МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждения образования «БЕЛОРУССКИЙ

ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет Информационных технологий

Кафедра Информационных систем и технологий

Специальность 1-98 01 03 Программное обеспечение информационной безопасности мобильных систем

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**к дипломному проекту:**

Система контроля и анализа эвакуации в условиях чрезвычайных ситуаций

Дипломник ­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Тышкевич Р.А.

Руководитель проекта ­ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Жиляк Н.А., к.т.н.,доцент

Заведующий кафедрой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Блинова Е. А., к.т.н., доцент

Консультант \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Соболевский А. С., ст. преп.

Нормоконтролер \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Николайчук А.Н., пред.-стажер

Дипломный проект защищен с оценкой

Председатель ГЭК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Минск 2025

**Содержание**

1. **Постановка задачи и анализ аналогичных решений**
   1. **Постановка задачи**

Цель проекта — разработка программной системы, предназначенной для контроля, анализа и информационного сопровождения эвакуационных процессов в условиях чрезвычайных ситуаций. Система должна обеспечивать визуализацию опасных зон на интерактивной карте, предоставление пользователям актуальной информации о маршрутах эвакуации, а также оперативное оповещение через Telegram-бота с рекомендациями по действиям в экстренных ситуациях.

Для достижения поставленной цели необходимо решить следующие задачи:

* Разработать веб-приложение, включающее:
  + Интерактивную карту Республики Беларусь с возможностью отображения опасных зон в реальном времени;
  + Систему визуальной маркировки территорий с различным уровнем риска (например, цветовое выделение зон);
  + Механизм динамического построения безопасных маршрутов с учётом текущей обстановки;
  + Удобный и понятный пользовательский интерфейс, ориентированный на широкий круг пользователей.
* Разработать Telegram-бота, обеспечивающего:
  + Оповещение пользователей о наступлении чрезвычайных ситуаций;
  + Автоматическую рассылку инструкций, рекомендаций и схем эвакуации;
  + Персонализированную информацию в зависимости от местоположения или привязанного объекта пользователя (например, университет, жилой район и др.);
  + Возможность интеграции с веб-приложением для получения актуальных данных об опасных зонах.
* Обеспечить архитектуру системы, которая:
  + Позволяет легко масштабировать проект и дополнять его новыми функциями;
  + Поддерживает централизованное и своевременное обновление информации;
  + Гарантирует надёжность и отказоустойчивость при передаче данных между компонентами системы.

Реализация проекта направлена на повышение уровня готовности и информированности населения в условиях чрезвычайных ситуаций, а также на поддержку эффективной и безопасной эвакуации с помощью современных информационных технологий.

* 1. **Анализ решений**
     1. **Требование к системе**

Разработчиком при создании программной системы контроля и анализа эвакуации в условиях чрезвычайных ситуаций были установлены следующие требования к функциональности компонентов:

Веб-приложение должно обеспечивать**:**

* отображение интерактивной карты территории Республики Беларусь;
* визуальное выделение опасных зон (например, при помощи цветовой маркировки: жёлтая — нежелательное пребывание, красная — закрытая зона);
* отображение текущей информации о чрезвычайных ситуациях в режиме реального времени;
* построение безопасных маршрутов с учётом опасных зон;
* отображение рекомендаций и инструкций по эвакуации в зависимости от местоположения;
* возможность обновления и расширения данных об опасных зонах через административный интерфейс.

Telegram-бот должен обеспечивать:

* регистрацию и авторизацию пользователя в системе;
* рассылку уведомлений при возникновении чрезвычайных ситуаций;
* отправку текстовых и графических инструкций по эвакуации (в том числе схем эвакуации зданий);
* персонализированные уведомления в зависимости от привязанного к пользователю объекта (например, университет, школа и т.д.);
* возможность получения справочной информации по действиям в различных ЧС.

Общие требования к системе:

* устойчивость к нагрузке при массовой отправке уведомлений;
* синхронизация данных между веб-приложением и Telegram-ботом;
* возможность масштабирования системы (добавление новых регионов, объектов, типов ЧС);
* обеспечение безопасности пользовательских данных;
* удобный и понятный интерфейс для пользователей и администраторов.
  + 1. **Выбор библиотек и технологий для серверной стороны**

Для реализации серверной части системы контроля и анализа эвакуации были выбраны современные и надёжные технологии, обеспечивающие стабильную работу, масштабируемость и удобство в разработке.

В качестве языка программирования используется Java версии 23, что позволяет применять новейшие языковые конструкции и гарантирует долгосрочную поддержку. Для управления сборкой и зависимостями выбран инструмент Gradle с использованием Groovy DSL, что обеспечивает гибкость конфигурации и простоту автоматизации.

Серверная часть выполняет функции API-сервиса и взаимодействует как с веб-приложением, так и с Telegram-ботом. Для разработки REST API применяется фреймворк Spring Boot, благодаря которому упрощается процесс настройки и ускоряется разработка, при этом обеспечивается высокая производительность приложения.

Дополнительно используются следующие технологии и библиотеки:

* Spring Web — для обработки HTTP-запросов и маршрутизации;
* Spring Security — для обеспечения авторизации и аутентификации;
* Spring Data JPA + Hibernate — для работы с базой данных;
* PostgreSQL — в качестве реляционной СУБД, хранящей информацию о пользователях, зонах риска, уведомлениях и т.д.;
* Telegram Bot Java Library (org.telegram.telegrambots) — для интеграции с Telegram и реализации функционала бота.
  + 1. **Выбор библиотек и технологий для клиентской стороны**

Клиентская сторона представлена в виде веб-приложения, основной задачей которого является отображение интерактивной карты с актуальной информацией об опасных зонах, маршрутах и рекомендациях по эвакуации.

Для реализации веб-интерфейса выбраны следующие технологии:

* HTML5 и CSS3 — для создания структуры и стилизации страниц;
* JavaScript — для реализации клиентской логики;
* React — как основной JavaScript-фреймворк для построения интерфейса с возможностью реактивного обновления компонентов;
* Leaflet.js — библиотека для работы с интерактивными картами, позволяющая отображать географические объекты и выделять зоны риска;
* Axios — для выполнения асинхронных HTTP-запросов к серверному API;
* Bootstrap (либо Tailwind CSS) — для ускоренной разработки адаптивного интерфейса.

Веб-клиент обеспечивает пользователю быстрый доступ к актуальной информации, возможность взаимодействия с картой и получение маршрутов в удобном и интуитивно понятном виде.

* + 1. **Выбор средств программирования**

В процессе реализации проекта используются современные средства программирования, обеспечивающие удобную разработку, отладку и сопровождение системы.

Основные средства разработки:

* IntelliJ IDEA — интегрированная среда разработки для Java, предоставляющая мощные инструменты для работы со Spring Boot, Gradle и другими фреймворками;
* Visual Studio Code — лёгкий и удобный редактор для разработки клиентской части с поддержкой React и JavaScript;
* PostgreSQL + pgAdmin — для управления базой данных;
* Git + GitHub — для контроля версий и совместной работы;
* Docker (опционально) — для контейнеризации и развертывания компонентов системы в разных средах;
* ngrok — для локального тестирования Telegram-бота с внешним доступом.

Выбор указанных средств обусловлен их популярностью, открытым исходным кодом, активным сообществом и богатым функционалом, что значительно упрощает процесс реализации дипломного проекта.

1. **Проектирование приложения**
   1. **Диаграмма вариантов использования**
   2. **Архитектура приложения**
   3. **Проектирование базы данных**

Перед началом разработки архитектуры приложения необходимо было спроектировать базу данных. На основе сформулированных функциональных требований была определена структура информации, подлежащей хранению, а также выявлены взаимосвязи между сущностями. Основной задачей дипломного проекта является создание приложения, которое обеспечивает учет и управление студентами и администраторами в контексте образовательного учреждения.

В соответствии с объектно-ориентированным подходом и использованием технологии ORM (Object-Relational Mapping), структура базы данных отражает иерархию сущностей: базовый абстрактный класс User, от которого наследуются классы Student и Admin. Это позволяет эффективно использовать механизмы наследования при работе с данными и минимизировать дублирование.

База данных состоит из следующих основных таблиц, описание которых представлено в таблицах 2.1–2.3.

Таблица 2.1 – Описание таблицы users

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| ID | BIGINT(PK) | Уникальный идентификатор пользователя |
| FIRST\_NAME | VARCHAR | Имя пользователя |
| LAST\_NAME | VARCHAR | Фамилия пользователя |
| PATRONYMIC | VARCHAR | Отчество пользователя |

Таблица users является базовой и содержит общие поля, характерные как для студентов, так и для администраторов. Все пользователи наследуют данные из данной таблицы.

Таблица 2.2 – Описание таблицы students

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| ID | BIGINT (PK/FK) | Ссылка на запись в таблице users |
| STUDENT\_NUMBER | CHAR(8) | Уникальный 8-значный номер студента |
| FACULTY | VARCHAR | Название факультета |
| FORM\_OF\_STUDY | VARCHAR | Форма обучения (очная, заочная и т.п.) |
| ADMISSION\_DATE | DATE | Дата поступления |
| TICKET\_ISSUE\_DATE | DATE | Дата выдачи студенческого билета |
| TICKET\_EXPIRY\_DATE | DATE | Дата окончания действия билета |

Таблица students хранит дополнительную информацию, специфичную для студентов.

Таблица 2.3 – Описание таблицы admins

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| ID | BIGINT (PK/FK) | Ссылка на запись в таблице users |
| DEPARTMENT | VARCHAR | Кафедра, за которую отвечает администратор |
| ACADEMIC\_DEGREE | VARCHAR | Ученая степень (например, «к.т.н.», «д.т.н.») |
| PRIVILEGE\_GRANTED | DATE | Дата выдачи прав администратора |
| PRIVILEGE\_EXPIRY | DATE | Дата окончания действия прав |

Таблица admins содержит информацию об администраторах, обладающих расширенными правами доступа в системе.

Таким образом, в базе данных приложения реализована структура из трех связанных таблиц, обеспечивающих хранение иерархически организованных данных. Это решение способствует повышению читаемости, расширяемости и устойчивости модели данных при последующем масштабировании.

Скрипты создания таблиц и диаграмма базы данных приведены в Приложении (будет добавлено позднее).

Следует отметить, что база данных играет ключевую роль в обеспечении целостности, надежности и безопасности хранения информации. Эффективная структура базы данных позволяет обеспечить быстрый доступ к данным, упрощает управление пользователями, а также способствует стабильной и бесперебойной работе всего программного комплекса.