Федеральное государственное автономное учебное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Мегафакультет компьютерных технологий и управления

Факультет программной инженерии и компьютерной техники

**Курсовая работа по дисциплине**

**“Информационные системы”**

**Этап 1**

Группа: P3318

Студент: Горло Евгений Николаевич

Преподаватель: Райла Мартин

2024

[**1. Introduction**](#_heading=h.ncnnft32olc8) **3**

[1.1 Purpose](#_heading=h.64118qxcym0o) 3

[1. 2 Scope](#_heading=h.tof39c50pf4x) 3

[1.3 Definitions, Acronyms and Abbreviations](#_heading=h.qbeio97n8wmn) 4

[1.4 Reference](#_heading=h.54zyjuw8qc2d) 4

[1.5 Overview](#_heading=h.iormrwokg7vw) 4

[**2. Overall Description**](#_heading=h.pp35ii7dfhaf) **4**

[2.1 Product functions](#_heading=h.j9xyuqprmtk5) 5

[2.2 User characteristics](#_heading=h.3kgir8tl139w) 5

[2.3 Assumptions and dependencies](#_heading=h.2mo9558dyk5y) 6

[2.4 Constraints](#_heading=h.o260ubrb98ah) 6

[**3. Specific Requirements**](#_heading=h.byk3tabbjb5) **6**

[3.1 Functionality](#_heading=h.mc0ou834izoc) 6

[3.1.1 Регистрация и аутентификация:](#_heading=h.yttqftyx5j0a) 6

[3.2 Usability](#_heading=h.tx1x0vyl4cp9) 10

[3.2.1 Интуитивный интерфейс:](#_heading=h.bzd80tbmqd0e) 10

[Система должна обеспечивать простой и интуитивно понятный интерфейс, позволяющий пользователям воспользоваться основным функционалом в 2-3 клика.](#_heading=h.j277bfm3nuw2) 10

[3.2.2 Навигационные элементы:](#_heading=h.pstdwdxsd6fq) 10

[3.2.3 Совместимость с устройствами:](#_heading=h.2ccr4gruc1a9) 10

[3.3 Reliability](#_heading=h.i06tc72vzvqf) 11

[3.4 Performance](#_heading=h.ipuk5sg8fyh) 11

[3.4.1 Время ответа системы](#_heading=h.2xi7gdg3y5kr) 12

[3.4.2 Обработка транзакций](#_heading=h.od9765v2s669) 12

[3.4.3 Количество одновременно работающих пользователей](#_heading=h.xdougr21mlul) 12

[3.5 Design Constraints](#_heading=h.hq6b08vi7l28) 12

[3.6 Interfaces](#_heading=h.1j4dp2sabcfx) 13

[3.6.1 User Interfaces (Пользовательские интерфейсы)](#_heading=h.2un4de9q168l) 13

[3.7 Licensing Requirements](#_heading=h.qnoe4gz4abkw) 15

[3.8 Data Protection and Privacy](#_heading=h.hzbp2dln94xi) 15

[**4. Use cases**](#_heading=h.buvmiet28rjb) **17**

**Software Requirements Specification**

# 1. Introduction

*[Введение предоставляет обзор на весь документ в целом и включает в себя следующие разделы - назначение, область применения, определения и аббревиатуры, ссылки и обзор.]*

Этот документ представляет собой комплексное описание требований к системе **ProTalent**, которая создается для объединения заказчиков и исполнителей на единой платформе.

## 1.1 Purpose

*[Укажите назначение данного документа****.****]*

Цель данного документа — подробно описать функциональные и нефункциональные требования к системе фриланс-биржи **ProTalent**. Это описание будет использовано для проектирования, разработки и тестирования системы, обеспечивая единое понимание требований всеми участниками процесса.

## 1. 2 Scope

*[Приведите краткое описание области применения данного документа, к какому(им) проекту(ам) он относится, кем будет использоваться и т.д.]*

Этот документ описывает проект разработки платформы **ProTalent**, предназначенной для взаимодействия заказчиков и исполнителей. Платформа предоставляет заказчикам возможность находить специалистов для выполнения различных задач, а фрилансерам — искать и брать в работу подходящие заказы.

## 1.3 Definitions, Acronyms and Abbreviations

*[Укажите значение терминов и аббревиатур, которые употребляются в данном документе. Возможно указание ссылки на Глоссарий проекта.]*

* **API** — Application Programming Interface (интерфейс программного приложения).
* **ProTalent** - название разрабатываемого продукта.

## 

## 

## 

## 

## 1.4 Reference

*[Перечислите списком названия документов, на которые ссылаетесь в данном, укажите их источники.]*

* Документация Spring Framework

https://docs.spring.io/spring-framework/reference/index.html

* Документация React.

https://react.dev/learn

* Документация PostgreSQL.

https://postgrespro.ru/docs/postgresql

## 1.5 Overview

*[Приведите краткое описание остальных разделов документа.]*

Документ включает описание общей структуры и характеристик системы (раздел 2), детализированные функциональные и нефункциональные требования (раздел 3), а также сценарии взаимодействия пользователей с системой и примеры использования (раздел 4), которые демонстрируют возможные действия пользователей на платформе.

# 2. Overall Description

*[Данный раздел содержит описание факторов, влияющих на требования к продукту, сами требования отписываются в следующем разделе.]*

Данный раздел описывает ключевые факторы, определяющие требования к системе **ProTalent**. Включены характеристики целевой аудитории, основные функциональные задачи, а также внешние зависимости и ограничения, которые необходимо учитывать при разработке. Эти аспекты помогают сформировать общее понимание системы, ее назначения и условий эксплуатации. Раздел также освещает ключевые взаимодействия пользователей с системой и основные технические особенности, влияющие на дизайн и реализацию платформы. Подробное описание всех требований к функциональности, производительности, надежности и удобству использования представлено в следующем разделе.

## 2.1 Product functions

*[Опишите основной функционал разрабатываемой системы, что она должна уметь делать****.****]*

Основные функции системы:

* **Регистрация и авторизация пользователей через email:** Пользователи смогут активировать учетную запись через подтверждение email и использовать двухфакторную аутентификацию для повышения безопасности.
* **Управление профилем**: пользователи (как заказчики, так и фрилансеры) могут редактировать свои профили, загружать фото, добавлять информацию о себе, указывать навыки и опыт работы
* **Поиск фрилансеров и проектов:** заказчики могут искать фрилансеров по ключевым навыкам, опыту работы, рейтингу и ценам. Фрилансеры могут искать проекты по категориям, ключевым словам, срокам выполнения и бюджету.
* **Система создания и размещения заказов**: позволяет заказчикам публиковать задания для фрилансеров, описывая все требования к проекту, сроки выполнения и бюджет.
* **Фильтрация и сортировка**: система предоставляет возможность использовать различные фильтры и сортировку для упрощения поиска проектов и фрилансеров (по рейтингу, цене, дате публикации и т.д.).
* **Чат для общения:** встроенная система обмена сообщениями между заказчиками и фрилансерами для обсуждения деталей проектов.
* **Оплата:** система оплаты на платформе организована таким образом, чтобы обеспечить безопасность и гарантировать, что ни заказчик, ни фрилансер не будут обмануты в процессе выполнения проекта.
* **Система рейтингов и отзывов**: по завершении проекта заказчики могут оставлять отзывы о фрилансерах, а фрилансеры могут оценивать заказчиков. Эти оценки влияют на общий рейтинг пользователей на платформе.
* **Система споров и арбитража**: в случае возникновения конфликтов между заказчиками и фрилансерами система предоставляет механизм подачи и разрешения споров с участием администраторов.

## 2.2 User characteristics

*[Опишите группы пользователей разрабатываемой системы, как именно они будут взаимодействовать с ней.]*

Основные типы пользователей системы:

* **Заказчики:** пользователи, публикующие проекты
* **Фрилансеры:** пользователи, предлагающие услуги
* **Администраторы:** пользователи, управляющие платформой, занимающиеся модерацией и поддержанием порядка на платформе. Они будут иметь доступ к управлению пользователями и проектами.

## 2.3 Assumptions and dependencies

*[Укажите дополнительные зависимости, которые могут повлиять на требования к системе.]*

Возможные внешние зависимости:

* Зависимость от стабильного интернет-соединения.
* Зависимость от сервисов, принимающих оплату.
* Наличие серверного окружения для развертывания и хранения данных.
* Необходимость обрабатывать персональные данные пользователей в соответствии с правовыми требованиями.

## 2.4 Constraints

*[Укажите ограничения, накладываемые на функционал системы.]*

* Ограничение на количество одновременно активных пользователей.
* Совместимость с последними версиями популярных браузеров (Chrome, Firefox, Safari).
* Платформа должна корректно работать на мобильных устройствах с операционными системами iOS и Android, обеспечивая адаптивный интерфейс и полноценную функциональность.

# 3. Specific Requirements

*[Данный раздел содержит описание всех требований к разрабатываемой системе. Данное описание будет использоваться как разработчиками при разработке системы, так и тестировщиками в процессе проверки её функционала.]*

В этом разделе представлен полный перечень требований к системе **ProTalent**. Эти требования будут использоваться разработчиками для создания функциональных компонентов системы и тестировщиками для проверки её корректности и соответствия заявленным функциональным и нефункциональным характеристикам. Раздел охватывает функциональные, пользовательские и технические аспекты, которые необходимы для успешной разработки, тестирования и внедрения системы.

## **3.**1 **Functionality**

*[Данный раздел содержит описание функциональных требований к системе.]*

#### 3.1.1 Регистрация и аутентификация:

Система должна поддерживать регистрацию пользователей через email.

* + Приоритет: Must Have
  + Трудоемкость: 10 попугаев
  + Риски: Нет.

**3.1.2 Восстановление пароля:**

Система должна предоставлять пользователям возможность восстановить пароль через функционал “Забыли пароль?”.

* + Приоритет: Must Have
  + Трудоемкость: 5 попугаев
  + Риски: Нет.

**3.1.3 Управление профилем:**

Система должна позволять пользователям (заказчикам и фрилансерам) создавать и редактировать свои профили, включая добавление фото, описания, портфолио и навыков.

* + Приоритет: Must Have
  + Трудоемкость: 10 попугаев
  + Риски: Нет.

Система должна предоставлять возможность фрилансерам указывать расценки на свои услуги (почасовая ставка или фиксированная стоимость за проект).

* + Приоритет: Must Have
  + Трудоемкость: 4 попугая
  + Риски: Нет.

**3.1.4 Поиск проектов и фрилансеров (фильтрация и сортировка):**

Система должна позволять заказчикам искать фрилансеров по ключевым навыкам, рейтингу, опыту и ценам, а также позволять сортировать по тому или иному показателю.

* + Приоритет: Must Have
  + Трудоемкость: 5 попугаев
  + Риски: Нет.

Система должна предоставлять фрилансерам возможность искать проекты по категориям, ключевым словам, срокам выполнения и бюджету, а также сортировать по срокам и бюджету.

* + Приоритет: Must Have
  + Трудоемкость: 5 попугаев
  + Риски: Нет.

**3.1.5 Создание и управление заказами:**

Система должна позволять заказчикам создавать заказы, указывая описание задачи, сроки выполнения и бюджет.

* + Приоритет: Must Have
  + Трудоемкость: 10 попугаев
  + Риски: Нет.

Система должна предоставлять заказчикам возможность управлять активными проектами, изменять их параметры или закрывать заказы по завершению работ.

* + Приоритет: Must Have
  + Трудоемкость: 5 попугаев
  + Риски: Нет.

**3.1.6 Загрузка файлов:**

Система должна позволять заказчикам прикреплять к проекту файлы (например, технические задания, примеры работ) для детального объяснения задания фрилансеру.

* + Приоритет: Must Have
  + Трудоемкость: 3 попугая
  + Риски: Нет.

**3.1.7 Подача заявок:**

Система должна позволять фрилансерам подавать заявки на проекты с указанием стоимости и сроков выполнения.

* + Приоритет: Must Have
  + Трудоемкость: 2 попугая
  + Риски: Нет.

**3.1.8 Система арбитража:**

Система должна предоставлять механизм разрешения споров, при котором администрация платформы может выступать арбитром в случае конфликта между заказчиком и фрилансером.

* + Приоритет: Must Have
  + Трудоемкость: 10 попугаев
  + Риски: Нет.

**3.1.9 Отзывы и рейтинг:**

Система должна позволять заказчикам оставлять отзывы о фрилансерах после завершения проекта, и фрилансерам — оставлять отзывы о заказчиках.

* + Приоритет: Must Have
  + Трудоемкость: 5 попугаев
  + Риски: Нет.

**3.1.10 Встроенный чат:**

Система должна предоставлять встроенный чат для общения заказчиков и фрилансеров в рамках обсуждения деталей проекта.

* + Приоритет: Must Have
  + Трудоемкость: 15 попугаев
  + Риски: Нет.

**3.1.11 Профиль администратора:**

Система должна предоставлять администратору возможность управлять пользователями (заблокировать, разблокировать), проектами (удалить, убрать из выдачи для доработки заказчиком).

* + Приоритет: Must Have
  + Трудоемкость: 3 попугая
  + Риски: Нет.

**3.1.12 Оплата и эскро:**

Система должна поддерживать внесение средств на эскроу-счет заказчика перед началом выполнения работы для обеспечения безопасности обеих сторон.

* + Приоритет: Must Have
  + Трудоемкость: 20 попугаев
  + Риски: трудно спрогнозировать риски из-за отсутствия опыта работы с платежными системами.

Система должна переводить средства фрилансеру только после подтверждения заказчиком успешного завершения работы или в случае решения спора администратором в пользу фрилансера.

* + Приоритет: Must Have
  + Трудоемкость: 10 попугаев
  + Риски: Нет.

## **3.**2 **Usability**

*[Данный раздел содержит требования к удобству использования системы. Например, время обучения обычного и опытного пользователя, среднее время выполнения типовых задач и т.д.]*

#### 3.2.1 Интуитивный интерфейс:

#### Система должна обеспечивать простой и интуитивно понятный интерфейс, позволяющий пользователям воспользоваться основным функционалом в 2-3 клика.

* + Приоритет: Must Have
  + Трудоемкость: 25 попугаев
  + Риски: Нет.

#### 

#### 

#### 3.2.2 Навигационные элементы:

Навигационные элементы должны быть доступны из любой точки платформы для быстрого перехода между разделами.

* + Приоритет: Must Have
  + Трудоемкость: 5 попугаев
  + Риски: Нет.

#### 3.2.3 Совместимость с устройствами:

Система должна быть адаптирована для использования как на десктопах, так и на мобильных устройствах, с корректным отображением интерфейса на экранах любых размеров.

* + Приоритет: Must Have
  + Трудоемкость: 10 попугаев
  + Риски: Нет.

## **3.**3 **Reliability**

*[Данный раздел содержит требования к надежности системы. Например, её доступность (в %), среднее время между возникновением ошибок, среднее время восстановления работоспособности, точность, количество найденных критических ошибок и т.д.]*

**3.3.1 Доступность системы:**

Доступность системы должна быть не менее 99% времени работы.

* + Приоритет: Must Have
  + Трудоемкость: 15 попугаев
  + Риски: Нет.

## 

## **3.**4 **Performance**

*[Данный раздел содержит требования к производительности системы. Например, время ответа (максимальное и среднее), максимальное количество обрабатываемых транзакций в секунду, максимальное количество одновременно работающих пользователей, использование ресурсов (память, дисковое пространство) и т.д.]*

#### 3.4.1 Обработка транзакций

Система должна обеспечивать возможность обработки не менее 100 транзакций в минуту (платежные операции, создание проектов, подача заявок и т.д.) без потерь производительности.

* + Приоритет: Must Have
  + Трудоемкость: 20 попугаев
  + Риски: есть вероятность, что сервер (helios) будет не способен поддерживать данное требование, возможно придется перейти на другой хост..

#### 3.4.2 Количество одновременно работающих пользователей

Система должна поддерживать одновременную работу не менее 100 активных пользователей, без значительного падения производительности и времени отклика.

* + Приоритет: Must Have
  + Трудоемкость: 20 попугаев
  + **Риски: плохая архитектура на ранних этапах может сорвать сроки.**

## 

## 

## **3.**5 **Design Constraints**

*[Данный раздел содержит все требования к процессу разработки. Например, используемый язык программирования, требования к процессу разработки (методологии), выбранные инструменты разработки, использование сторонних библиотек и т.д.]*

* Система должна быть разработана с использованием Spring Boot для backend части.
* Клиентская часть должна быть разработана на React.
* В качестве базы данных используется PostgreSQL.

## **3.**6 **Interfaces**

*[Данный раздел описывает интерфейсы, которые должна поддерживать система].*

## 3.6.1 User Interfaces (Пользовательские интерфейсы)

*[Описание пользовательского интерфейса системы.]*

Пользовательский интерфейс системы **ProTalent** должен быть простым, интуитивно понятным и адаптированным для различных категорий пользователей: неавторизированные пользователи, фрилансеры, заказчики, администраторы. Основные требования к пользовательскому интерфейсу:

* **Адаптивный дизайн: Интерфейс должен быть доступен на настольных компьютерах, планшетах и мобильных устройствах, с поддержкой различных размеров экранов. Все элементы интерфейса должны масштабироваться без потери функциональности.**
* **Регистрация и авторизация**: Форма регистрации должна **быть простой** и поддерживать ввод email и пароля, с подтверждением через письмо. Интерфейс для авторизации должен обеспечивать легкий вход с минимальными шагами.
* **Личный кабинет**: Для фрилансеров и заказчиков должна быть предусмотрена отдельная панель управления, где они могут управлять своими профилями, проектами и заявками.
* **Панель администратора**: Администраторы должны иметь доступ к специальному интерфейсу для управления пользователями, проектами и отзывами.
* **Удобная навигация**: Интерфейс должен иметь продуманную навигацию, чтобы пользователи могли легко находить нужные функции: создание проектов, просмотр заявок, фильтрацию проектов по категориям и т.д.
* **Чат для общения**: Встроенная система сообщений должна быть доступна в интерфейсе как заказчикам, так и фрилансерам для быстрого общения по проекту.

**3.6.2 Hardware Interfaces (Аппаратные интерфейсы)**

*[Описание аппаратных интерфейсов системы - физические адреса и т.д.]*

Система **ProTalent** не требует специализированных аппаратных интерфейсов, однако должна поддерживать взаимодействие с распространёнными устройствами пользователей и серверной инфраструктурой:

**Серверное оборудование**: Платформа должна быть развернута на серверном оборудовании, поддерживающем современные процессоры, достаточный объём оперативной памяти и хранилище данных (SSD).

**Поддержка клиентских устройств**: Система должна корректно работать на стандартных пользовательских устройствах:

* Настольные компьютеры (Windows, macOS, Linux).
* Мобильные устройства (iOS, Android).
* Планшеты.

**3.6.3 Software Interfaces (Программные интерфейсы)**

*[Описание интерфейсов, связывающих данную систему с другими приложениями.]*

Программные интерфейсы определяют, как система **ProTalent** взаимодействует с другими приложениями и сервисами:

* **API для интеграции с платёжными системами**: Система должна поддерживать RESTful API для интеграции с платёжными шлюзами (например, PayPal, Stripe) для обеспечения проведения транзакций между заказчиками и фрилансерами.
* **API для авторизации**: Платформа должна иметь возможность интеграции с другими сервисами для восстановления пароля и отправки уведомлений на почту через внешние сервисы.
* **Интерфейсы для обмена файлами**: Система должна поддерживать интеграцию с облачными сервисами хранения данных (Google Drive, Dropbox) для обмена файлами между заказчиками и фрилансерами.
* **Интерфейсы безопасности**: Должны быть реализованы программные интерфейсы для шифрования данных пользователей, включая личные данные и пароли, с использованием современных стандартов, таких как SSL/TLS.

**3.6.4 Communications Interfaces (Сетевые интерфейсы)**

*[Описание сетевых интерфейсов, которые используются системой для связи с другими устройствами.]*

Сетевые интерфейсы обеспечивают связь системы с внешними устройствами и сервисами:

* **Протоколы связи**: Система должна поддерживать стандартные сетевые протоколы, такие как TCP/IP и HTTP, для передачи данных через сеть.
* **Интерфейс передачи данных с платёжными системами**: Для обеспечения безопасности при обработке финансовых транзакций платформа должна взаимодействовать с платёжными шлюзами через защищённые каналы, поддерживающие шифрование данных.
* **Управление сессиями пользователей**: Система должна поддерживать сессии пользователей через защищённые соединения с использованием токенов (например, JWT) для аутентификации и авторизации.
* **Поддержка веб-сокетов**: Для обеспечения реального времени общения между пользователями (например, через встроенный чат) платформа должна поддерживать веб-сокеты или аналогичные технологии для быстрого обмена сообщениями.

## 

## 

## 

## 

## **3.**7 **Licensing Requirements**

*[Данные раздел описывает, по какой лицензии следует распространять разрабатываемый продукт].*

Продукт **ProTalent** является личным проектом и предназначен для использования исключительно его владельцем. Лицензионные требования определяют права на использование и распространение системы:

* **Личное использование**: Продукт является частным проектом, и его использование допускается только владельцем без каких-либо ограничений. Использование третьими лицами возможно только после покупки соответствующей лицензии или предоставления разрешения владельцем.
* **Коммерческое использование**: Вся интеллектуальная собственность, связанная с проектом, принадлежит владельцу сайта. Продажа лицензий на использование системы, предоставление доступа или услуг на её основе регулируется исключительно владельцем.
* **Запрещённое копирование и распространение**: Любое копирование, изменение, распространение или использование системы **ProTalent** без разрешения владельца строго запрещено. Нарушение данных условий может привести к правовым последствиям.

.

# 4. Use cases

# 

| **Прецедент:** Регистрация нового пользователя через почту |
| --- |
| **ID:** 1 |
| **Краткое описание:** Новый пользователь регистрируется на платформе ProTalent через email, заполняя форму регистрации с указанием адреса электронной почты и пароля. |
| **Главные актеры**: Посетитель сайта (пользователь без регистрации). |
| **Второстепенные актеры**: Нет. |
| **Предусловия**: Посетитель должен иметь доступ к интернету и действующий адрес электронной почты. |
| Основной поток:  1. Пользователь переходит на страницу "Регистрация". 2. Вводит адрес электронной почты и придумывает пароль. 3. Нажимает кнопку "Зарегистрироваться". 4. Система отправляет письмо с подтверждением на указанный email. 5. Пользователь переходит по ссылке из письма для подтверждения аккаунта. 6. Система подтверждает активацию аккаунта и предоставляет доступ к платформе. 7. Пользователь авторизуется в системе. |
| **Постусловие:** Пользователь успешно зарегистрирован и может использовать функции платформы (создание проектов, отклики на проекты и т.д.). |
| **Альтернативный поток:**  Пользователь не заполнил обязательные поля (например, email или пароль).  Система отображает сообщение с указанием на обязательные поля, которые необходимо заполнить.  Пользователь вводит недостающие данные и повторяет шаги основного потока. |
| **Точка отклонения:** Пользователь не заполнил обязательные поля или ввёл недействительный email, и система не позволяет завершить регистрацию. |

# 

| **Прецедент:** Подача заявки на проект фрилансером |
| --- |
| **ID:** 2 |
| **Краткое описание**: Фрилансер находит проект на платформе ProTalent и подаёт заявку с предложением по цене и срокам выполнения. |
| **Главные актеры**: Пользователь с ролью "фрилансер". |
| **Второстепенные актеры**: Нет. |
| **Предусловия**: Фрилансер зарегистрирован и авторизован в системе. У него есть доступ к активным проектам на платформе. |
| Основной поток:  1. Фрилансер авторизуется в системе. 2. Переходит в раздел "Поиск проектов". 3. Выбирает подходящий проект. 4. Нажимает кнопку "Подать заявку". 5. Заполняет поля с предложением по цене и срокам выполнения. 6. Пишет сообщение заказчику (опционально). 7. Нажимает кнопку "Отправить заявку". 8. Система сохраняет заявку и уведомляет заказчика о новом предложении. |
| **Постусловие:** Заявка успешно подана, заказчик получает уведомление и может рассмотреть её. |
| **Альтернативный поток:**   1. Фрилансер не заполнил обязательные поля в заявке. 2. Система отображает сообщение с указанием на обязательные поля. 3. Фрилансер вводит недостающие данные и повторяет шаг 7 основного потока. |
| **Точка отклонения:** Фрилансер не заполнил обязательные поля и не может подать заявку на проект. |

| **Прецедент:** Управление спором между заказчиком и фрилансером |
| --- |
| **ID:** 3 |
| **Краткое описание**: Администратор платформы ProTalent вмешивается в спор между заказчиком и фрилансером по поводу качества или сроков выполнения работы и принимает решение о разрешении спора. |
| **Главные актеры**: Администратор |
| **Второстепенные актеры**: Заказчик, Фрилансер. |
| **Предусловия**: Проект между заказчиком и фрилансером находится в стадии спора, стороны не могут прийти к соглашению, и один из участников подал жалобу в администрацию. |
| **Основной поток:**   1. Заказчик или фрилансер инициирует спор, заполняя форму с жалобой на платформе (например, несоответствие выполненной работы требованиям или несоблюдение сроков). 2. Система уведомляет администратора о возникшем споре и предоставляет ему доступ к проекту, сообщениям, и загруженным файлам. 3. Администратор входит в систему и переходит в раздел "Управление спорами". 4. Администратор просматривает детали проекта и анализирует жалобу (включает общение через встроенный чат с обеими сторонами для получения дополнительных разъяснений). 5. Администратор принимает решение: одобряет запрос заказчика (например, возврат денег) или отклоняет его в пользу фрилансера (производится оплата). 6. Администратор фиксирует своё решение в системе, и стороны получают уведомление о его итогах. 7. Система выполняет необходимые действия: перевод средств фрилансеру или возврат средств заказчику. |
| **Постусловие**: Спор разрешён, средства перераспределены согласно решению администратора. История спора сохраняется в базе данных платформы для дальнейшего анализа. |
| **Альтернативный поток:**   1. Администратор не может принять решение на основе предоставленной информации и запрашивает у сторон дополнительные материалы (например, корректировки или отчёты). 2. Спор временно приостанавливается до получения всех необходимых данных. 3. После получения данных основной поток возобновляется с шага 4. |
| **Точка отклонения:** Спор не может быть разрешён на основании имеющихся данных, и администратор принимает решение об удалении проекта с платформы или приостанавливает спор для дальнейшего рассмотрения. |