2 Проектирование интернет-сервиса

2.1 Функциональное наполнение интернет-сервиса

Интернет-сервис, разрабатываемый в рамках данного дипломного проекта, в первую очередь, предназначен для создания удобной в использовании системы учета и контроля выполнения дипломного проектирования в вузе в виде веб-приложения для максимальной совместимости с устройствами.

Юз-кейс диаграмма интернет-сервиса представлена на рисунке 2.1.

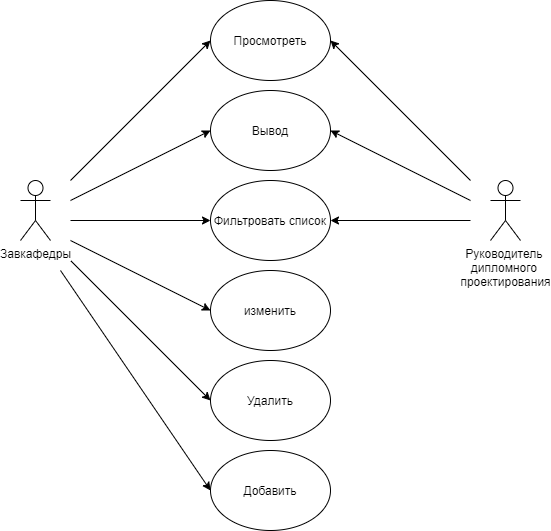


Рис.2.1 – Юз-кейс диаграмма приложения

В таблице 2.1 представлены роли интернет-сервиса и их назначение.

Таблица 2.1 — Виды ролей в приложении

|  |  |
| --- | --- |
| Роль | Назначение |
| Заведующий кафедры | Просмотр, изменение, удаление, добавление данных. Фильтрация данных. Вывод данных в документ. |
| Руководитель дипломного проектирования | Просмотр и фильтрация данных. Вывод данных в документ. |

В таблице 2.2 представлены функции доступные пользователю в интернет-сервисе.

Таблица 2.2 — Функции интернет-сервиса

|  |  |
| --- | --- |
| Функция | Пояснение |
| Просмотр | Просмотр информации в виде списка. |
| Вывод | Вывод отфильтрованных списков в виде документа |
| Фильтровать список | Изменение списка для просмотра по параметрам |
| Изменить | Изменение данных в списках. |
| Удалить | Удаление данных из списков. |
| Добавить | Добавление новых данных в списки. |

2.2 Архитектура интернет-сервиса

Целью дипломного проекта является разработка интернет-сервиса для учета и контроля выполнения дипломного проектирования в вузе для оптимизации контроля за ходом дипломных проектов или работ. Интернет-сервис должен иметь серверную часть, представляющую REST-сервисы к которым будет обращаться клиентская часть интернет-сервиса. Также должна иметься база данных для хранения данных о дипломных проектах или работ, руководителях дипломных проектов, нормконтролерах, рецензентах, председателях, приказах, комиссии и специальностях.

На рисунке 2.2 представлена архитектура интернет-сервиса данного дипломного проекта.

Клиентская часть представлена веб-приложением, которое создано для кроссбраузерной работы для учета и контроля выполнения дипломного проектирования в вузе. Клиентская часть имеет связь с серверной частью при помощи протокола http.

Серверная-часть - это часть системы, представленная REST-сервисами, предоставляющими функциональность описанную в пункте 2.1 и связующая между базой данных при помощи протокола tcp/ip и клиентской частью при помощи протокола http.

База данных – это часть интернет-сервиса ,которая предназначена для хранения информации связанной с дипломным проектированием. Данная часть должна реализовывать примерно следующий функционал: сохранение информации в формате utf-8, возможность «достать» эту информацию при помощи серверной части. Также данная часть должна поддерживать большой набор типов объектов.

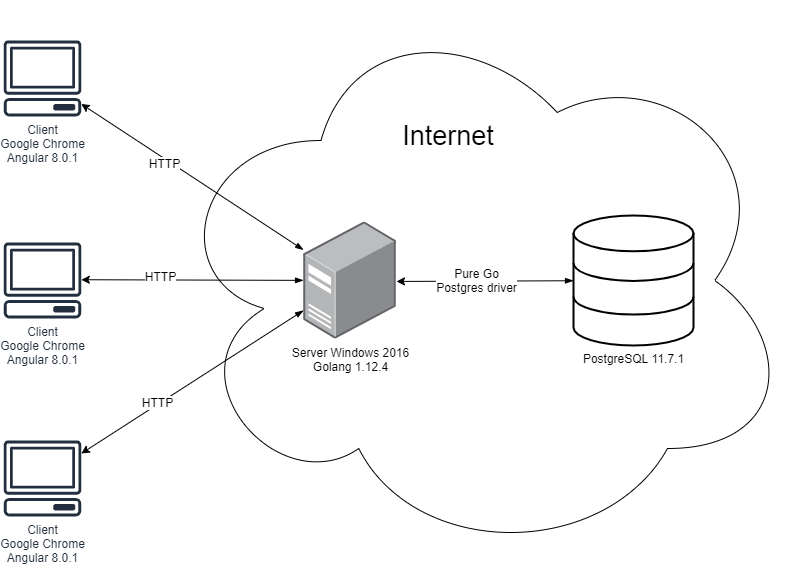


Рис.2.2 – Общая структура компонентов приложения

Основные параметры клиентской части приведены в таблице 2.3.

Таблица 2.3 — Описание клиентской части приложения

|  |  |
| --- | --- |
| Технология | Angular |
| Протокол взаимодействия | HTTP |
| Назначение | Отображение данных о дипломных проектах и работах, а так же манипуляции с ними |
| Дополнительные библиотеки | RxJs, ngx-bootstrap |

Основные параметры серверной части приведены в таблице 2.4.

Таблица 2.4 — Описание back-end части приложения

|  |  |
| --- | --- |
| Технология | Go |
| Протокол взаимодействия | HTTP, TCP/IP |
| Назначение | Связь клиентского приложения с базой данных, работа в качестве сервера статических файлов, в том числе и клиентского приложения |
| Дополнительные библиотеки | Gorilla, PQ |

Основные параметры базы данных приведены в таблице 2.5.

Таблица 2.5 — Описание базы данных приложения

|  |  |
| --- | --- |
| Технология | PostgreSQL |
| Протокол взаимодействия | TCP\IP |
| Назначение | Хранение информации о дипломном проектировании |

2.3 База данных интернет-сервиса

На рисунке 2.3 представлена схема таблиц базы данных.

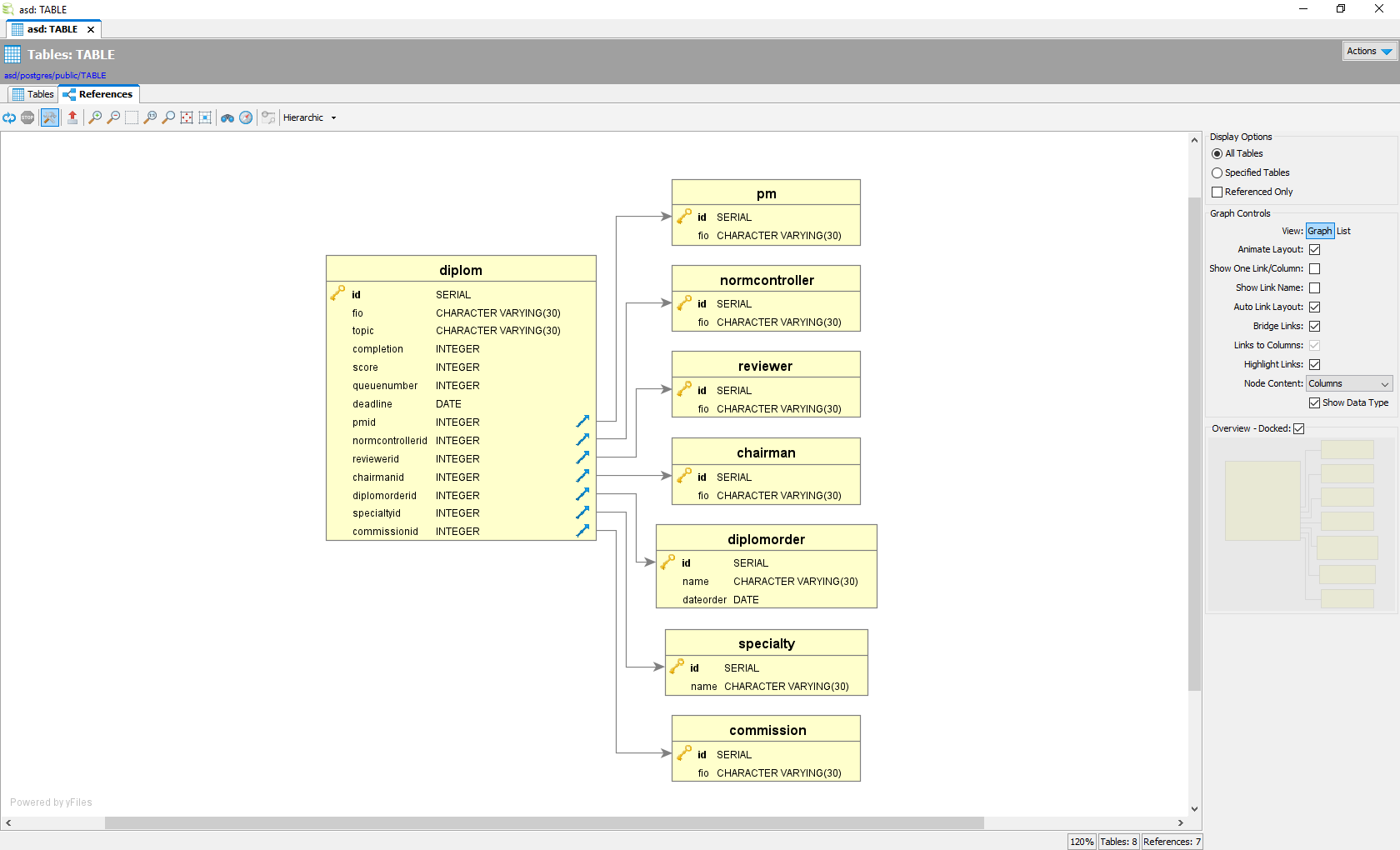


Рис.2.3 – Общая схема таблиц базы данных

Таблица «pm» предназначена для хранения руководителей дипломных проектов. Описание полей «pm» приведено в таблице 2.6.

Таблица 2.5 — Описание полей таблицы «pm»

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| id | Integer | Первичный ключ |
| fio | Varchar(30) | Имя руководителя |

Таблица «normcontroller» предназначена для хранения нормоконтролеров. Описание полей «normcontroller» приведено в таблице 2.6.

Таблица 2.6 — Описание полей таблицы «normcontroller»

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| id | Integer | Первичный ключ |
| fio | Varchar(30) | Имя нормоконтролера |

Таблица «reviewer» предназначена для хранения рецензентов. Описание полей «reviewer» приведено в таблице 2.6.

Таблица 2.6 — Описание полей таблицы «reviewer»

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| id | Integer | Первичный ключ |
| fio | Varchar(30) | Имя рецензента |

Таблица «chairman» предназначена для хранения председателей. Описание полей «chairman» приведено в таблице 2.6.

Таблица 2.6 — Описание полей таблицы «chairman»

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| id | Integer | Первичный ключ |
| fio | Varchar(30) | Имя председателя |

Таблица «diplomorder» предназначена для хранения приказов. Описание полей «diplomorder» приведено в таблице 2.7.

Таблица 2.7 — Описание полей таблицы «diplomorder»

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| id | Integer | Первичный ключ |
| name | Varchar(30) | Название приказа |
| dateorder | Date | Дата приказа |

Таблица «specialty» предназначена для хранения специальностей. Описание полей «specialty» приведено в таблице 2.8.

Таблица 2.8 — Описание полей таблицы «specialty»

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| id | Integer | Первичный ключ |
| name | Varchar(30) | Название специальности |

Таблица «commission» предназначена для хранения комиссии. Описание полей «commission» приведено в таблице 2.9.

Таблица 2.9 — Описание полей таблицы «commission»

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| id | Integer | Первичный ключ |
| name | Varchar(30) | Имя комиссии |

Таблица «diplom» предназначена для хранения данных о дипломном проекте или работе. Описание полей «diplom» приведено в таблице 2.10.

Таблица 2.10 — Описание полей таблицы «diplom»

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| id | Integer | Первичный ключ |
| fio | Varchar(30) | Фамилия студента |

Продолжение таблицы 2.10

|  |  |  |
| --- | --- | --- |
| topic | Varchar(100) | Тема диплома |
| completion | Integer | Процент выполнения |
| score | Integer | Оценка |
| deadline | Date | Дата сдачи |
| queuenumber | Integer | Номер в очереди |
| pmid | Integer | Вторичный ключ  Руководитель проекта |
| normcontrollerid | Integer | Вторичный ключ  Нормокотролер |
| reviewerid | Integer | Вторичный ключ  Рецензент |
| chairmanid | Integer | Вторичный ключ  Председатель |
| diplomorderid | Integer | Вторичный ключ  Приказ |
| specialtyid | Varchar(30) | Вторичный ключ  Специальность |
| execution | Date | Срок |
| type | Integer | Имя комиссии |
| commissioncomment | Varchar(100) | Имя комиссии |

2.3 Выводы по разделу 2

Во время выполнения раздела 2 был сформирован основной функционал, соответствующий целям дипломного проекта, для этого была спроектирована юз-кейс диаграмма, описаны роли и соответствующие им функции.

Спроектирована архитектура интернет-сервиса, описаны основные компоненты которые позволяют реализовать весь функционал описанный ранее, их технологии и назначение.

Спроектирована база данных, предназначенная для хранения данных, описаны таблицы, поля этих таблиц, и связи между таблицами.

В результате выполнения раздела 2 были сформированы необходимые данные которые позволяют перейти к разработке интернет-сервиса.

3. Разработка интернет-сервиса

Архитектура интернет-сервиса представлена в главе 2 и включает 3 основных компонента: клиентская часть интернет-сервиса, серверная часть интернет-сервиса и база данных.

3.1 Разработка клиентской части интернет-сервиса

Клиентская часть представляет из себя веб-приложение написанное на TypeScript. Для ускорения разработки был выбран фреймворк Angular. Angular - фреймворк от компании Google для создания клиентских приложений. Одной из ключевых особенностей Angular является то, что он использует в качестве языка программирования TypeScript.

Основные структурные компоненты приведены в таблице 3.1.

Таблица 3.1 — Описание структурных компонентов клиентской части

|  |  |
| --- | --- |
| Компонент | Функция |
| Папка node\_modules | Папка с npm модулями |
| Папка src | Папка с исходным кодом приложения |
| Папка public | Папка с html файлом, к которому подключаются бандлы и остальная метаинформация |
| Папка app | Папка с компонентами |
| Папка assets | Папка с картинками |
| Файл main.ts | Инициализационный файл всего приложения |
| Файл Angular.json | Файл с настройками проекта |
| Файл package.json | Файл с информацией о проекте |

Структура веб-приложения представлена на рисунке 3.1

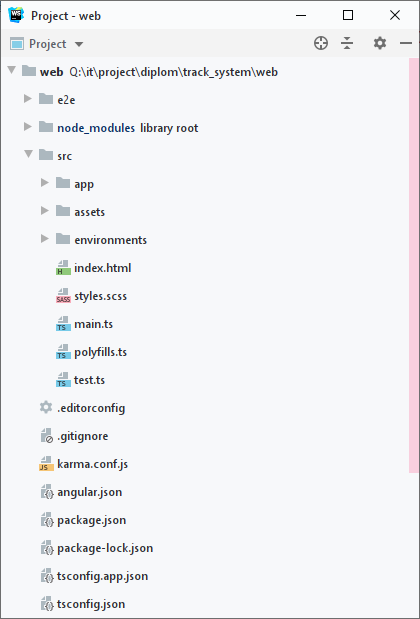


Рисунок 3.1 – Структура клиентской части приложения

Папка node\_modules хранит в себе необходимые модули для приложения, один из которых это сам Angular. Папка src содержит корневой компонент приложения, в который так или иначе подключаются другие компоненты. Папка assets содержит и прочие общие ассеты. В папке app находятся компоненты – маленькие элементы так называемого «конструктора», из которых позже собираются страницы. Файл main.ts содержит код инициализации приложения.

Остальные файлы являются конфигурационными и предназначены для разработки и развертывания, а также для написания правил проверки валидности и семантики кода.

Основные компоненты приведены в таблице 3.2.

Таблица 3.2 — Описание основных компонентов клиентской части

|  |  |
| --- | --- |
| Компонент | Функция |
| Diplom-layout | Компонент связывающий все компоненты, и пристраивающий роутинг |
| Diplom | Компонент содержащий в себе логику и отображение связанную с дипломных проектом или работой |

Продолжение таблицы 3.2

|  |  |
| --- | --- |
| diplom-chairman | Компонент содержащий в себе логику и отображение связанную с председателем |
| diplom-commission | Компонент содержащий в себе логику и отображение связанную с комиссией |
| diplom-normocontroller | Компонент содержащий в себе логику и отображение связанную с нормконтролером |
| diplom-order | Компонент содержащий в себе логику и отображение связанную с приказом |
| diplom-pm | Компонент содержащий в себе логику и отображение связанную с руководителем проекта |
| diplom-reviewer | Компонент содержащий в себе логику и отображение связанную с рецензентом |

Исходный код компонентов представлен в приложении А.

3.2 Разработка серверной части интернет-сервиса

Для реализации серверной части был выбран Go. Одной из ключевых возможностей языка Go является возможность работы с сетевыми сервисами: отправлять запросы к ресурсам в сети и, наоборот, обрабатывать входящие запросы. Основной функционал по работе с сетью представлен пакетом net. Этот пакет предоставляет различные низкоуровневые сетевые примитивы, через которые идет взаимодействие по сети.

Для упрощения маршрутизации был выбран Gorilla. Gorilla - пакет разработчика созданный для упрощения создания веб-приложений на языке Go, который, в свою очередь, включает ряд пакетов, мы будем использовать gorilla/mux который позволяет определять более сложные маршруты, которые могут использовать регулярные выражения.

Структура приложения представлена на рисунке 3.2.

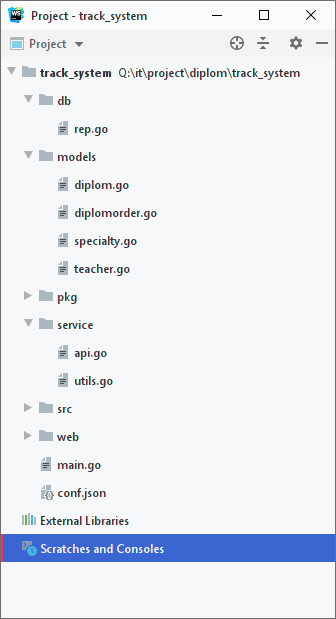


Рисунок 3.2 – Структура серверной части интернет-сервиса

Основные компоненты приведены в таблице 3.3.

Таблица 3.3 — Описание основных компонентов серверной части

|  |  |
| --- | --- |
| Компонент | Функция |
| Rep.go | Компонент описывающий работу с базой данных |
| Папка models | Содержит в себе файлы с описанием основных моделей интернет-сервиса |
| Api.go | Роутер для разделения обработки логики согласно маршруту |
| Main.go | Стартовая точка приложения |

Исходный код компонентов представлен в приложении Б.

3.3 Разработка базы данных интернет-сервиса

База данных была сформирована в разделе 2.3 и для ее реализации был создан скрипт, представленный на рисунке 3.3

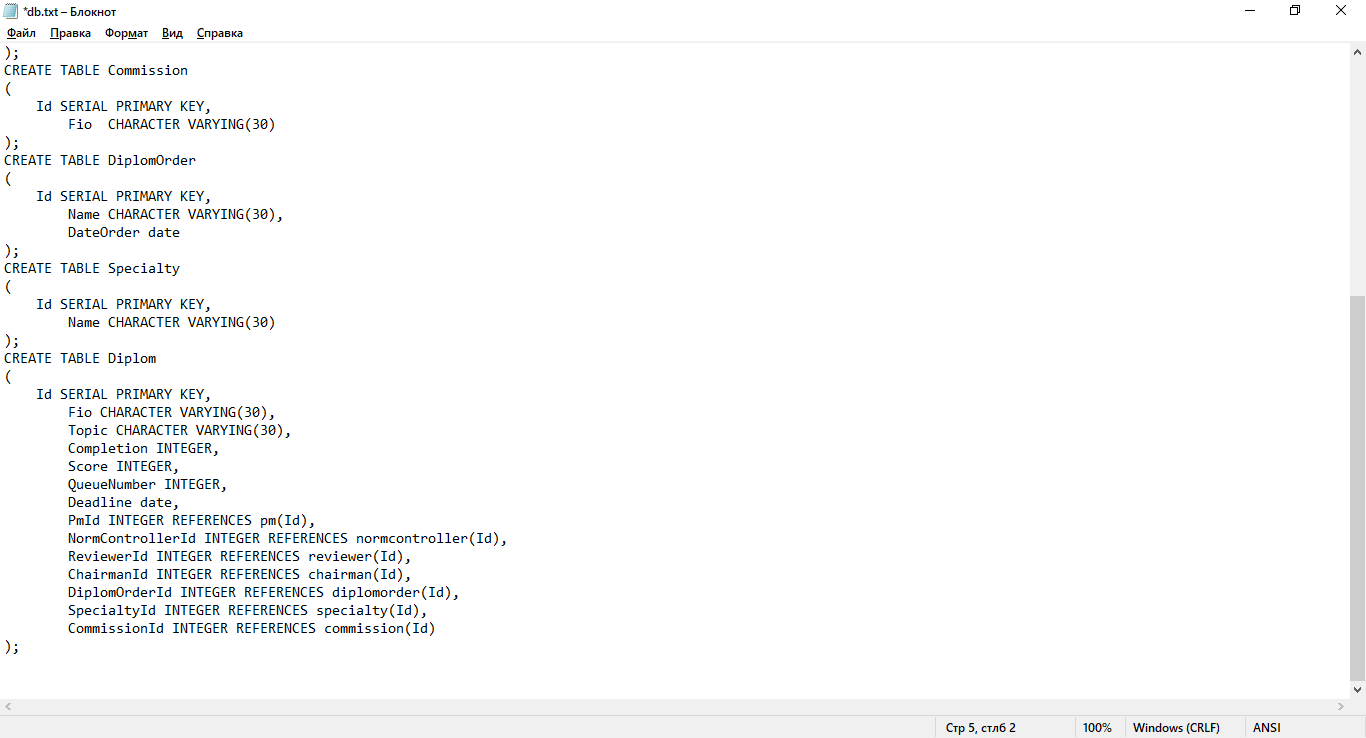


Рисунок 3.3 – Скрипт создания таблиц базы данных