Содержание

[Введение 3](#_Toc5092)

[1 Термины и определения 4](#_Toc3463)

[2 Перечень сокращений 5](#_Toc6573)

[3 Основные сведения о разработке 6](#_Toc6170)

[3.1 Наименование разработки 6](#_Toc27367)

[3.2 Цель и задачи 6](#_Toc9262)

[3.3 Сведения о заказчике 6](#_Toc20355)

[3.4 Сроки разработки 6](#_Toc17273)

[3.5 Назначение разработки 7](#_Toc21344)

[3.5.1 Функциональное назначение программы 7](#_Toc14145)

[4 Описание предметной области 8](#_Toc3089)

[5 Требования к программе 14](#_Toc6846)

[5.1 Требования к пользовательскому интерфейсу 14](#_Toc23565)

[5.2 Требования к функциональным характеристикам 18](#_Toc21001)

[5.3 Требования к показателям назначения 18](#_Toc31883)

[5.4 Требование к видам обеспечения 18](#_Toc26901)

[5.4.1 Требование к математическому обеспечению 18](#_Toc27384)

[5.4.2 Требования к информационному обеспечению 19](#_Toc11193)

[5.4.3 Требования к лингвистическому обеспечению 19](#_Toc32620)

[5.4.4 Требования к метрологическому обеспечению 19](#_Toc18143)

[5.4.5 Требования к техническому обеспечению 19](#_Toc23441)

[5.5 Требования к надёжности 19](#_Toc1103)

[5.6 Требование к безопасности 20](#_Toc7624)

[5.7 Требования к патентной частоте 20](#_Toc32177)

[5.8 Требования к перспективам развития 20](#_Toc25251)

[6 Состав и содержание работ 21](#_Toc18619)

[7 Порядок разработки 22](#_Toc11998)

[7.1 Стадии разработки 22](#_Toc28064)

[7.2 Этапы разработки 22](#_Toc28294)

[8 Требования к приемо-сдаточным процедурам 24](#_Toc16963)

# 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.

# Термины и определения

В настоящем документе используются следующие термины и определения:

Интерфейс - граница между двумя функциональными объектами, требования к которой определяются стандартом;

База данных - упорядоченный набор структурированной информации или данных, которые обычно хранятся в электронном виде в компьютерной системе;

PyQt5 - библиотека Python для создания графических интерфейсов пользователя (GUI) с использованием Qt;

Telebot - библиотека Python, которая предоставляет интерфейс для взаимодействия с Telegram Bot API;

Psycorg2 - библиотека Python, предназначенная для работы с базами данными PostgreSQL;

PostgreSQL - свободная объектно-реляционная система управления базами данных (СУБД).

# Перечень сокращений

В настоящем документе используются следующие сокращения:

ЯП - Язык программирования;

ИС - информационная система;

ЭВМ - электронно-вычислительная машина;

IDE - Интегрированная среда разработки;

ПК - персональный компьютер;

БД - база данных;

СУБД - система управления базами данными;

ГОСТ - государственный стандарт;

РФ - Российская Федерация.

# Основные сведения о разработке

## Наименование разработки

Наименование разрабатываемой в ходе работы программы: «Telegram-бот для распределения заявок по ремонту».

## Цель и задачи

Целью работы является разработать Telegram-бота для распределения заявок по ремонту.

Задачи разработки:

* Проанализировать предметную область;
* Спроектировать БД;
* Спроектировать админ-панель;
* Спроектировать telegram-бота студетов;
* Спроектировать telegram-бота сотрудников.

Исполнителями являются студенты Колледжа ФГБОУ ВО «Вятский государственный университет»

* Огородников Матвей Дмитриевич, ИСПк-304-52-00;
* Седов Павел Алексеевич, ИСПк-304-52-00;
* Шишкина Мария Андреевна, ИСПк-304-52-00;
* Карина Яна Владленовна , ИСПк-304-52-00;
* Ушакова Алена Матвеевна, ИСПк-304-52-00.

## **Сведения о заказчике**

Заказчиком является коллектив преподавателей колледжа ВятГУ в составе:

* преподаватель по УП.07 Учебная практика - Волкова Екатерина Владимировна;
* руководитель специальности «Информационные системы и программирование» - Сергеева Елизавета Григорьевна

## Сроки разработки

Работа должна быть выполнена в следующие сроки:

Начало разработки - 02.09.2024

Окончание разработки - 12.04.2024

## Назначение разработки

### Функциональное назначение программы

Назначение разработки «Telegram-бот для распределения заявок по ремонту» заключается в обеспечении эффективного управления заявками на обслуживание в общежитии с использованием телеграмм ботов и админ-панели.

# Описание предметной области

В данной предметной области были рассмотрены несколько аналогов телеграмм-ботов, предназначенных для приёма и обработок заявок на ремонт и техподдержку, а также были выявлены их основные достоинства и недостатки.

1. Телеграм-бот «factors\_bot» - бот, который предназначен для приёма и обработки заявок по ремонту и техподдержки.

После запуска бот требует авторизацию через номер телефона, привязанный к используемому аккаунту, затем ввести необходимые личные данные и описать проблему. После создания заявки ее можно проверить по коду, либо связаться с оператором. Интерфейс аналога представлен на рисунке 1.

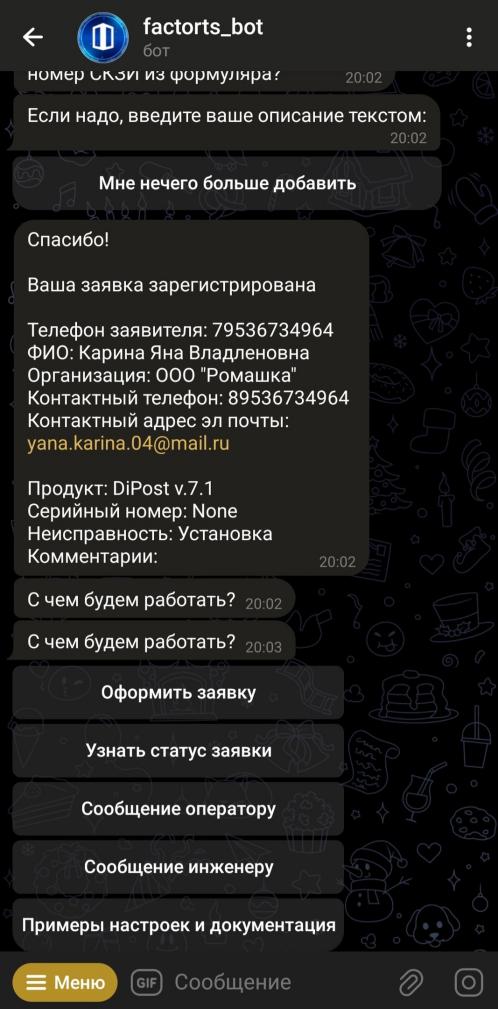


Рисунок 1 - factors\_bot

Из плюсов можно выделить удобство регистрации пользователя, описание проблемы выбором из списка, возможность добавить описание от руки и возможность связи с оператором.

Из минусов, в заявке не указан ее номер, следовательно ее статус нельзя посмотреть.

1. Телеграм-бот «Михалыч» - бот, который создан для работы исполнителей с заявками. В его функционал входит: показать назначенные заявки, менять статус назначенных заявок, создавать личную анкету о работнике.

После запуска бот запрашивает авторизацию через номер телефона, привязанный к аккаунту, затем личные данные, после чего создаёт анкету работника. Приём заявок происходит через отдельную группу, где появляются заявки. Так же бот может запросить геолокацию для подбора заявок поблизости. Когда работник принимает заявку, он может добавить фото, сменить ее статус, либо перенести на другое время, добавить описание выполненных работ. Завершить работу с заявкой работник может отказом либо выполнением. Интерфейс аналога представлен на рисунке 2.

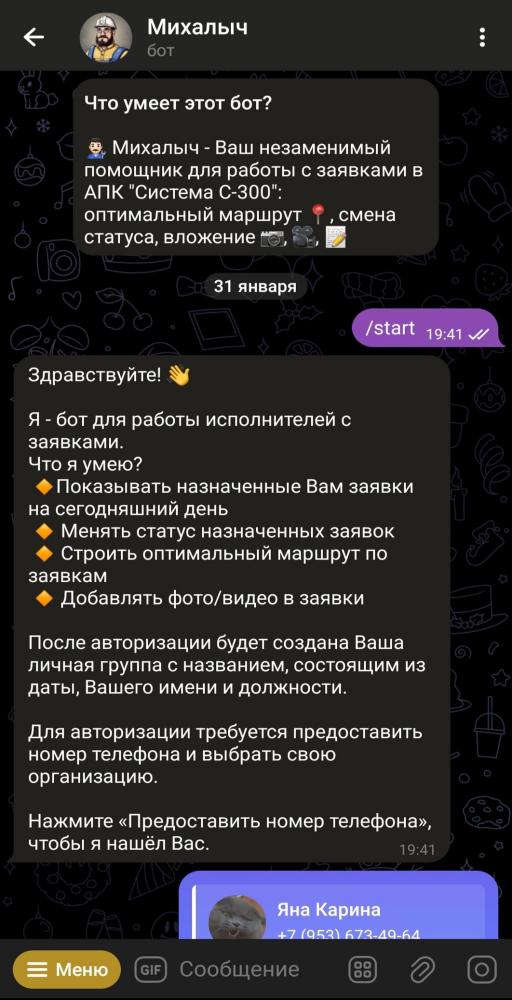


Рисунок 2 - Михалыч

Из плюсов бота можно выделить удобную регистрацию, возможность подбора заявок с учётом геолокации, возможность отказа от заявки и её переноса, подробное описание выполненной работы.

Из минусов можно выделить наличие отдельной группы для просмотра заявок, вместо нахождения их в том же боте.

1. OkdeskBot - бот, который создан для службы поддержки, приёма и обработки заявок. Интерфейс аналога представлен на рисунке 3, 4.



Рисунок 3 - OkdeskBot

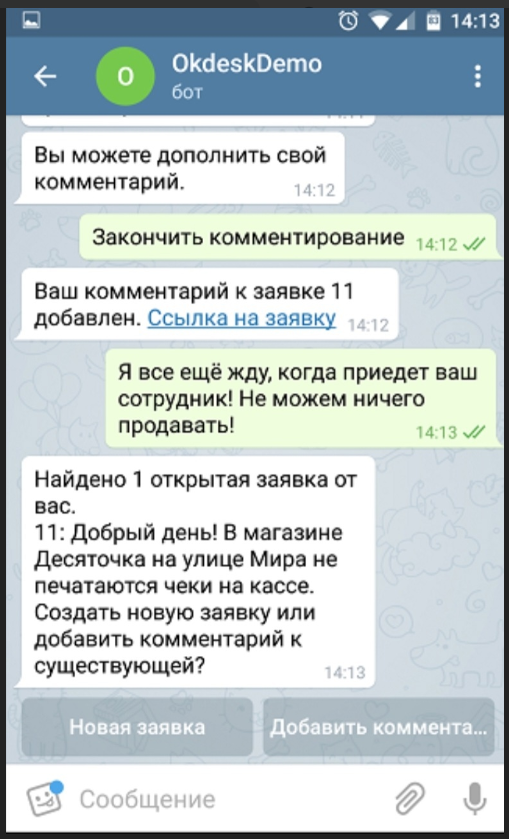


Рисунок 4 - OkdeskBot

Из плюсов можно выделить удобство использования. OkdeskBot предлагает простой и интуитивно понятный интерфейс, что делает его удобным в использовании для всех пользователей, так же быстрое получение ответов, благодаря автоматическому обработчику заявок, OkdeskBot способен быстро и эффективно отвечать на вопросы пользователей.

Из минусов можно выделить, ограничение функционала, так как OkdeskBot предлагает базовый функционал для работы с заявками, что может ограничить возможности управления более сложными задачами.

1. «Аpple.Service» **-** телеграмм бот для сбора заявок на ремонт техники Apple. Интерфейс аналога представлен на рисунках 5-8.

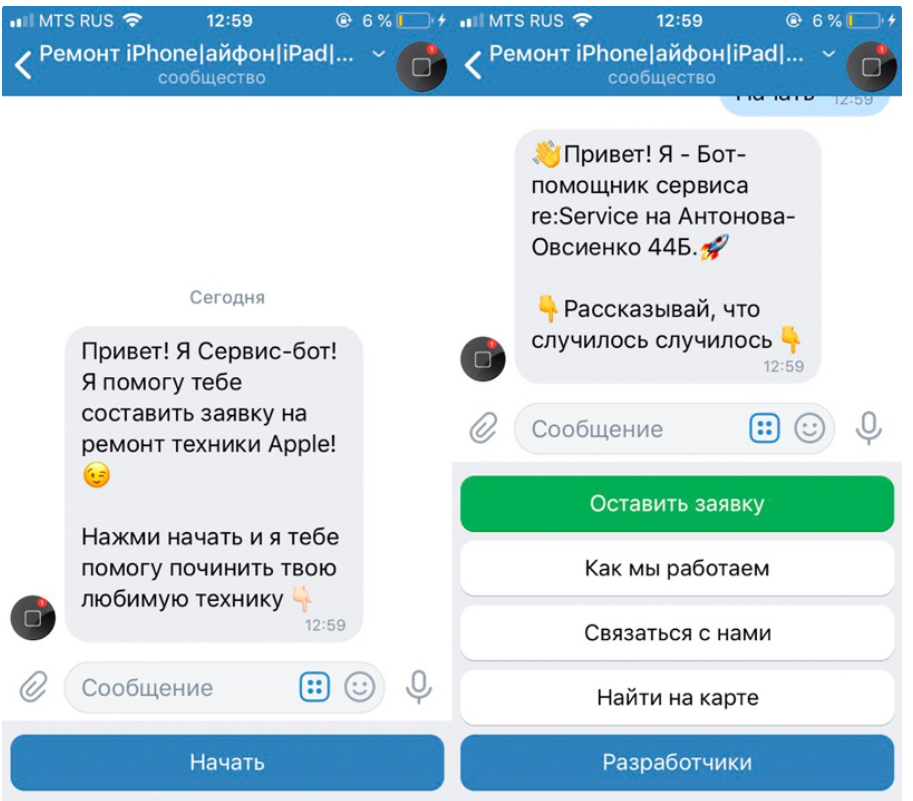


Рисунок 5 - Apple.Serice

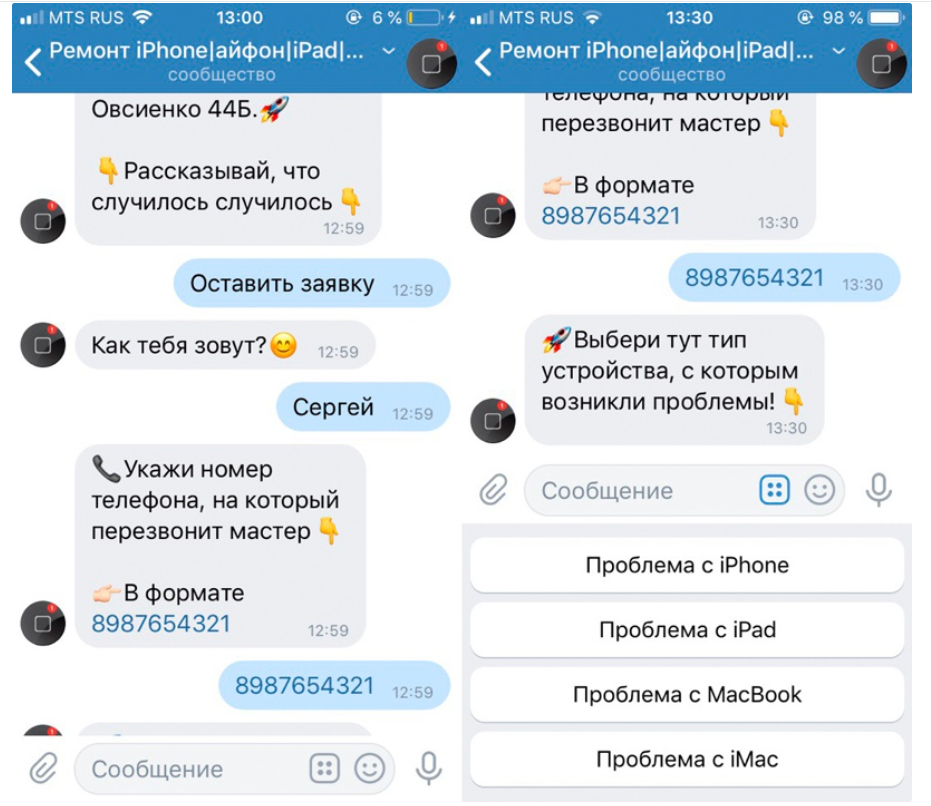


Рисунок 6 - Apple.Serice

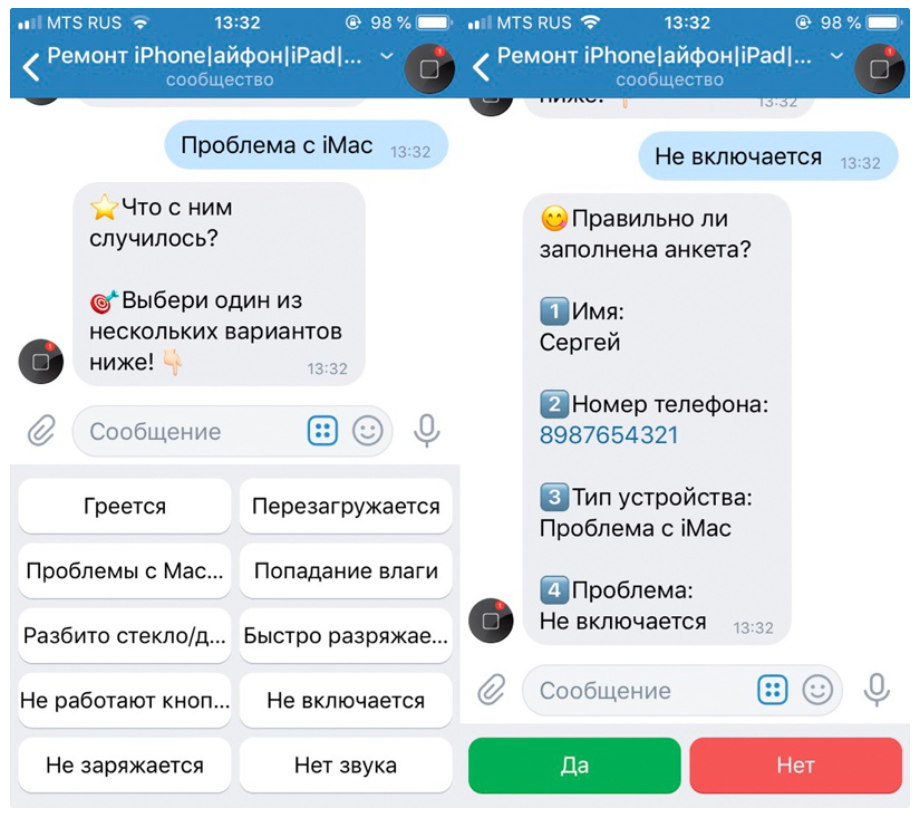


Рисунок 7 - Apple.Serice

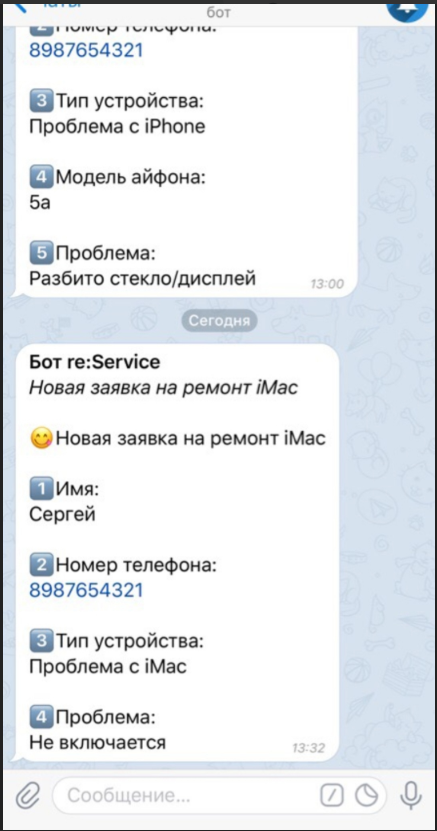


Рисунок 8 - Apple.Serice

Из плюсов бота можно выделить простоту использования. Бот позволяет пользователям легко и быстро подать заявку на ремонт техники Apple. Все необходимые поля присутствуют, и нет необходимости дополнительно взаимодействовать с оператором, также он автоматически принимает заявки, что позволяет оперативно начать их обработку.

Из минусов бот предназначен только для сбора заявок на ремонт техники Apple, и не предоставляет другие дополнительные услуги или возможности.

# Требования к программе

## Требования к пользовательскому интерфейсу

Панель администратора должна содержать 3 вкладки:

* Вкладка «Студенты», где будут указываться id, ФИО студентов, корпус и комната;
* Кнопка «Добавить» для добавления нового студента;
* Кнопка «Сохранить» для сохранения информации о студенте;
* Кнопка «Удалить» для удаления студента.

Прототип представлен на рисунке 9.

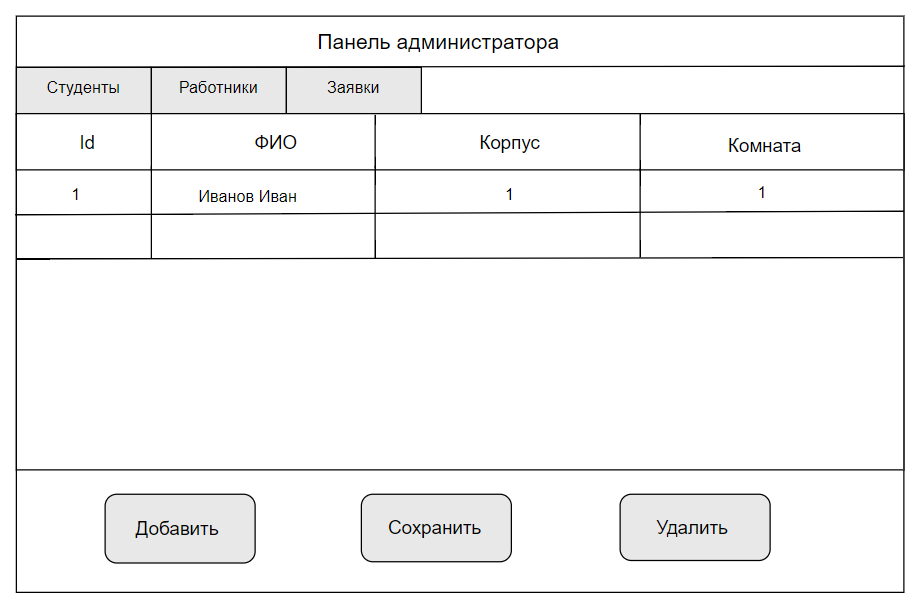


Рисунок 9 - вкладка «Студенты»

* Вкладка «Работники» , где будут указываться id, ФИО работника, должность и корпус;
* Кнопка «Добавить» для добавления нового работника;
* Кнопка «Сохранить» для сохранения информации о работнике;
* Кнопка «Удалить» для удаления работника.

Прототип представлен на рисунке 10.

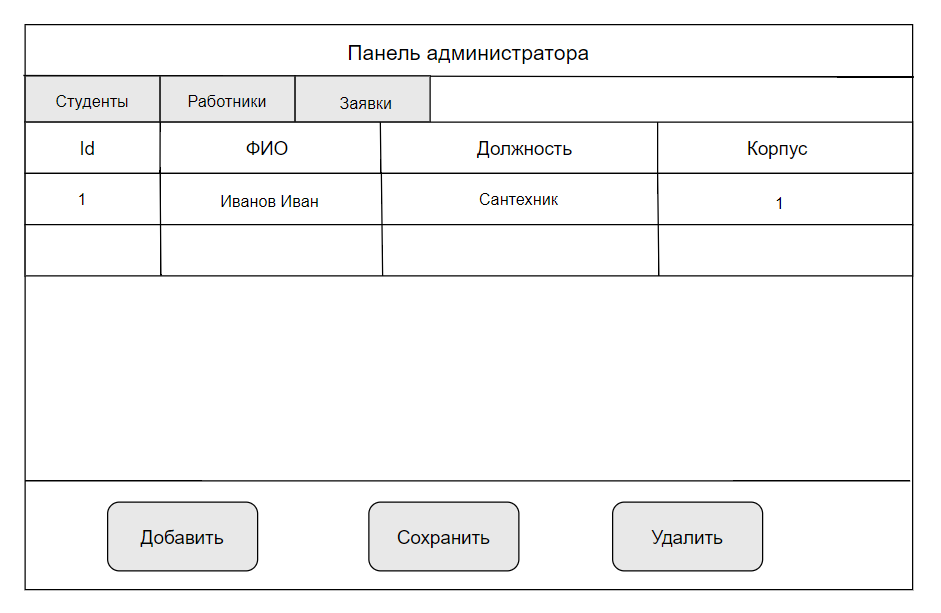


Рисунок 10 - вкладка «Работники»

* Вкладка «Заявки», где будут указываться: id, ФИО студента, корпус, ФИО работника, должность, а также описание и статус работы;
* Кнопка «Добавить» для добавления новой заявки;
* Кнопка «Сохранить» для сохранения информации о заявке;
* Кнопка «Удалить» для удаления заявки.

Прототип представлен на рисунке 11.



Рисунок 11 - вкладка «Заявки»

Телеграмм бот «Сотрудники» будет функционировать следующим образом:

* При вводе команды \start бот запускается и будет готов к работе;
* После запуска, бот приветствует пользователя и запрашивает его номер телефона;
* После введения номера телефона, бот благодарит пользователя за его отправку;
* При нажатии на кнопку «Принять», бот принимает заявку;
* При нажатии на кнопку «Отклонить», бот отклоняет заявку и запрашивает причину отклонения;
* После того, как указывается причина, заявка отклоняется по данной причине.

Прототип телеграмм бота «Сотрудники» представлен на рисунке 12.

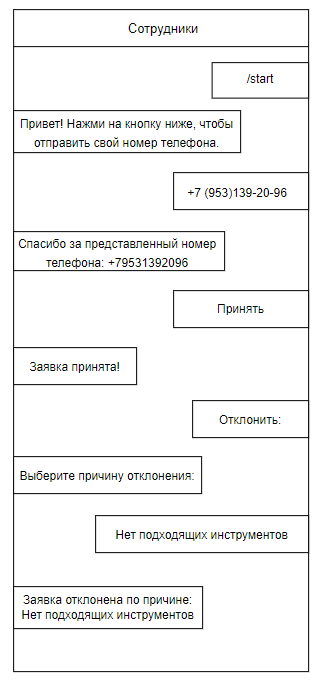


Рисунок 12 - телеграмм бот «Сотрудники»

Телеграмм бот «Студенты» будет функционировать следующим образом:

* При вводе команды \start бот запускается и будет готов к работа;
* После запуска, бот приветствует пользователя и запрашивает его номер телефона;
* После введения номера телефона, бот благодарит пользователя за его отправку;
* При нажатии на кнопку «Заполнить анкету», бот запрашивает должность работника;
* После ввода должности работника, бот выводит данную должность и далее запрашивает описание работы;
* После ввода описания работы, бот также выводит работу и благодарит пользователя за заполнение анкеты.

Прототип представлен на рисунке 13.

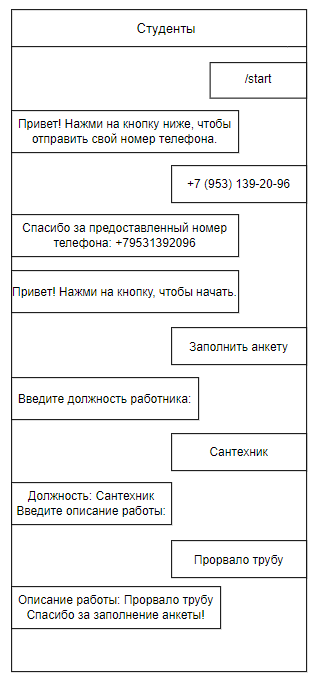


Рисунок 13 - телеграмм бот «Студенты»

## Требования к функциональным характеристикам

## Требования к показателям назначения

Требования к показателям назначения не предъявляются.

## Требование к видам обеспечения

### Требование к математическому обеспечению

Требования к показателям назначения не предъявляются.

### Требования к информационному обеспечению

Информационная система должна предоставлять удобный и простой интерфейс, который будет выполнять корректную работу с БД и будет доступен для каждого.

### Требования к лингвистическому обеспечению

Развитие программы должно осуществляться на ЯП Python, с использованием библиотек PyQt5 для создания графического интерфейса, telebot для интеграции с Telegram, psycorg2 для работы с базой данных PostgreSQL.

### Требования к метрологическому обеспечению

Требования к метрологическому обеспечению не предъявляются.

### Требования к техническому обеспечению

Разрабатываемый программный продукт должен исполняться на ПК, удовлетворяющем следующим минимальным требованиям к конфигурации:

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

оперативная память от 4 ГБ;

монитор;

компьютерная мышь;

клавиатура.

## Требования к надёжности

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

а) организацией бесперебойного питания технических средств;

б) осуществлением контроля входных данных;

в) 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»;

г) регулярным выполнением требований ГОСТ 51188-98. Защита информации. Испытание программных средств на наличие компьютерных вирусов;

## Требование к безопасности

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

## Требования к патентной частоте

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

## Требования к перспективам развития

Для дальнейшего развития Telegram-бота следует учитывать следующие аспекты:

* Развитие функциональности. Система должна постоянно развиваться и обновляться, чтобы удовлетворять потребности пользователей и соответствовать современным технологическим требованиям;
* Улучшение производительности. Система должна иметь возможность повышения производительности, обеспечивая более эффективное выполнение задач.

# Состав и содержание 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.

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 программы должна проходить в следующих стадиях:

* Настройка рабочего окружения;
* Разработка техническое задания;
* Проектирование;
* Написание кода программного обеспечения;
* Доработка информационной системы;
* Подготовка эксплуатационной документации.

## Этапы разработки

Этап разработки содержание работ представлены в Таблице 1

Таблица 1 –Этапы разработки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № этапа | Наименование | Длительность | Состав работ | Результат |
| 1 | Настройка рабочего окружения | 1 неделя | Выбор язык для написания программы;  Выбор IDE на котором будет писаться данная программа;  Выбор библиотеки которые будут использоваться для написания программы; | Готовое к написанию кода рабочее место; |
| 2 | Техническое задание | 1 месяца | Поиск аналогов;  Написание предметной области; | Техническое задание; |
| 3 | Проектирование | 3 недели | Проектирование интерфейса приложения; | Структура программного обеспечения, база данных; |
| 4 | Написание кода программного обеспечения | 2 месяц | 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 | 1 недели | Исправление ошибок в программе которые были выявлены в тестирование; | Ошибки и недоработки в работе программного обеспечения устранены; |
| 7 | Подготовка эксплуатационной документации | 2 недели | Написание курсового проекта | Курсовой проект; |

# Требования к приемо-сдаточным процедурам

Приемо-сдаточные процедуры проводятся комиссией, в которую входят представители

заказчика.

Назначение места и времени приемо-сдаточных процедур осуществляется учебным

отделом колледжа ВятГУ.

Процедура защиты состоит из следующих шагов:

Разработанный продукт необходимо запустить на устройстве, соответствующем требованиям.

Предоставить на проверку ранее запущенную программу с техническим заданием,

исходным кодом, программой.

После этого сдающий должен выступить с докладом о процессе разработки

программы и её особенностях в течение 3 минут.

Оценка работы проводится коллегиальным решением комиссии на основе результатов

защиты и представления преподавателей.