.2 Формализация задачи

Под экологическим инцидентом будем понимать любое происшествие, которое привело или может привести к неблагоприятным последствиям для окружающей среды.

Разрабатываемая система должна выделять следующие типы экологических инцидентов:

- разлив нефти или нефтепродуктов;
- выброс радиоактивных веществ;
- выброс аварийно химически опасных веществ;
- выброс биологически опасных веществ;
- пожар;
- несанкционированных свалка, скопление мусора;
- другие экологические инциденты.

Каждая запись об экологическом инциденте должна содержать следующие данные:

- краткое описание;
- тип;
- координаты;
- дата;
- статус (подтвержден/не подтвержден);
- пользователь, опубликовавший инцидент.

Каждая запись об экологическом инциденте может содержать следующие данные:

- фотография;
- комментарий.

Работа пользователей с базой экологических инцидентов должна осуществляться посредством клиент-серверного веб-приложения с возможностью авторизации. Приложение должно поддерживать работу четырех типов пользователей со следующими возможностями:

- неавторизованный пользователь:
  - просмотр записей об экологических инцидентах в виде списка;
  - просмотр записей об экологических инцидентах в виде карты;
  - регистрация нового аккаунта или вход в существующий.
- авторизованный пользователь:
  - все возможности неавторизованного пользователя;
  - добавление записи об экологическом инциденте;

- удаление собственной записи об инциденте со статусом "неподтвержден";
- редактирование собственной записи об инциденте со статусом "неподтвержден";
- запрос на удаление собственной записи об инциденте со статусом "подтвержден";
- запрос удаление собственной записи об инциденте со статусом "подтвержден";
- выход из аккаунта.
- модератор:
  - все возможности авторизованного пользователя;
  - установка статуса инцидента;
  - установка для инцидента метки "Экологическая катастрофа";
  - принятие/отклонение запроса авторизованного пользователя на редактирование записи об инциденте со статусом "подтвержден";
  - принятие/отклонение запроса авторизованного пользователя на удаление записи об инциденте со статусом "подтвержден";
  - удаление записи об инциденте;
  - редактирование записи об инциденте.
- администратор:
  - все возможности модератора;
  - назначение авторизованному пользователю роли модератора;
  - снятие авторизованного пользователя с роли модератора.

## 1.3 Анализ 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 ко-

торых в памяти представляются построчно. Строковые базы данных используются в транзакционных системах (англ. OLTP [8]). 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 [9]). 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.

## **1.4 Вывод**

В данном разделе была формализована решаемая задача, проведен обзор существующих аналогов и СУБД. Для решения задачи выбран построчный способ хранения данных, так как:

- задача предполагает постоянное добавление и изменение данных;
- задача предполагает быструю отзывчивость на запросы пользователя;
- задача не предполагает выполнения аналитических запросов.

## 2 Конструкторский раздел



Рис. 2.1: UseCase диаграмма



## 3 Технологический раздел

### 3.1 Обзор СУБД с построчным хранением

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

#### PostgreSQL

PostgreSQL [10] – это свободно распространяемая объектно-реляционная система управления базами данных, наиболее развитая из открытых СУБД в мире и являющаяся реальной альтернативой коммерческим базам данных [11].

PostgreSQL предоставляет транзакции со свойствами атомарности, согласованности, изоляции, долговечности (ACID [12]), 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 [13]). 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 [14]) и гарантирует, что база данных поддерживает принципы ACID.

#### Oracle Database

Oracle Database [15] – объектно-реляционная система управления базами данных компании Oracle [16]. На данный момент, рассматриваемая СУБД является самой популярной в мире. [17]

Все транзакции Oracle Database соответствуют обладают свойствами ACID, поддерживает триггеры, внешние ключи и хранимые процедуры. Данная СУБД подходит для разнообразных рабочих нагрузок и может использо-

ваться 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 [18] – свободная реляционная система управления базами данных. Разработку и поддержку MySQL осуществляет корпорация Oracle.

Рассматриваемая СУБД имеет два основных движка хранения данных: InnoDB [19] и myISAM [20]. 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 [21] и Perl [22].

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

## **Вывод**

В результате анализа для решения задачи выбрана СУБД PostgreSQL.

# Литература

- [1] Разлив дизтоплива в Норильске стал самым масштабным на планете - РИА Новости [Электронный ресурс]. Режим доступа: <https://ria.ru/20201224/razliv-1590698352.html> (дата обращения: 13.05.2022).
- [2] В Сибири горят более 25 тысяч гектаров леса - РИА Новости [Электронный ресурс]. Режим доступа: <https://ria.ru/20200429/1570719017.html> (дата обращения: 13.05.2022).
- [3] Экологическая катастрофа на Камчатке - РИА Новости [Электронный ресурс].
- [4] эксперт Greenpeace об экологических катастрофах на Камчатке и в Норильске и их последствиях - Forbes [Электронный ресурс]. Режим доступа: <https://www.forbes.ru/forbeslife/410857-my-ostaem-sya-v-plenu-sovetskih-tehnologiy-i-okazhemsya-v-glubokom-ekonomicheskom> (дата обращения: 13.05.2022).
- [5] Предотвратить экологические катастрофы можно только при участии общественности - WWF [Электронный ресурс]. Режим доступа: <https://wwf.ru/resources/news/ekologicheskaya-politika/wwf-predotvratit-ekologicheskie-katastrofy-mozhno-tolko-pri-uchastii-obshchestvennosti/> (дата обращения: 13.05.2022).
- [6] Предотвратить экологические катастрофы можно только при участии общественности - WWF [Электронный ресурс]. Режим доступа: <https://wwf.ru/resources/news/Regulirovanie/wwf-rossii-i-brend-fairy-zapustili-obshchestvennyy-monitoring-avariynykh-ekologicheskikh-situatsiy-v/> (дата обращения: 13.05.2022).
- [7] Сильные идеи для нового времени [Электронный ресурс]. Режим доступа: <https://idea.asi.ru>. (дата обращения: 13.05.2022).
- [8] What is OLTP? | IBM [Электронный ресурс]. Режим доступа: <https://www.ibm.com/cloud/learn/oltp> (дата обращения: 07.06.2021).

- [9] What is OLAP? | IBM [Электронный ресурс]. Режим доступа: <https://www.ibm.com/cloud/learn/olap> (дата обращения: 07.06.2021).
- [10] PostgreSQL: Документация. [Электронный ресурс]. Режим доступа: <https://postgrespro.ru/docs/postgresql/> (дата обращения: 07.06.2021).
- [11] PostgreSQL: вчера, сегодня, завтра [Электронный ресурс]. Режим доступа: <https://postgrespro.ru/blog/media/17768> (дата обращения: 07.06.2021).
- [12] Транзакции, ACID, CAP | GeekBrains [Электронный ресурс]. Режим доступа: <https://gb.ru/posts/acidcaptransactions> (дата обращения: 07.06.2021).
- [13] Documentation: 12: 13.1. Introduction - PostgreSQL [Электронный ресурс]. Режим доступа: <https://www.postgresql.org/docs/12/mvcc-intro.html> (дата обращения: 07.06.2021).
- [14] 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).
- [15] SQL Language | Oracle[Электронный ресурс]. Режим доступа: <https://www.oracle.com/database/technologies/appdev/sql.html> (дата обращения: 07.06.2021).
- [16] Oracle | Integrated Cloud Applications and Platform Services [Электронный ресурс]. Режим доступа: <https://www.oracle.com/index.html> (дата обращения: 07.06.2021).
- [17] DB-Engines Ranking [Электронный ресурс]. Режим доступа: <https://db-engines.com/en/ranking> (дата обращения: 07.06.2021).
- [18] 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).
- [19] 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).

- [20] 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).
- [21] PHP: Hypertext Preprocessor [Электронный ресурс]. Режим доступа: <https://www.php.net/> (дата обращения: 07.06.2021).
- [22] The Perl Programming Language [Электронный ресурс]. Режим доступа: <https://www.perl.org/> (дата обращения: 07.06.2021).