|  |  |  |
| --- | --- | --- |
|  |  | УТВЕРЖДАЮ |
|  |  | Руководитель  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.  М.П. |

ТЕХНИЧЕСКИЙ ПРОЕКТ

«Telegram-бот шведский стол»

2024

Содержание

[1. Уточнение структур данных 2](#_Toc182593511)

[2. Формы для их представления 4](#_Toc182593512)

[3. Разработка алгоритма решения задачи 13](#_Toc182593513)

[4. Определения языка 18](#_Toc182593514)

[5. Структура программы и требования к техническим средствам 19](#_Toc182593515)

# Уточнение структур данных

В «Telegram-бот шведский стол» будет иметься база данных. База данных в виде логической и физической ER диаграмме изображены на рисунках 1 и 2.

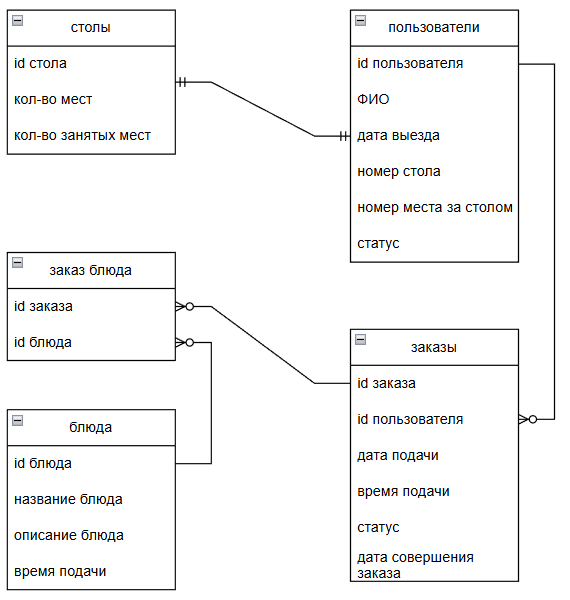


Рисунок 1 – Логическая ER диаграмма базы данных.

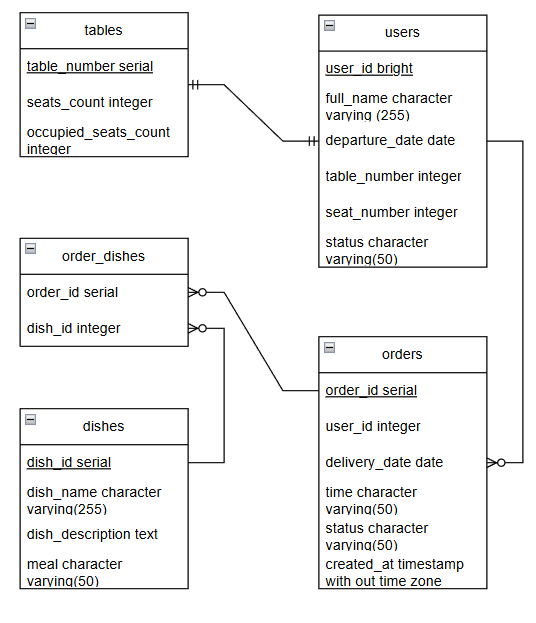


Рисунок 2 – Физическая ER диаграмма базы данных.

На ER диаграммах изображено 5 таблиц: Пользователи, заказы, блюда, заказ блюда, столы.

В таблице «пользователи» - содержатся данные о всех пользователях, которые есть в telegram-боте «шведский стол». Таблица содержит следующие данные:

* id пользователя;
* ФИО;
* Дата выезда;
* Номер стола;
* Номер места за столом;
* Статус.

В таблице «столы» - содержатся данные о всех столах с количеством общих и занятых мест. Таблица содержит следующие данные:

* Id стола;
* Кол-во мест;
* Кол-во занятых мест.

В таблице «заказы» - содержатся данные о всех заказах блюд от пользователей. Таблица содержит следующие данные:

* id заказа;
* id пользователя;
* Дата подачи;
* Время подачи;
* Статус;
* Дата совершения заказа.

В таблице «блюда» - содержатся данные о всех блюдах с описанием и времени, на которое подается блюдо. Таблица содержит следующие данные:

* id блюда;
* Название блюда;
* Описание блюда;
* Время подачи.

В таблице «заказ блюда» - содержатся данные о заказе с закрепленным к нему блюдом. Таблица содержит следующие данные:

* id заказа;
* id блюда.

# Формы для их представления

Прототип экранной формы «Регистрация» представлен на рисунке 3.

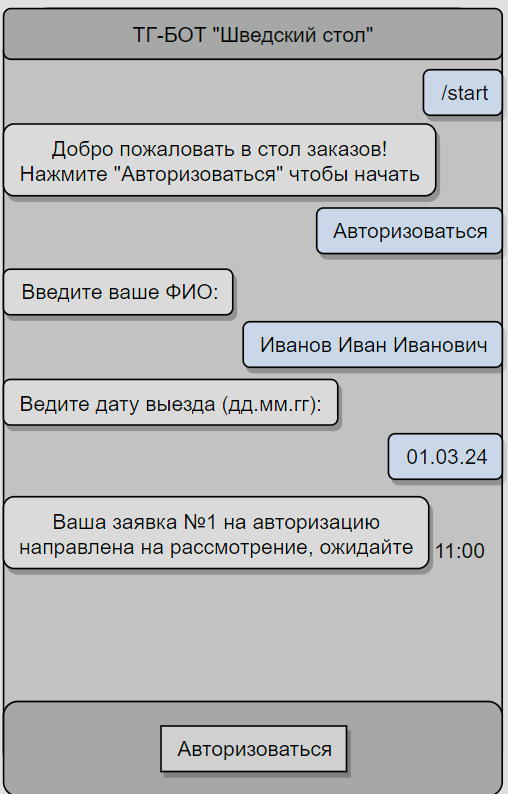


Рисунок 3 – Прототип экранной формы «Регистрация»

На прототипе экранной формы представлена первоначальная регистрация пользователя в системе. Пользователь вводит специальную команду, чтобы запустить бота, затем появляется приветственное окно, в котором написано, что нужно нажать для начала регистрации. После нажатия на кнопку «Авторизоваться» пользователю нужно ввести свои ФИО, а также дату выезда, затем заявка с введенные пользователем данными отправляется на обработку в админ-панель.

Прототип экранной формы «Выбор действия» представлен на рисунке 4.

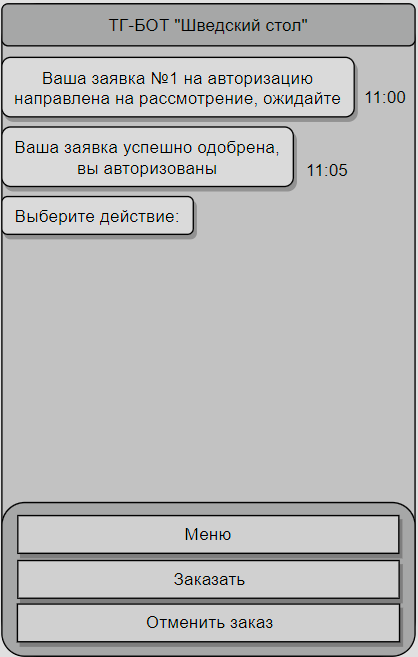


Рисунок 4 – Прототип экранной формы «Выбор действия»

После того, как администратор подтвердил регистрацию пользователя, ему приходит уведомление о одобрении заявки. Пользователю становится доступен весь функционал: просмотр меню, возможность сделать заказ или отменить уже созданный им ранее заказ.

Прототип экранной формы кнопки «Меню» представлен на рисунке 5.

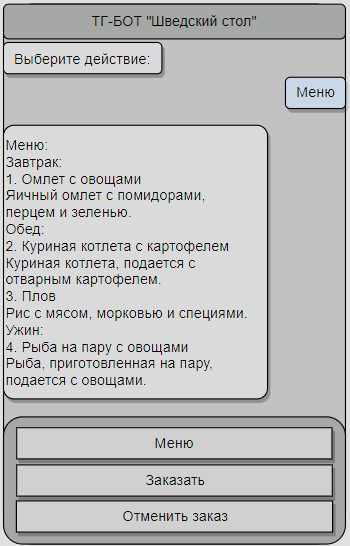


Рисунок 5 – Прототип экранной формы кнопки «Меню»

При нажатии на кнопку «Меню» пользователю высвечивается сообщение с ассортиментом. Он делится на завтрак, обед и ужин.

Прототип экранной формы кнопки «Заказать» представлен на рисунке 6.

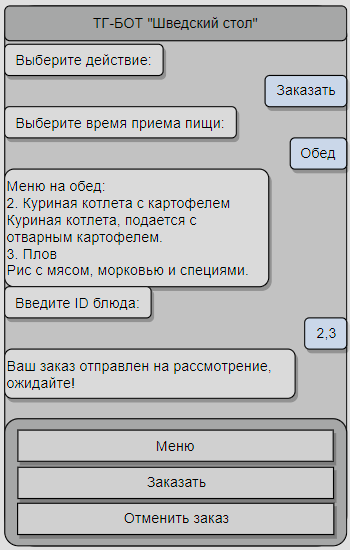


Рисунок 6 – Прототип экранной формы «Заказать»

На прототипе представлен ход действий для оформления заказа. Пользователь выбирает в какое время он хочет увидеть свой будущий заказ на столе, затем выбирает какие блюда он желает. Для успешного оформления заказа пользователю нужно выписать номер блюда. Если блюд несколько, нужно написать их номера через запятую. После всех выполненных действий заказ автоматически отправляется на обработку администратору.

Прототип экранной формы кнопки «Отменить заказ» представлен на рисунке 7.

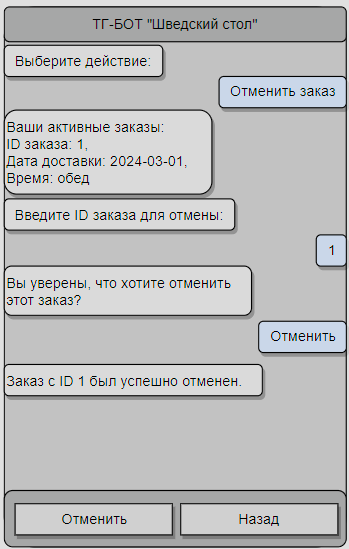


Рисунок 7 – Прототип экранной формы «Отменить заказ»

Для того, чтобы пользователь мог отменить ранее созданный им заказ, ему следует нажать на кнопку «Отменить заказ», после чего ему вылезет сообщение с информацией его заказа. Для того, чтобы его удалить, нужно ввести номер заказа в чат, после чего нажать на кнопку «Отменить» для полной отмены заказа. Следует учесть, что заказ можно удалить в течение часа после его создания. Далее, сделать это будет невозможно.

Прототип экранной формы входа в админ-панель представлен на рисунке 8.

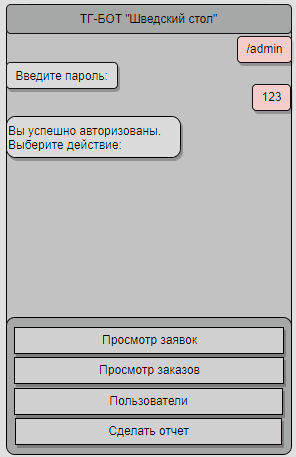


Рисунок 8 – Прототип экранной формы входа в админ-панель

Для того, чтобы войти в панель администратора, следует в этот же чат ввести специальную команду и ввести пароль. После успешного входа становятся доступны следующие кнопки: просмотр заявок, просмотр заказов, пользователи, сделать отчёт.

Прототип экранной формы кнопки «Просмотр заявок» представлен на рисунке 9.

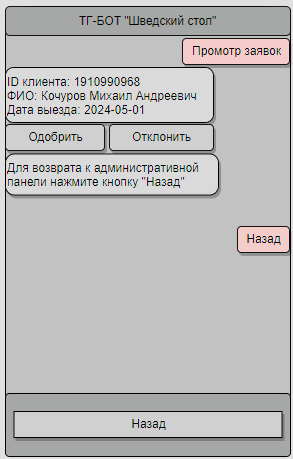


Рисунок 9 – Прототип экранной формы кнопки «Просмотр заказов»

При нажатии на кнопку просмотра заявок, вылазят непринятые на регистрацию заявки, в них пишется ID клиента, его ФИО и дата выезда. После просмотра введенной информации администратор может как одобрить заявку, так и отклонить. Для этих действий созданы две отдельные кнопки.

Прототип экранной кнопки «Просмотр заявок» при нажатии на «Одобрить» представлен на рисунке 10.

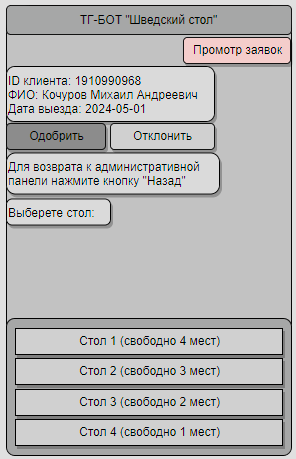


Рисунок 10 – Прототип экранной формы одобрения заявки

После нажатия на кнопку «Одобрить» администратору вылазит список свободных мест за столами, он сам присваивает место клиенту, либо обговаривает с ним вживую. После выполнения всех действий администратору вылазит сообщении об успешной регистрации нового клиента.

Прототип экранной формы кнопки «Просмотр заказов» представлен на рисунке 11.

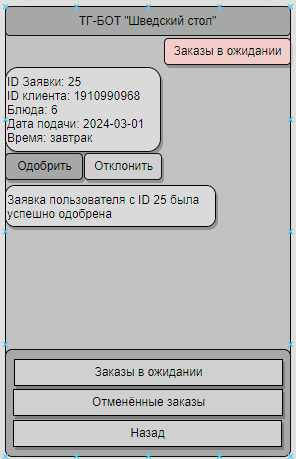


Рисунок 11 – Прототип экранной формы кнопки «Просмотр заказов»

При нажатии на кнопку просмотр заказов администратору высвечиваются все заказы, которые были сделаны клиентами. В них пишется вся информация о клиентах, блюдах и времени подачи. У администратора есть две кнопки: «Одобрить» или «Отклонить» конкретный заказ. После одобрения заказа его статус в БД становится активный.

Прототип экранной формы «Отмена заказов» представлен на рисунке 12.

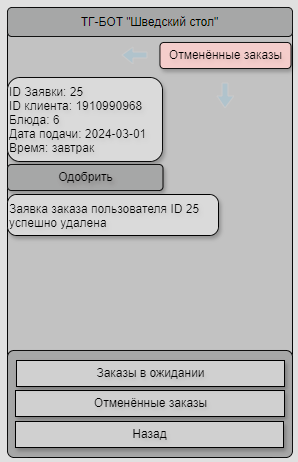


Рисунок 12 – Прототип экранной формы кнопки «Отмена заказов»

После того, как клиент отказался от уже созданного им заказа, такой заказ попадает в админ панель в отменённые заказы. Администратор может просмотреть клиента, который сделал заказ и сам заказ. В этой функции имеется только одна кнопка- «Одобрить», которая безвозвратно удалит заказ из БД.

Прототип экранной формы кнопки «Пользователи» представлен на рисунке 13.

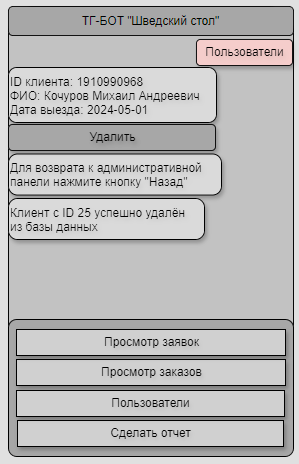


Рисунок 13 – Прототип экранной формы кнопки «Пользователи»

В этом разделе имеется возможность просмотреть всех пользователей, которые регистрировались. Выводится полная информация о клиенте, а также его дата выезда. В этом окне имеется только одна кнопка, которая позволяет администратору удалить пользователя из системы по какой-либо причине.

Прототип экранной формы «Отчёт» представлен на рисунке 14.

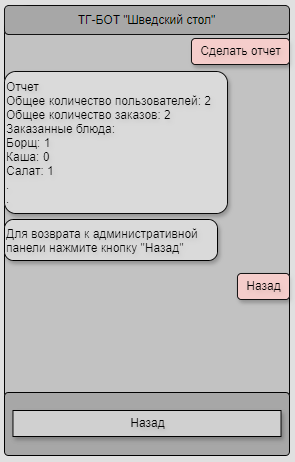


Рисунок 14 – Прототип экранной формы «Отчёт»

Данный раздел предназначен исключительно для кухни. После нажатия на кнопку о формировании отчёта открывается Excel файл, в котором показано какие позиции должны быть поданы на определенное место к определенному рациону.

# Разработка алгоритма решения задачи

Алгоритм: **Авторизация пользователя.**

Схема алгоритма «Авторизация пользователя» представлена на рисунке 15.

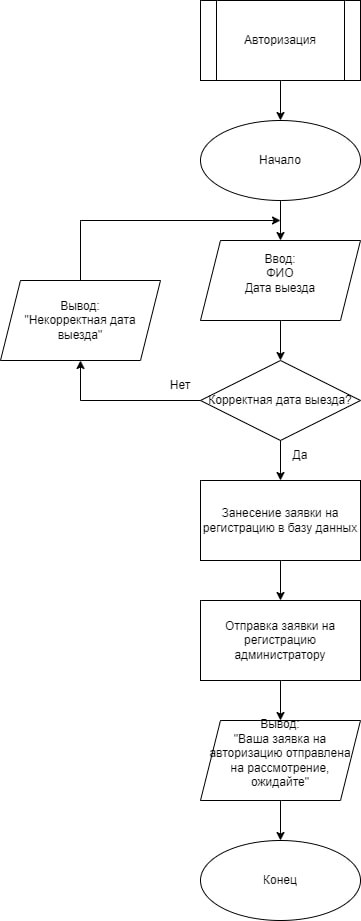


Рисунок 15 - Схема алгоритма «Авторизация пользователя»

Алгоритм: **Создание заказа.**

Схема алгоритма «Создание заказа» представлена на рисунке 16.

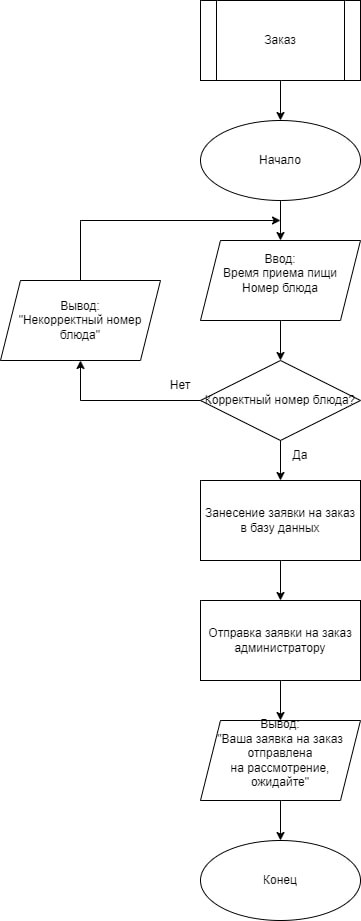


Рисунок 16 - Схема алгоритма «Создание заказа»

Алгоритм: **Отмена заказа.**

Схема алгоритма «Отмена заказа» представлена на рисунке 17.

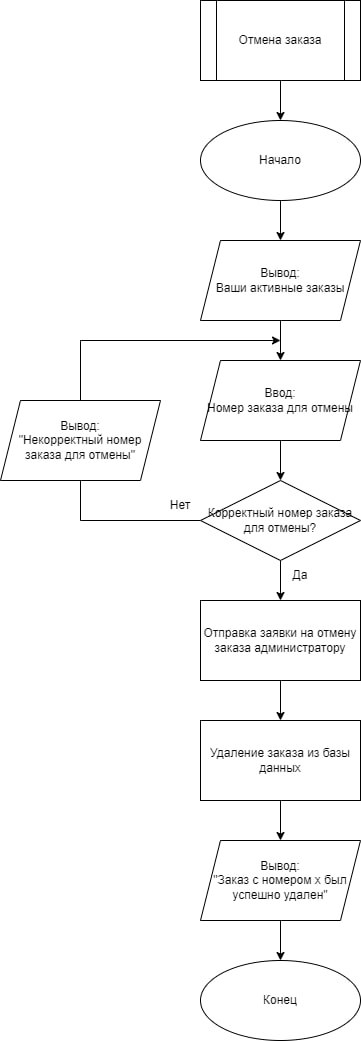


Рисунок 17 - Схема алгоритма «Отмена заказа»

Алгоритм: **Авторизация в админ-панель.**

Схема алгоритма «Авторизация в админ-панель» представлена на рисунке 18.

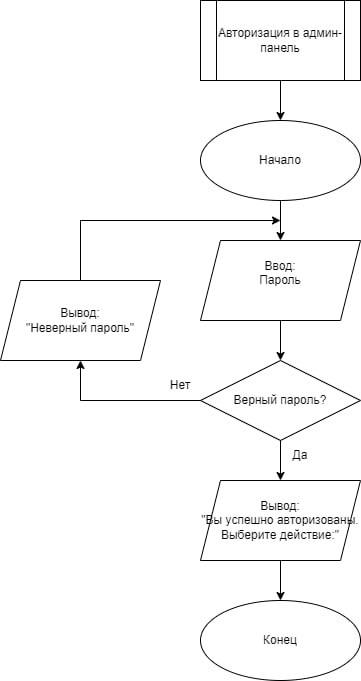


Рисунок 18 - Схема алгоритма «Завершение работы по заявке»

Так же для понимания как реагирует система был составлен пользовательский сценарий, который указан в таблице A.

Таблица A – Пользовательские сценарий

| **№ функции** | **Пользователь** | **Система** |
| --- | --- | --- |
| 1 | Регистрация | 1. Пользователь вводит команду для начала регистрации. 2. Вводит ФИО и дату выезда. 3. Заявка отправляется в админ-панель на обработку. |
| 2 | Просмотр меню | 1. Пользователь нажимает кнопку «Меню». 2. Система отображает доступное меню, разделенное на завтрак, обед и ужин. |
| 3 | Создание заказа | 1. Пользователь выбирает блюда и время подачи заказа. 2. Вводит номера выбранных блюд. 3. Заказ отправляется на обработку администратору. |
| 4 | Отмена заказа | 1. Пользователь нажимает кнопку «Отменить заказ». 2. Система отображает информацию о текущем заказе. 3. Пользователь вводит номер заказа для отмены. |
| 5 | Авторизация в админ-панель | 1. Ввод специальной команды и пароля для входа. 2. Система предоставляет доступ к функциям: просмотр заявок, заказов, пользователей и создание отчетов. |
| 6 | Одобрение заявки | 1. Администратор просматривает заявку с ID пользователя и его данными. 2. Нажимает кнопку «Одобрить». 3. Система назначает пользователю место за столом. |
| 7 | Просмотр заказов | 1. Администратор просматривает заказы с информацией о клиентах и блюдах. 2. Администратор может одобрить или отклонить заказ. |

# Определения языка

Для реализации проекта "Telegram-бот шведский стол", будут использоваться следующие языки программирования и технологии:

* Python – мультипарадигмальный высокоуровневый язык программирования общего назначения с динамической строгой типизацией и автоматическим управлением памятью, ориентированный на повышение производительности разработчика, читаемости кода и его качества, а также на обеспечение переносимости написанных на нём программ.
* PostgreSQL – свободная объектно-реляционная система управления базами данных (СУБД). Существует в реализациях для множества UNIX-подобных платформ, включая различные BSD-системы, HP-UX, IRIX, Linux, macOS, Solaris/OpenSolaris, Tru64, QNX, а также для Microsoft Windows.
* Библиотеки и фреймворки. В проекте будут использоваться специализированные библиотеки и фреймворки, которые обеспечат создание функционала подключения telegram-бота и отчетов. В том числе: xlwt для создания отчета. Telebot для реализации telegram-бота. Psycopg2 для работы с базой данных postgresql. Datetime для работы с временем.

# Структура программы и требования к техническим средствам

Функциональные требования:

* Авторизация пользователей;
* Просмотр меню;
* Заказ блюд;
* Отмена заказа блюд;
* Просмотр заявок;
* Одобрение заявок;
* Отклонение заявок;
* Удаление пользователей;
* Назначение столов.

Нефункциональные требования:

* Надежность: бот должен быть стабильным и доступным большую часть времени.