**ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

на разработку

\_\_\_\_\_\_\_\_\_\_\_\_\_\_Информационной системы для шведского стола\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2024

Содержание

[Введение 2](#_Toc182321246)

[1 Перечень сокращений 3](#_Toc182321247)

[2 Термины и определения 4](#_Toc182321248)

[3 Основные сведения о разработке 5](#_Toc182321249)

[3.1 Наименование программы 5](#_Toc182321250)

[3.2 Цель и задачи разработки 5](#_Toc182321251)

[3.3 Сведения об исполнителе 5](#_Toc182321252)

[3.4 Сведения о заказчике 5](#_Toc182321253)

[3.5 Сроки разработки 5](#_Toc182321254)

[3.6 Назначение разработки 6](#_Toc182321255)

[4 Требования к программе 9](#_Toc182321256)

[4.1 Требования к пользовательскому интерфейсу 9](#_Toc182321257)

[4.2 Требования к функциональным характеристикам 16](#_Toc182321258)

[4.3 Требования к показателям назначения 16](#_Toc182321259)

[4.4 Требования к видам обеспечения 16](#_Toc182321260)

[4.4.1 Требования к математическому обеспечению 16](#_Toc182321261)

[4.4.2 Требования к информационному обеспечению 17](#_Toc182321262)

[4.4.3 Требования к метрологическому обеспечению 17](#_Toc182321263)

[4.4.4 Требования к техническому обеспечению 17](#_Toc182321264)

[4.5 Требования к надежности 17](#_Toc182321265)

[4.6 Требования к безопасности 18](#_Toc182321266)

[4.7 Требования к патентной чистоте 18](#_Toc182321267)

[4.8 Требования к перспективам развития 18](#_Toc182321268)

[5 Состав и содержания работ 19](#_Toc182321269)

[6 Порядок разработки 20](#_Toc182321270)

[6.1 Стадии разработки 20](#_Toc182321271)

[6.2 Этапы разработки 20](#_Toc182321272)

[7 Требования к документированию 22](#_Toc182321273)

[8 Порядок контроля и приёмки 23](#_Toc182321274)

[8.1 Виды испытаний 23](#_Toc182321275)

# Введение

Данный документ фиксирует требования к проекту, его назначению и техническим характеристикам, регламентирует перечень необходимых стадий создания программного продукта и его документирования, а также специальные требования.

Документ предназначен для технических специалистов, задействованных в процессе разработки, а также для участников приемо-сдаточных испытаний, в том числе представителей Заказчика.

Документ необходим для ознакомления с техническими требованиями и целевыми свойствами разрабатываемой программы, формирует представление об ожидаемом результате и обеспечивает формирование критериев оценки полученного результата.

# Перечень сокращений

ПК – персональный компьютер.

IDE – Integrated Development Environment, единая среда разработки.

ЭВМ – электронная вычислительная машина.

ПЭВМ – Персональная электронная вычислительная машина.

API – Application Programming Interface, программный интерфейс приложения.

ГОСТ – Государственный стандарт.

ИС – информационная система.

БД – база данных.

СУБД – система управления базами данных.

# Термины и определения

Интерфейс – граница между двумя функциональными объектами, требования к которой определяются стандартом.

Информационная система – система, предназначенная для хранения, поиска и обработки информации.

Данные – сведения, показатели, характеризующие кого-либо или что-либо, служащие для каких-либо выводов, решений.

База данных – систематический сбор данных, хранящихся в электронном виде.

СУБД – система управления базами данных.

PostgreSQL – реляционная база данных.

ID – аббревиатура от английского слова Identifier, которое переводится как идентификатор.

# Основные сведения о разработке

## Наименование программы

В рамках проектирования по учебной практике предлагается разработать «Telegram-бот для шведского стола».

## Цель и задачи разработки

Цель работы заключается в создании эффективного и интуитивно понятного Telegram-бота для организации и управления шведским столом. Проект требует разработки бота, способного обеспечивать удобный процесс заказа еды из предложенного меню, а также предоставлять административные функции для управления бронированием столов и учетом пользователей.

Задачи разработки:

проанализировать предметную область;

выбор СУБД;

проектирование БД;

реализовать систему заказов и их просмотр через панель администратора;

реализовать админ-панель;

разработать список позиций для заказа;

спроектировать систему отчётов.

## Сведения об исполнителе

Исполнителем является студент Колледжа ФГБОУ ВО «Вятский государственный университет» группы ИСПк-404-52-00 Бакулин Семён Александрович

Исполнителем является студент Колледжа ФГБОУ ВО «Вятский государственный университет» группы ИСПк-404-52-00 Беляев Данил Игоревич

## Сведения о заказчике

Заказчиком является коллектив преподавателей колледжа ВятГУ в составе:

* преподаватель по УП.03 Долженкова Мария Львовна;

## Сроки разработки

Сроки исполнения работ:

Начало разработки – ???

Окончание разработки – ???

## Назначение разработки

Назначением разработки ИС «Telegram-бот для шведского стола» является создание Telegram-бота, который обеспечивает удобный и эффективный процесс заказа еды из определенного меню и управления бронированием столов для пользователей, а также административные функции для администратора.

Проблемное поле данного проекта заключается в создании эффективного и интуитивно понятного Telegram-бота для организации и управления шведским столом. Проект требует разработки бота, способного обеспечивать удобный процесс заказа еды из предложенного меню, а также предоставлять административные функции для управления бронированием столов и учетом пользователей.

«Telegram-бот для шведского стола» предназначен для людей, работающих в сфере обслуживания, которые нуждаются в автоматизации некоторых процессов, а точнее для перевода с письменного вида заполнения бумаг о заказе блюда на онлайн-формат через Telegram-бота.

Существуют аналоги ИС «Telegram-бот для шведского стола», рассмотрим некоторые из них.

Telegram-бот для ресторанов «Quick Resto».

Интерфейс сервиса Telegram-бот для ресторанов «Quick Resto» представлен на Рисунке 1.

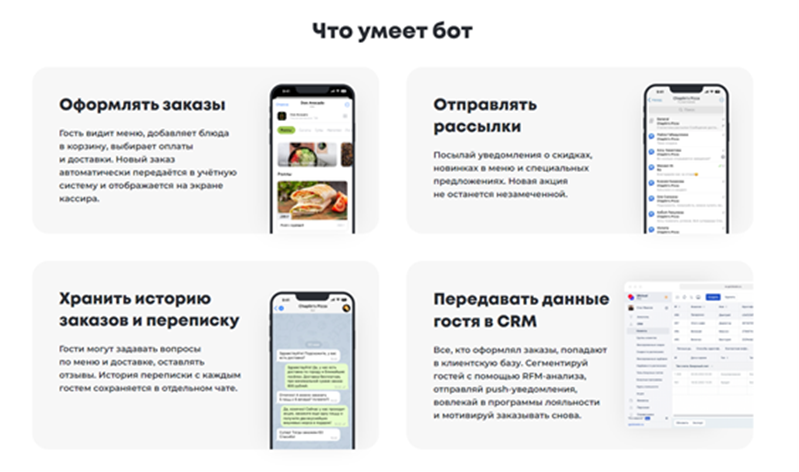


Рисунок 1 – Telegram-бот для ресторанов «Quick Resto»

Достоинствами данного аналога является простое оформление заказов, отправление рассылок, хранение истории заказов, переписки и просмотр данных гостя.

Telegram-бот «WorkEat».

Интерфейс сервиса Telegram-бот «WorkEat» представлен на Рисунке 2.

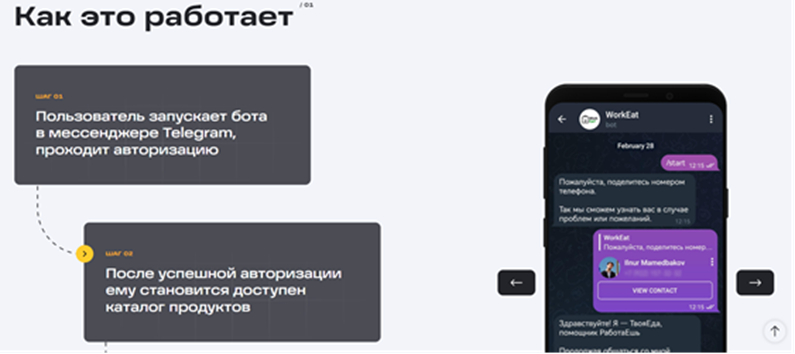
****

Рисунок 2 – Telegram-бот «WorkEat»

Достоинством сервиса является быстрая авторизация, доступность каталога продуктов и автоматическое добавление продукта в корзину, быстрая оплата покупки в окне с банковским эквайрингом.

Telegram-бот «tgEDA».

Интерфейс сервиса Telegram-бот «tgEDA» представлен на Рисунке 3.

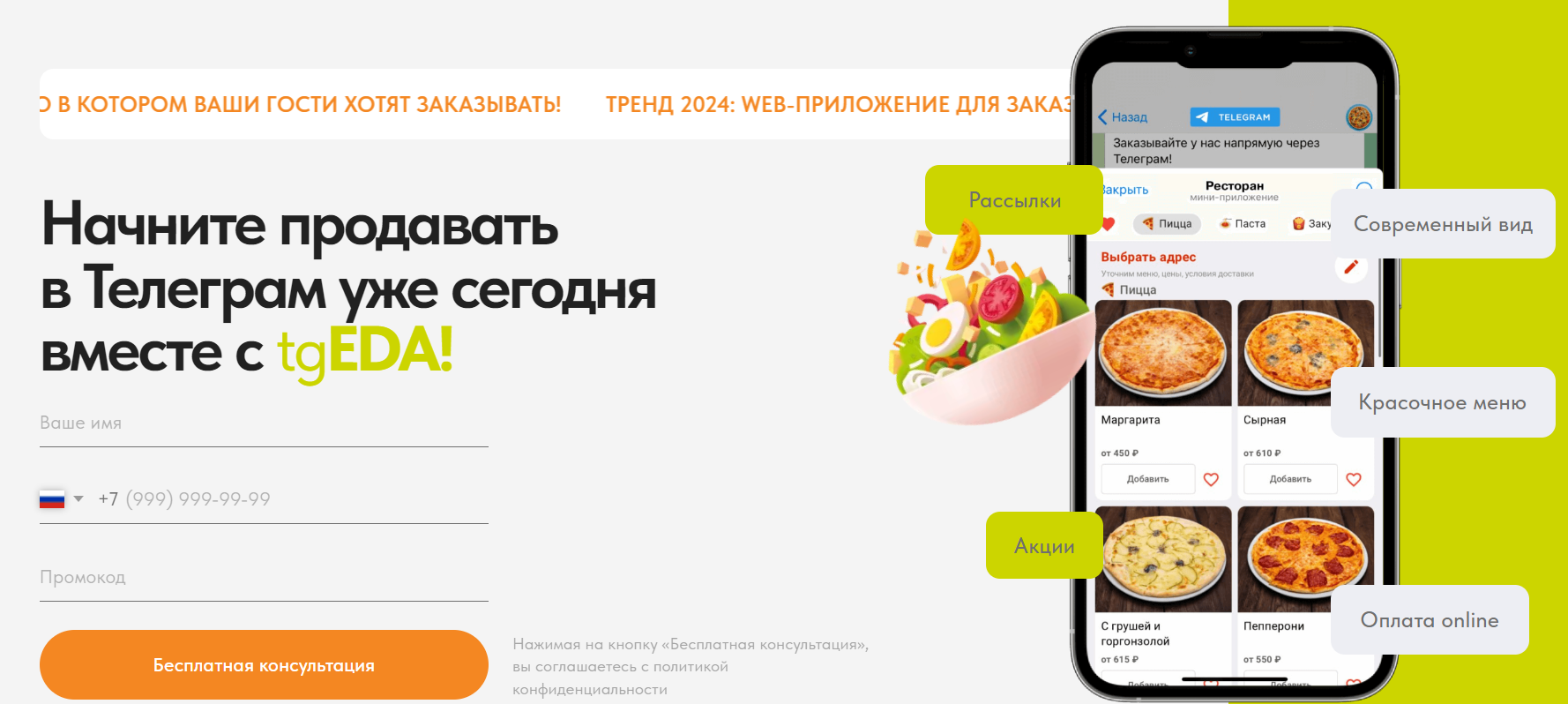


Рисунок 3 – Telegram-бот «tgEDA»

Достоинством сервиса является современный вид, красочное удобное меню, акции, оплата заказов, рассылки.

Общими недостатками рассмотренных аналогов является то, что в большинстве примеров невозможно реализовать администрирование из одного Telegram-бота, где будут совершать заказы пользователи.

Проанализировав три аналога ИС «Telegram-бот для шведского стола», можно сделать вывод, что Telegram-бот должен включать возможность сделать дистанционно заказ, например, на завтрак, обед или ужин, просмотр статистики, информации о заказах и просмотр меню. А также полное отсутствие рекламы.

# Требования к программе

## Требования к пользовательскому интерфейсу

Прототип экранной формы «Регистрация» представлен на Рисунке 4.

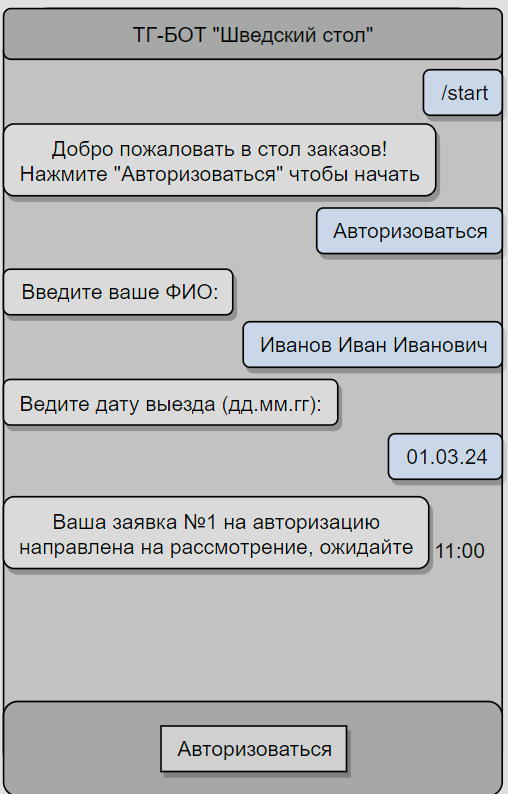


Рисунок 4 – Прототип экранной формы «Регистрация»

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 5.

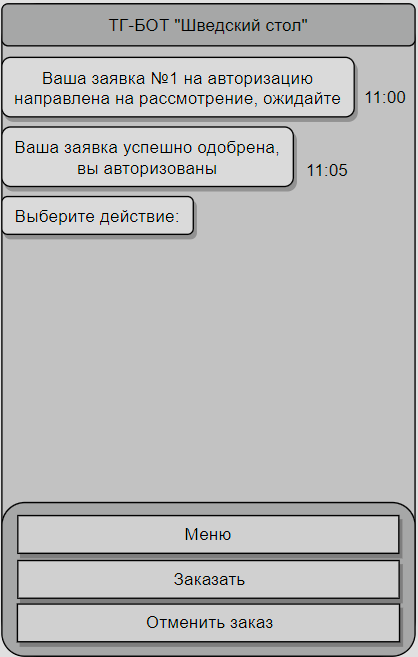


Рисунок 5 – Прототип экранной формы «Выбор действия»

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 6.

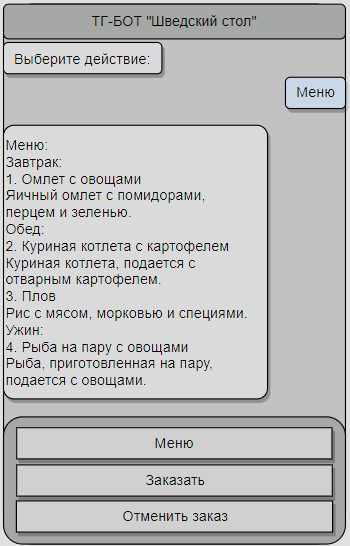


Рисунок 6 – Прототип экранной формы кнопки «Меню»

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 7.

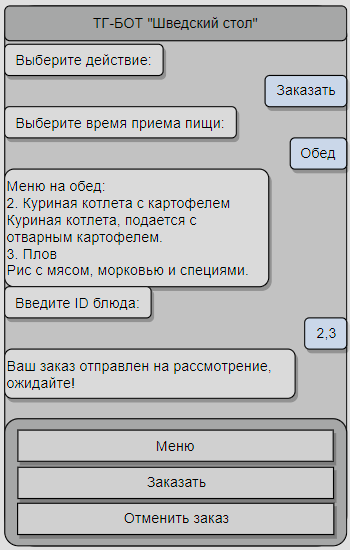


Рисунок 7 – Прототип экранной формы «Заказать»

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 8.

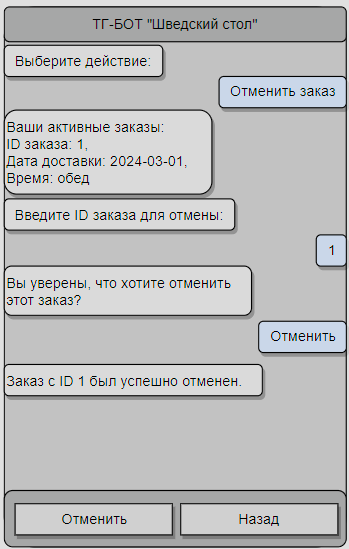


Рисунок 8 – Прототип экранной формы «Отменить заказ»

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 9.

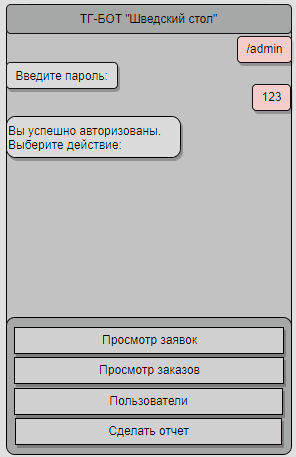


Рисунок 9 – Прототип экранной формы входа в админ-панель

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 10.

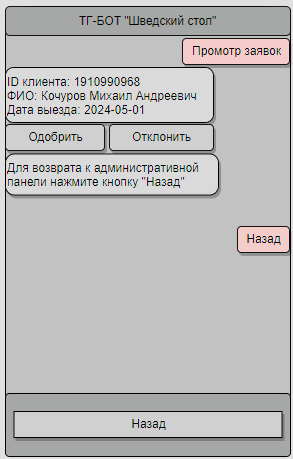


Рисунок 10 – Прототип экранной формы кнопки «Просмотр заказов»

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 11.

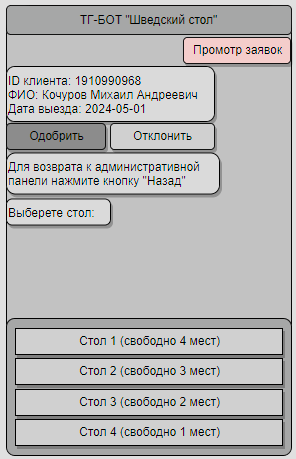


Рисунок 11 – Прототип экранной формы одобрения заявки

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 12.

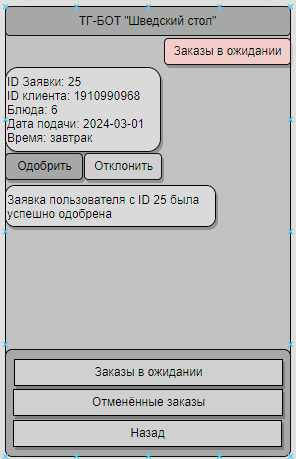


Рисунок 12 – Прототип экранной формы кнопки «Просмотр заказов»

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 13.

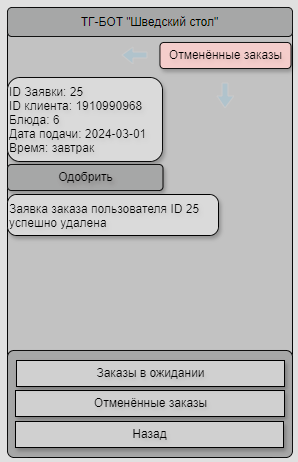


Рисунок 13 – Прототип экранной формы кнопки «Отмена заказов»

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 14.

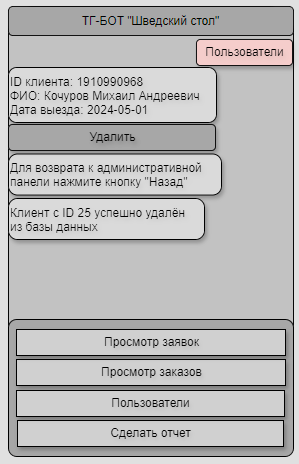


Рисунок 14 – Прототип экранной формы кнопки «Пользователи»

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 15.

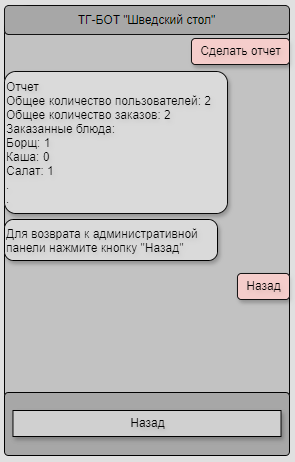


Рисунок 15 – Прототип экранной формы «Отчёт»

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.

## Требования к функциональным характеристикам

ИС «Telegram-бот для шведского стола» должно выполнять следующие функции:

Инициализация главного меню – функция проверяет статус пользователя и инициализирует главное меню при его первом взаимодействии с ботом.

Авторизация пользователя – функция собирает данные пользователя для авторизации (ФИО и дата выезда) и отправляет их на обработку в систему.

Просмотр меню – пользователь выбирает время приема пищи (завтрак, обед, ужин), после чего функция выводит доступное меню для указанного времени.

Оформление заказа – функция позволяет пользователю оформить заказ, выбрав блюда из меню, и сохраняет данные о заказе в системе.

Отмена заказа – пользователь может просмотреть свои активные заказы и отменить любой из них в течение ограниченного времени.

Администратор: просмотр заявок – функция позволяет администратору просматривать заявки на авторизацию, а также одобрять или отклонять их.

Администратор: просмотр заказов – администратор может просматривать активные и отмененные заказы пользователей, а также управлять ими.

Формирование отчета – администратор может сгенерировать отчет о текущих пользователях, заказах и блюдах, который будет выгружен в формате Excel.

Удаление пользователя по дате выезда – функция автоматически удаляет пользователя из системы по истечении даты выезда, вместе с его заказами.

## Требования к показателям назначения

Требования к показателям назначения не предъявляются.

## Требования к видам обеспечения

### Требования к математическому обеспечению

Математическое обеспечение должно обеспечивать точность вычислений

### Требования к информационному обеспечению

Telegram-бот должен предоставлять корректную информацию, которая позволяет осуществлять работу с базой данных.

#### Требования к форматам хранения данных

Для хранения структурированных данных, таких как информация о продуктах, клиентах, заказах, будет использоваться реляционная база данных PostgreSQL

Для хранения документов, изображений, отчётов, видео и других неструктурированных данных, приложение может использовать различные форматы файлов, такие как XLS.

#### Требования к лингвистическому обеспечению

В интерфейсе приложения используется русский язык. Telegram-бот будет разрабатываться на языке программирования Python.

### Требования к метрологическому обеспечению

Требования к метрологическому обеспечению не предъявляются.

### Требования к техническому обеспечению

Разрабатываемый программный продукт должен исполняться на ПК и смартфоне удовлетворяющим следующим минимальным требованиям к конфигурации:

процессор Intel(R) Core(TM) i3 1.20GHz;

оперативная память от 4 ГБ;

монитор;

компьютерная мышь;

клавиатура.

## Требования к надежности

Надежное (устойчивое) функционирование программы должно быть обеспечено выполнением совокупности организационно-технических мероприятий, перечень которых приведен ниже:

* + организацией бесперебойного питания технических средств;
  + осуществлением контроля входных данных;
  + регулярным выполнением рекомендаций Министерства труда и социального развития РФ, изложенных в Постановлении от 23 июля 1998 г. «Об утверждении межотраслевых типовых норм времени на работы по сервисному обслуживанию ПЭВМ и оргтехники и сопровождению программных средств»;
  + регулярным выполнением требований ГОСТ 51188–98. Защита информации. Испытания программных средств на наличие компьютерных вирусов.

## Требования к безопасности

Реализуемые решения должны соответствовать нормам электро- и  
пожаробезопасности в соответствии с требованиями законодательства РФ.

## Требования к патентной чистоте

Система должна отвечать требованиям к патентной чистоте согласно действующему законодательству Российской Федерации.

## Требования к перспективам развития

Развитие программы должно осуществляться посредством языка программирования Python, а также PostgreSQL.

# Состав и содержания работ

Анализ требований. На этом этапе определяются требования к приложению, его функциональность и основные возможности. Анализируются существующие аналоги и определяются их преимущества и недостатки.

В настройку рабочего окружения входят определение языка программирования и IDE, необходимых библиотек.

Проектирование приложения. На этом этапе разрабатывается архитектура приложения, определяется интерфейс и проектируются основные функции, а также БД.

Разработка приложения. На этом этапе создаются компоненты приложения, осуществляющие выполнение всех функций и элементов интерфейса.

Тестирование и отладка. После завершения разработки проводится тестирование приложения на наличие ошибок и недоработок. В случае обнаружения ошибок они устраняются.

# Порядок разработки

## Стадии разработки

Разработка должна быть проведена в семь стадий:

* настройка рабочего окружения;

разработка технического задания;

проектирование;

написание кода программного обеспечения;

тестирование программы;

доработка программы;

подготовка эксплуатационной документации.

## Этапы разработки

Этапы разработки содержание работ представлены в Таблице 1.

Таблица 1 – Этапы разработки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № этапа | Наименование этапа | Длительность | Состав работ | Результат |
| 1 | Настройка рабочего окружения | 1 неделя | Выбор языка программирования;  выбор IDE, в которой будет разрабатываться программа;  выбор библиотек языка программирования | Готовое к написанию кода рабочее место; |
| 2 | Техническое задание | 1 неделя | Поиск аналогов;  написание предметной области;  написание функциональных характеристик | Техническое задание; |
| 3 | Проектирование | 3 недели | Проектирование интерфейса приложения и БД, а также архитектуру приложения | Структура программного обеспечения, база данных; |
| 4 | Написание кода программного обеспечения | 1 месяца | Написание прототипа;  написание готовой программы | Программное обеспечение; |
| 5 | Тестирование программы | 3 недели | Протестировать приложение на основе методики тестирования | Список недоработок и ошибок в работе; |
| 6 | Доработка программы | 3 недели | Исправление ошибок, выявленных в ходе тестирования | Ошибки и недоработки в работе программного обеспечения устранены; |
| 7 | Подготовка эксплуатационной документации | 2 недели | Написание руководства пользователя | Курсовой проект; |

# Требования к документированию

Состав программной документации должен включать в себя:

* техническое задание;
* исходный код;
* курсовой проект.

# Порядок контроля и приёмки

## Виды испытаний

Испытания проводятся комиссией, в состав которой входят представители заказчика:

* Руководитель по УП.03 Долженкова Мария Львовна

Во время испытаний комиссия проверяет работу программы в соответствии со следующими позициями:

* корректное функционирование заданных в техническом задании функций;
* возможность функционирования на ПК с указанными минимальными системными требованиями.

Комиссии должны быть представлены техническое задание, курсовой проект, разработанная программа и доклад.

Структура доклада должна отражать следующие вопросы разработки:

* краткое описание задачи;
* результаты рассмотрения предметной области, аналогов, описание проблематики;
* описание этапа проектирования, возникавших проблем и путей их решения;
* выводы по результатам работы.