



## **МИНОБРНАУКИ РОССИИ**

Федеральное государственное бюджетное образовательное  
учреждение высшего образования

**«МИРЭА – Российский технологический университет» РТУ  
МИРЭА**

Колледж программирования и кибербезопасности

**Практическая работа №2 учебной дисциплины:**

**ОП.15 Управление проектами**

**Тема: Описание содержания проекта**

Выполнил студент группы ПКС-41

Бондаренко П. Е.

Преподаватель Мымрина М. Л.

Москва  
2025

## Описание содержания проекта.

### Оглавление

Название проекта.....	3
Цели и задачи проекта .....	4
Требования к проектному решению и результаты проекта.....	5
Границы проекта .....	7
Способ реализации проекта .....	8
Первоначальная иерархическая структура работ .....	9
Потребность в ресурсах, штатное расписание и организационная структура ...	10
Укрупнённый календарный план .....	11
Критические факторы успеха .....	12
Допущения проекта.....	13
Ограничения проекта .....	14
Связь с прочими текущими программами и проектами .....	15
Первоначально сформулированные риски .....	16
Смета расходов .....	17
Требования к управлению конфигурацией проекта.....	18
Критерии приёмки результатов проекта.....	19

## **Название проекта.**

CRM система для управляющей компании жилищно-коммунальных услуг.

## **Цели и задачи проекта**

### **Цели:**

1. Автоматизировать процесс приёма и обработки заявок жителей;
2. Повысить эффективность работы управляющей компании;
3. Упростить взаимодействие компании с клиентами и обеспечить прозрачность в нём.

### **Задачи:**

1. Разработать модуль регистрации и авторизации пользователей;
2. Реализовать функционал подачи заявок;
3. Создать интерфейс для сотрудников позволяющий распределять заявки;
4. Реализовать систему создания отчётности, ведения метрик.

# Требования к проектному решению и результаты проекта

## *Требования к функциональным характеристикам*

Система должна обеспечивать:

- Регистрацию и аутентификацию пользователей;
- Просмотр и обработку заявок сотрудниками управляющей компании (назначение ответственных, изменение статуса, добавление комментариев);
- Подачу заявок жителями с возможностью прикрепления текстового описания и изображений;
- Ведение истории изменения статусов заявок;
- Формирование отчетов (количество заявок за период, среднее время обработки, распределение по категориям);
- Разграничение прав доступа в зависимости от роли пользователя.

## *Требования к надёжности:*

- Доступность системы не ниже 95% времени в месяц;
- Сохранность данных при сбоях;
- Корректная работа при одновременной работе до 100 активных пользователей.

## *Требования к составу и параметрам технических средств:*

- Сервер приложений: 2 CPU, 4 ГБ RAM, 50 ГБ дискового пространства;
- База данных: PostgreSQL 15;
- ОС сервера: Linux.

## *Условия эксплуатации:*

- Эксплуатация в среде современных веб-браузеров (Chrome, Firefox, Edge, Safari);
- Доступ к системе по сети Интернет;
- Поддерживаемые устройства: ПК, ноутбуки, планшеты, телефоны.

## *Требования к документации:*

- Руководство пользователя;
- Руководство администратора;
- Установка и эксплуатация.

## *Результаты проекта:*

- Разработанная CRM-система, доступная в веб-интерфейсе;
- База данных заявок и отчётов;

- Полный комплект документации для пользователей и администраторов;
- Развёрнутая система на тестовом сервере.

## **Границы проекта**

В рамках проекта создаётся MVP-система (минимально жизнеспособный продукт). В них входит реализация всего функционала необходимого для удовлетворения заявленного функционала: учёт заявок, роли пользователей, базовые отчёты.

В рамки проекта не входят: интеграции с внешними платёжными системами, приложение для ПК, приложение для ОС IOS и macOS, чат-боты.

## **Способ реализации проекта**

Серверная часть – Java (Spring Boot)

Клиентская часть – JavaScript

База данных – PostgreSQL

RestAPI для работы модулей друг с другом.

Docker для развёртки.



# **Первоначальная иерархическая структура работ**

## **Этап 1. Анализ и проектирование (срок: 3 недели)**

- 1.1. Сбор требований
  - интервью с заказчиком (3 дня)
  - анализ бизнес-процессов управляющей компании (4 дня)
- 1.2. Формализация требований
  - подготовка документа «Требования к системе» (1 неделя)
- 1.3. Проектирование архитектуры системы
  - описание компонентов (3 дня)
  - проектирование базы данных (3 дня)

## **Этап 2. Разработка (срок: 6 недель)**

- 2.1. Серверная часть
  - реализация регистрации и аутентификации (1 неделя)
  - модуль подачи и обработки заявок (2 недели)
  - административный модуль (1 неделя)
- 2.2. Клиентская часть
  - интерфейс для жителей (1 неделя)
  - интерфейс для сотрудников (1 неделя)

## **Этап 3. Тестирование (срок: 2 недели)**

- 3.1. Функциональное тестирование (1 неделя)
- 3.2. Нагрузочное тестирование (3 дня)
- 3.3. Исправление выявленных ошибок (4 дня)

## **Этап 4. Внедрение (срок: 2 недели)**

- 4.1. Подготовка пользовательской документации (5 дней)
- 4.2. Настройка окружения и развертывание системы (3 дня)
- 4.3. Обучение персонала (4 дня)

## **Этап 5. Сопровождение (после завершения внедрения, не входит в срок реализации MVP)**

- 5.1. Техническая поддержка пользователей
- 5.2. Устранение ошибок и доработка по заявкам
- 5.3. Выпуск обновлений

## **Потребность в ресурсах, штатное расписание и организационная структура**

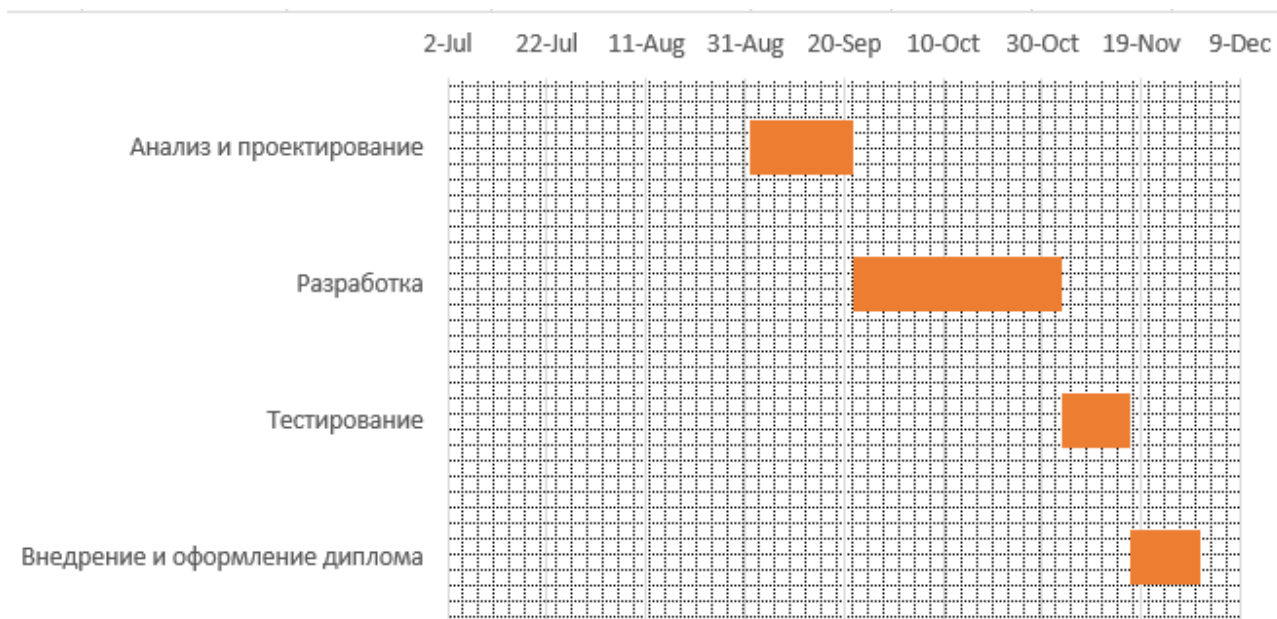
Исполнитель: один студент.

Роли: аналитик, разработчик, тестировщик.

Ресурсы: ПК, доступ к интернету.

Организационная структура: студент работает самостоятельно под наблюдением дипломного руководителя.

## Укрупнённый календарный план



## **Критические факторы успеха**

- **Корректная реализация базового функционала:** все ключевые функции системы (регистрация, подача заявок, обработка заявок сотрудниками) должны работать без ошибок.
- **Соблюдение сроков выполнения:** своевременное выполнение каждого этапа (анализ, разработка, тестирование, внедрение) критично для успешной сдачи диплома.
- **Тестирование и исправление ошибок:** выявление и устранение критических багов до демонстрации проекта.
- **Документирование результатов:** наличие полноценной документации (руководство пользователя, описание установки) повышает восприятие проекта и его оценку.
- **Соответствие требованиям ТЗ:** все функции и ограничения должны строго соответствовать заранее утвержденным требованиям.

## **Допущения проекта**

Все работы выполняются одним студентом.

Используются только открытые технологии: Java, Spring Boot, PostgreSQL.

Доступ к информации о бизнес-процессах получен в открытых источниках или моделируется.

Система проверяется и принимается без привлечения реальных пользователей.

Доступ к современным веб-браузерам у условного «жителя» и «сотрудника» предполагается как обеспеченный.

## **Ограничения проекта**

Проект выполняется в рамках дипломного графика ( $\approx 6$  месяца).

Реализация ограничена минимально жизнеспособным продуктом (MVP) — без интеграции с внешними системами, мобильного приложения и расширенных отчетов.

Один исполнитель — все этапы (анализ, разработка, тестирование, внедрение) выполняются без команды.

Использование только локального ПК с ограниченными ресурсами.

Ограниченные знания и опыт студента в части некоторых технологий.

## **Связь с прочими текущими программами и проектами**

Проект не зависит от сторонних разработок.

## **Первоначально сформулированные риски**

1. Возможные задержки на этапе анализа требований и разработки, приводящие к сдвигу календарного плана.
2. Ограниченное тестирование может привести к выявлению ошибок после демонстрации.
3. Ограниченные аппаратные ресурсы могут повлиять на развертывание и тестирование системы.



## **Смета расходов**

Программное обеспечение: бесплатные и открытые технологии — 0 руб.

Аппаратные ресурсы: использование личного ПК — без финансовых затрат.

Учебные материалы и документация: печатные материалы — ориентировочно 500 руб.

Основной ресурс проекта — время и знания исполнителя: 13 недель  $\times$  40 часов/неделя = ~520 часов.

Общие финансовые затраты минимальны; основной вклад — труд студента.

## Требования к управлению конфигурацией проекта

Проект должен иметь **модульную структуру**, включающую работу с клиентским приложением, серверным приложением и базу данных.

Структура проекта должна обеспечивать логическое разделение компонентов:

- модуль авторизации и аутентификации,
- модуль работы с заявками,
- административный модуль,
- модуль формирования отчетов.

Код проекта хранится в системе контроля версий Git с ветвлением: `main` — стабильная версия, `dev` — разработка.

Документация хранится совместно с исходным кодом и описывает архитектуру, установку и эксплуатацию проекта.

## **Критерии приёмки результатов проекта**

1. **Функциональность:** жители могут регистрироваться и создавать заявки; сотрудники могут обрабатывать заявки, назначать ответственных и изменять статусы.
2. **Корректность работы модулей:** все функции работают без ошибок, данные корректно сохраняются в базе и отображаются в интерфейсе.
3. **Доступность и стабильность:** система развёрнута и доступна в веб-браузере, корректно работает при нагрузке до расчетного числа пользователей.
4. **Соответствие требованиям ТЗ:** реализованы все функции, указанные в ТЗ, включая разграничение прав доступа и формирование отчетов.
5. **Документированность и воспроизводимость:** подготовлена полная документация и обеспечена возможность развернуть систему на другом ПК или сервере без потери функциональности.