Курсовая работа

Система управления проектами

Шабалина Дарья Евгеньевна

22.05.2023

**СОДЕРЖАНИЕ**

[1 ПЛАН АНАЛИТИЧЕСКИХ РАБОТ 5](#_c0pnmp3ew2jg)

[1.1 Тема курсовой работы 5](#_cte9voc22kbg)

[1.2 Цели курсовой работы 5](#_vdjs7aqiip4t)

[1.3 Этапы выполнения проекта 5](#_birxs16ptn6a)

[1.4 Дорожная карта проекта 6](#_y77fys7btz34)

[2 КОНЦЕПЦИЯ И ГРАНИЦЫ 8](#_za8gl1ul7igq)

[2.1 Исходные данные 8](#_74533k6l71lq)

[2.1.1 Описание организации, для которой будет выполнена разработка системы 8](#_e7pgey4szrki)

[2.1.2 Предпосылки, обоснование реализации продукта или изменений 8](#_a23inygm5utt)

[2.1.3 Сравнительная оценка существующих продуктов и возможных решений 8](#_jvwqrrhlc2wg)

[2.2 Бизнес-цели и критерии успеха 10](#_ieqtaz5mbsnx)

[2.2.1 Бизнес-цели 10](#_o7922txfgq6q)

[2.2.2 Критерии успеха 11](#_67ls8in6tr9x)

[2.3 Концепция продукта и основные функции 11](#_3wctdoq4ywl7)

[2.3.1 Покупатель продукта 11](#_6l8wuagrgqef)

[2.3.2 Пользователи продукта 11](#_cxtd69thu58n)

[2.3.3 Ценность продукта 12](#_rh6l8srz2sre)

[2.3.4 Конкурентные преимущества 12](#_jqojs4d96ga4)

[2.3.5 Основные функции продукта 12](#_t78ac5gnm81t)

[2.4 Бизнес-риски, предположения и зависимости 13](#_7kmlbd72zyct)

[2.4.1 Риски 13](#_kuqbxivg5qo5)

[2.4.2 Зависимости 15](#_19zlr3e0myqq)

[2.4.3 Допущения, на основании которых построена концепция продукта 15](#_z9e65mwbkja2)

[2.4.4 Открытые вопросы, гипотезы, требующие уточнения в процессе реализации 15](#_9ar9fx5noqtk)

[2.5 Рамки и ограничения проекта 16](#_yufexpy51adu)

[2.5.1 Приоритет основных функций продукта и объем первоначальной и последующих версий 16](#_coq9ulk7xw0v)

[2.5.2 Ограничения и исключения 17](#_2j6a9c219zgp)

[3 ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ 18](#_lruhomnojohr)

[3.1 Приоритеты 18](#_7vu14dt5son7)

[3.2 Требования 18](#_10zt2u2eugue)

[3.2.1 ФТ-1 Авторизация пользователей 18](#_6s30mec3etby)

[3.2.2 ФТ-2 Создание проекта 19](#_pvlh98hp3icg)

[3.2.3 ФТ-3 Просмотр списка задач 19](#_vgg63k232ubb)

[3.2.4 ФТ-4 Создание задачи 20](#_od5spy1f9i86)

[3.2.5 ФТ-5 Просмотр задачи 21](#_mrlx8618cr8d)

[3.2.6 ФТ-6 Редактирование задачи 22](#_34gtubexdbyz)

[3.2.7 ФТ-7 Удаление задачи 23](#_707x8e95zard)

[3.2.8 ФТ-8 Поиск задачи 23](#_y4saz0d7i2kd)

[3.2.9 ФТ-9 Добавление комментариев 24](#_ddez4zvtnj3s)

[3.2.10 ФТ-10 Редактирование комментариев 25](#_fzpbcfy3mm8t)

[4 НЕФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ 26](#_7sbxy0pnrxpz)

[4.1 Приоритеты 26](#_2txsx1xv5h9g)

[4.2 Требования 26](#_omirc2oeayra)

[4.2.1 НФТ1 - Создание учетной записи 26](#_foysirkmpx8b)

[4.2.2 НФТ2 - Аутентификация 26](#_6t9rzb4uanm2)

[4.2.3 НФТ3 - Авторизация 27](#_dmnlqiwjanr6)

[4.2.4 НФТ4 - Взаимодействие компонентов 27](#_9z2bdnub42gr)

[4.2.5 НФТ5 - Регистрация событий безопасности 28](#_x2ds8gdt1q23)

[4.2.6 НФТ6 - Доступ к журналу регистрации событий безопасности 28](#_n02lt5rlfris)

[4.2.7 НФТ7 - Установка пароля пользователем 29](#_1qxy7kfeh83l)

[4.2.8 НФТ8 - Смена пароля пользователем при первичной аутентификации 29](#_sb87zo73juva)

[4.2.9 НФТ9 - Принудительная смена пароля 30](#_d3ke5inbd2cu)

[4.2.10 НФТ10 - Ограничение пароля 30](#_mqerzga52e9l)

[4.2.11 НФТ11 - Доступность системы 31](#_a5016pck6q5q)

[4.2.12 НФТ12 - Язык программирования 31](#_dsj4u12uqj70)

[4.2.13 НФТ13 - Логирование системы 32](#_gjlx1y9dnyan)

[4.2.14 НФТ14 - Темная тема 32](#_dvfb71a8bygc)

[4.2.15 НФТ15 - Время отклика загрузки страниц 33](#_r9epsdies0ay)

[4.2.16 НФТ16 - Время отклика загрузки отчетов 33](#_ajqstneuwqw0)

[4.2.17 НФТ17 - Персональные данные 33](#_xmuwemhl5vq)

[4.2.18 НФТ18 - Кроссбраузерность 34](#_5mwa5pgdrczb)

[4.2.19 НФТ19 - Поддерживаемые языки 34](#_iolyd98qwf1f)

[4.2.20 НФТ20 - Время и дата 36](#_voxbfmk1qsn3)

[5 ДИАГРАММЫ К КУРСОВОМУ ПРОЕКТУ 37](#_jsvm8suvjh9q)

[5.1 Контекстная диаграмма 37](#_um9lshrkrq4v)

[5.2 Модель сущность-связь 37](#_l5kiyckv44f)

[5.3 Описание бизнес-процессов (BPMN диаграммы) 38](#_z01osahzl7kl)

[5.3.1 Описание процесса создания нового функционала 38](#_nfxcamqv0ib0)

[5.3.2 Описание процесса работы с ошибками 39](#_yvny2b9f2zi3)

[5.4 Сценарии использования(MVP) 40](#_ou2sf2pde5r8)

[6 API 41](#_m636xuvvqofd)

[6.1 Экраны 41](#_je1c2kxvd03w)

[6.1 API 41](#_4v3m2csleh2p)

[6.2 Проектирование асинхронного взаимодействия 42](#_d2hvn7t2nksq)

[6.2.1 Описание функциональности 42](#_jz3v6q88hmq1)

[6.2.2 ФТ-11 отчет по исполнителям 43](#_qqp580p9ht98)

[6.2.3 ФТ-12 отчет по трудозатратам 45](#_360d0hjo034)

[7 ПРОЕКТИРОВАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ 48](#_rysabyfzop4i)

[7.1 Описание сущностей 48](#_jwbp0jxglbum)

[7.2 Концептуальная модель 55](#_lk8kvx4x18qr)

[7.3 Логическая модель 56](#_ey8ku7lhabkb)

[7.4 Физическое представление модели 56](#_iwuw2d30htgk)

[8 Практика SQL 57](#_usxmgpxh4xux)

[8.1 Инструменты 57](#_2sv1l25jhl3y)

[8.2 Скрипт на создание таблиц 57](#_z9yoaco9htkp)

[8.3 Скрипт на наполнение бд данными 60](#_kzk1m25yg1jv)

[8.4 Скрипт на получение данных 63](#_4z61txec8hg2)

[8.5 Скрипт на изменение данных 63](#_jhl8iao6v3sq)

# 

# **1 ПЛАН АНАЛИТИЧЕСКИХ РАБОТ**

## **1.1 Тема курсовой работы**

Система управления проектами.

## **1.2 Цели курсовой работы**

1. Создать спецификацию требований программного обеспечения к разрабатываемому продукту.
2. Провести конкурентный анализ рынка для создания более эффективного решения.
3. Декомпозировать проект на конкретные задачи, которые будут понятны разработчикам.

## 1.3 Этапы выполнения проекта

1. Постановка цели.
2. Подготовка плана аналитических работ.
3. Создание документа о концепциях и границах.
4. Выявление, сбор, анализ и документирование требований.
5. Моделирование системы.
6. Разработка прототипа программного интерфейса.
7. Проектирование верхнеуровневой архитектуры.
8. Проектирование модели данных.
9. Проектирование и документирование API.
10. Создание презентации и подготовка к защите.
11. Защита курсовой работы.

## 

## **1.4 Дорожная карта проекта**

| **Этапы работ** | **Февраль** | | | | **Март** | | | | | | | | | **Апрель** | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  | |  | |  | |  | |  |  |
| **Постановка цели** | 13.02 - 19.02  (1 неделя) | |  | |  |  | | | |  | |  | |  | |  |  |
|  |  |  |  |  |  |  |  |  | |  | |  | |  | |  |  |
| **Подготовка плана аналитических работ** |  |  | 20.02 - 05.03  (2 недели) | | | |  | | |  | |  | |  | |  |  |
|  |  |  |  |  |  |  |  |  | |  | |  | |  | |  |  |
| **Создание документа о концепциях и границах** |  |  |  | Изучение предметной области  27.02 - 05.03  (2 недели) | | |  |  | |  | |  | |  | | | |
|  |  |  |  |  | Сбор бизнес  требований от заказчика и подготовка документа  01.03 - 15.03  (2 недели) | | | |  | |  | |  | |  | |
|  |  |  |  |  |  | Анализ рисков  08.03  - 15.03  (1 неделя) | | |  | |  | |  | |  |  |
|  |  |  |  |  |  |  |  |  | |  | |  | |  | |  |  |
| **Выявление, сбор, анализ и документирование требований** |  |  |  |  |  | Выявление требований  01.03 - 22.03  (3 недели) | | | | | |  | |  | |  |  |
|  |  |  |  |  |  |  | Анализ требований  08.03 - 22.03  (2 недели) | | | |  | |  | |  |  |
|  |  |  |  |  |  |  | Документирование требований  10.03 - 31.03  (3 недели) | | | | | |  | |  |  |
|  |  |  |  |  |  |  |  |  | | | | | |  | |  |  |
| **Моделирование системы** |  |  |  |  |  |  |  |  | | | Моделирование процессов и системы с помощью различных нотаций(BPMN, UML)  22.03 - 06.04  (2 недели) | | | | | |  |
|  |  |  |  |  |  |  |  |  | |  | |  | |  | |  |  |

| **Этапы работ** | **Апрель** | | | | | | **Май** | | | | | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | | |  |  |  |  |  | | |  | | |  | | | | | | |
| **Разработка прототипа программного интерфейса** | 03.04 - 09.04  (1 неделя) | |  | | | |  |  | | | | | |  | | |  | | | | | | |
|  |  |  |  |  | | |  |  |  |  |  | | |  | | |  | | | | | | |
| **Проектирование верхнеуровневой архитектуры** |  |  | 10.04 - 23.04  (2 недели) | | |  |  |  |  | | | | |  | | |  | | | | | | |
|  |  |  |  |  | | |  |  |  |  |  | | |  | | |  | | | | | | |
| **Проектирование модели данных** |  |  |  |  | | Концептуальная  модель  24.04 - 07.05  (2 недели) | | | |  |  | | |  | | |  | | | | | | |
|  |  |  |  | | |  | Логическая модель  04.05 - 18.05  (2 недели) | | | | | |  | | |  | | | | | | |
|  |  |  |  | | |  |  |  | Физическое представление модели  11.05 - 18.05  (1 неделя) | | | |  | | |  | | | | | | |
|  |  |  |  |  | | |  |  |  |  |  | | |  | | |  | | | | | | |
| **Проектирование и документирование API** |  |  |  |  | | |  |  |  |  | Описание моделей и контрактов в yml в формате OpenAPI v3  15.05 - 22.05  (1 неделя) | | | | | | | | | | | |  |
|  | | | | | | | | | | | | | | | | | | | | | | | |
| **Создание презентации и подготовка к защите** |  |  |  |  | | |  |  |  |  |  | | | | 17.05 - 22.05 | | | | | | | |  |
|  | | | | | | | | | | | | | | | | | | | | | | | |
| **Защита курсовой работы**  **22.05** | | | | | | | | | | | | | | | | | | | | | | | |

# 

# **2 КОНЦЕПЦИЯ И ГРАНИЦЫ**

## **2.1 Исходные данные**

### **2.1.1 Описание организации, для которой будет выполнена разработка системы**

Компания «XXX». Специализируется на разработке программно-аппаратных комплексов, предназначенных для повышения безопасности, производительности и эффективности в горнодобывающей промышленности.

### 2.1.2 Предпосылки, обоснование реализации продукта или изменений

На данный момент для планирования и учета выполнения задач в рамках работы компании используется бесплатная версия «Redmine». Компания сталкивается со сложностями в работе над проектами из-за ограниченного функционала данной системы. В связи с расширением штата и увеличением количества задач возникла необходимость использования более современной и функциональной системы для управления проектами, которая позволит централизованно руководить задачами и их своевременным выполнением.

Также в связи с указом Президента РФ от 30 марта 2022 г. N 166 "О мерах по обеспечению технологической независимости и безопасности критической информационной инфраструктуры Российской Федерации” компания не может использовать популярные иностранные таск-трекинговые системы, а существующие российские аналоги не удовлетворяют предъявляемым требованиям в полном объеме.

В связи с этим компания «XXX» хочет в ближайшее время создать собственную систему управления проектами.

### 2.1.3 Сравнительная оценка существующих продуктов и возможных решений

В рамках анализа предметной области были выбраны следующие популярные иностранные системы управления проектами - Jira, Trello, Asana, также для сравнения были выбраны следующие российские аналоги Яндекс.Трекер и Турбо Трекинг.

Системы оценивались по следующим критериям:

1. **Тарифный план**. Заказчик не готов оплачивать подписку, в приоритете бесплатное использование. Для сравнения тарифных планов следует учитывать, что компания активно растет. На данный момент в ней работает 150+ людей.
2. **Возможность интеграции с другими системами**. Необходима гибкая настройка под индивидуальный стиль работы компании и возможность интеграции с инструментами, которые уже используют команды.
3. **Возможность работы и взаимодействия разных отделов**. Нужен гибкий инструмент, который подойдет для работы всем подразделениям в компании. Также необходимо, чтобы система имела поддержку русского языка.
4. **Исчерпывающая отчетность**. Для заказчика важно собирать данные по разным параметрам: по количеству затраченного на проект времени, по количеству успешно завершенных задач, по плановому и фактическому бюджетам и т.д.
5. **Гибкие права доступа**.
6. **Поддержка agile-процессов**.
7. **Пробный период**.

| Система управления проектами | **Jira** | **Trello** | **Asana** | **Яндекс.Трекер** | **Турбо Трекинг** |
| --- | --- | --- | --- | --- | --- |
| Тарифный план | Standard - $77.50 в месяц, Premium - $152.50 в месяц | Standard - $6 в месяц, Premium - $12.50 в месяц | Premium - $13.49 в месяц, Business - $30.49 в месяц | Для компаний от 150 до 250 пользователей - 222 ₽ за пользователя в месяц,  для компаний от 250 до 500 пользователей - 185 ₽ за пользователя в месяц | для компаний от 150 до 250 пользователей - 222 ₽ за пользователя в месяц,  для компаний от 250 до 500 пользователей - 185 ₽ за пользователя в месяц |

| Система управления проектами | **Jira** | **Trello** | **Asana** | **Яндекс.Трекер** | **Турбо Трекинг** |
| --- | --- | --- | --- | --- | --- |
| Интеграции с другими системами | + | + | + | +/-  есть возможность настройки интеграций с помощью API Tracker | ? |
| Возможность работы и взаимодействия разных отделов | - | + | + | + | + |
| Исчерпывающая отчетность | + | +/-  поддерживает интеграции для работы с аналитикой и отчетами | + | - | + |
| Гибкие права доступа | + | + | + | + | + |
| Поддержка agile-процессов | + | + | - | + | + |
| Пробный период | Standard - 14 дней, Premium - 30 дней | 14 дней | 30 дней | 2 месяца | - |

## 

## **2.2 Бизнес-цели и критерии успеха**

### 2.2.1 Бизнес-цели

1. **Структурировать и сделать прозрачной работу.** В таск-менеджере должна быть отражена стратегия и тактика компании, загружены все важные документы, описаны конкретные задачи, отмечены сотрудники, которые отвечают за каждый отдельный блок работы. Каждый член команды должен иметь возможность посмотреть, чем занимается коллега и четко представлять свой фронт работ.
2. **Спланировать работу и повысить вовлеченность персонала.** Каждый сотрудник и руководитель должен видеть сроки всех задач, их приоритет и прогресс по каждой из них.
3. **Повысить эффективность команды.** Каждый видит свои задачи и знает сроки их выполнения. Когда задача оказывается горящей, она подсвечивается, это замечает руководитель и может взять на контроль. В систему мотивации введено правило поощрения своевременной сдачи и санкций за то, что задача оказалась в «просроченных». Дополнительно таск-менеджер присылает ответственным напоминания и уведомления о коротком остатке жизни задачи.
4. **Перевести отдел разработки ПО на удаленную работу.** Благодаря прозрачности работы планируется перевести отдел разработки на удаленную работу. Это также позволит нанимать кандидатов из разных регионов РФ.
5. **Автоматизировать бизнес-процессы любых подразделений.**

### 2.2.2 Критерии успеха

1. 80% сотрудников ведут задачи в таск-менеджере.
2. 90% задач проходит через систему.
3. Через два месяца после внедрения MVP видна загрузка всех исполнителей и можно понять, какой именно отдел срывает сроки.
4. Через два месяца после внедрения MVP можно оценить сколько ресурсов уходит на выполнение любого проекта.

## **2.3 Концепция продукта и основные функции**

### 2.3.1 Покупатель продукта

Продукт разрабатывается для внутреннего использования.

### 2.3.2 Пользователи продукта

Сотрудники и заказчики компании «XXX».

### 2.3.3 Ценность продукта

Продукт поможет компании «XXX» увидеть уязвимые места в работе над проектами, оценить загрузку и эффективность сотрудников, работа команды станет прозрачной, что в свою очередь позволит повысить эффективность работы компании.

### 2.3.4 Конкурентные преимущества

1. Система "условно" бесплатная и будет поддерживаться усилиями разработчиков компании «XXX».
2. Продукт будет подстраиваться под процессы компании, а не процессы под продукт.
3. Планируется дать клиентам компании доступ к большинству информации. Это позволит обеспечить большую прозрачность процесса для существующих клиентов, которая поднимет доверие к компании и позволит улучшить коммуникацию. Также наличие такого мощного инструмента говорит о том, что внутренний процесс отлажен и предсказуем. А это поможет при общении с потенциальными заказчиками.

### 2.3.5 Основные функции продукта

1. Организация работ с проектами.
2. Поддержка системы доступа пользователей, основанная на ролях.
3. Организация работ с задачами.
4. Учёт временных затрат исполнителей.
5. Визуализация работ на диаграмме Ганта и календарях.
6. Организация обсуждений проблем, задач и т.п. между пользователями.
7. Учет документов и управление файлами.
8. Оповещение участников проекта об изменениях.
9. Организация управления версиями и выпусками продукта.
10. Формирование разнообразных отчетов для анализа.
11. Автоматическое резервное копирование.
12. Поддержка интеграции с сторонними сервисами.
13. Создание и поддержка базы знаний.
14. Поддержка Agile.
15. Создание дорожных карт проекта.

## **2.4 Бизнес-риски, предположения и зависимости**

### 2.4.1 Риски

1. Риск проблемы с внедрением ПО из-за консервативности сотрудников

| Описание риска | Сотрудники не хотят менять существующие процессы. Открыто критикуют новый подход. Внедрение нового ПО затягивается по времени. |
| --- | --- |
| Влияние на проект | Дополнительные временные трудозатраты. |
| Действия по предотвращению или снижению влияния | Все результаты должны обсуждаться не только с командой разработки, но и с другими коллегами, т.к. потребителем продукта будет являться практически каждый из них. Беседы и интервью должны начаться еще до проектирования и не останавливаться на всем протяжении проекта. |
| Индикаторы | В рамках процесса сбора отзывов более 15% сотрудников недовольны работой новой системы. |
| План устранения последствий | 1. Руководители команд различных направлений в деталях рассказывают о том, как у них построен процесс, что важно и что мешает в текущей версии. 2. Пересмотр требований. 3. Работа с сотрудниками. |

1. Риск проблемы с внедрением ПО из-за консервативности сотрудников

| Описание риска | Заказчик в MVP хочет видеть большое количество функций. Из-за того, что раньше сотрудники компании не работали по такой системе появляются проблемы с внедрением. |
| --- | --- |
| Влияние на проект | 1. Дополнительные затраты времени на перепланирование графика реализации проекта. 2. Парализована работа большого количества сотрудников. |
| Действия по предотвращению или снижению влияния | 1. В начальных версиях внедряются только базовые функции. 2. В дальнейшем в процессы внедряются по 1-3 новых возможностей за раз при условии, что в предыдущей версии всё работало идеально. 3. Пользовательская документация поддерживается в актуальном состоянии. |
| Индикаторы | В течении месяца после выдачи в релиз MVP менее 80% сотрудников ведут задачи в таск-менеджере и/или менее 90% задач проходит через систему. |
| План устранения последствий | 1. Нормализация выдачи версий. 2. Перенос реализации некоторых требований. 3. Организация обучения для сотрудников по работе с системой. |

1. Риск выделения ресурсов на проект в объеме, меньше чем планировалось

| Описание риска | Сотрудники компании будут тратить на проект не запланированные 30-40% от рабочего времени, а гораздо меньше. |
| --- | --- |
| Влияние на проект | Выделение ресурсов на проект в объеме, меньше чем планировалось, приведет к срыву сроков по задачам проекта. |
| Действия по предотвращению или снижению влияния | 1. Привнося изменения, необходимо сначала тщательно проанализировать, как это отразится на текущем состоянии проекта, сколько усилий потребует и существует ли риск задержки. 2. Создать резервы по времени. 3. Ежедневные митинги для обсуждения прогресса и выделения блокеров. |
| Индикаторы | Риск считается случившимся, если выдача версий происходит с опозданием на срок более, чем 1 неделя от запланированного графика. |
| План устранения последствий | 1. Пересмотр сроков и плана работ. 2. Отказ от реализации некоторых требований. 3. Реорганизация нагрузки. |

### 2.4.2 Зависимости

1. Конечный вид системы и ее функциональность зависеть от сотрудников компании «XXX».
2. Система будет разрабатываться с учетом особенностей бизнес-процессов компании «XXX».

### 2.4.3 Допущения, на основании которых построена концепция продукта

1. Сотрудники компании будут тратить на реализацию проекта 30-40% времени.
2. Проект будет разрабатываться с использованием гибкой методологии. Раз в три спринта будет выпускаться новая версия.

### 2.4.4 Открытые вопросы, гипотезы, требующие уточнения в процессе реализации

1. Все результаты должны обсуждаться не только с командой разработки, но и с другими коллегами, т.к. потребителем продукта будет являться практически каждый из них. Беседы и интервью должны начаться еще до проектирования и не останавливаться на всем протяжении проекта.
2. Программа должна иметь максимально низкий порог входа при широких возможностях. Среднестатистический сотрудник должен иметь возможность разобраться с основными функциями системы за 15-20 минут.

## **2.5** Рамки и ограничения проекта

### 2.5.1 Приоритет основных функций продукта и объем первоначальной и последующих версий

1. Приоритет основных функций будет оцениваться по следующим метрикам:
2. Должно быть сделано обязательно.
3. Было бы хорошо сделать.
4. Можно будет сделать.

В MVP войдут функции с приоритетом - 1, в версию v.2 функции с приоритетом 2, в версию v.3 функции с приоритетом - 3.

| Функционал | Версия |
| --- | --- |
| Организация работ с проектами | MVP |
| Поддержка системы доступа пользователей, основанная на ролях(администратор, остальные пользователи) | MVP |
| Организация работ с задачами | MVP |
| Учёт временных затрат исполнителей | MVP |
| Организация обсуждений проблем, задач и т.п. между пользователями | MVP |
| Формирование разнообразных отчетов для анализа(отчет по пользователям, отчет по трудозатратам) | MVP |
| Поддержка интеграции со сторонними сервисами(GitLab) | MVP |
| Поддержка системы доступа пользователей, основанная на ролях(более гибкая система) | v.2 |
| Формирование разнообразных отчетов для анализа(создание конструктора отчетов) | v.2 |
| Поддержка интеграции со сторонними сервисами(другие сервисы необходимые для работы компании) | v.2 |
| Автоматическое резервное копирование. | v.2 |
| Поддержка Agile | v.2 |
| Учет документов и управление файлами. | v.2 |
| Визуализация работ на диаграмме Ганта и календарях | v.3 |
| Оповещение участников проекта об изменениях | v.3 |
| Организация управления версиями и выпусками продукта | v.3 |
| Создание и поддержка базы знаний | v.3 |
| Создание дорожных карт проекта | v.3 |

### 2.5.2 Ограничения и исключения

1. В рамках проекта не планируется разрабатывать мобильное приложение и desktop версию.
2. Бюджет проекта ограничен 20 млн рублей.

## 

# **3 ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ**

## **3.1 Приоритеты**

Для каждого пункта функциональных требований определен приоритет, который может быть:

1. Критический.
2. Высокий.
3. Нормальный.
4. Низкий.

## 3.2 Требования

### 3.2.1 ФТ-1 Авторизация пользователей

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 16.03.2022 |
| Участники варианта использования | Неавторизованный пользователь системы. |
| Предусловие | Пользователь находится на странице авторизации. |
| Триггер | 1. Пользователь вводит логин и пароль в соответствующие формы.  2. Пользователь подтверждает ввод, нажатием на кнопку «Вход». |
| Основной сценарий | 1. Система проверяет введённые пользователем данные.  2. Если введенные пользователем данные верные, система разрешает доступ.  3. Пользователя перебрасывает на домашнюю страницу системы. |
| Альтернативный сценарий | 1. Если введенные пользователем данные неверные, система отказывает в доступе.  2. Система отображает сообщение:  «Неправильное имя пользователя или пароль». |
| Приоритет | Критический. |

### 

### 3.2.2 ФТ-2 Создание проекта

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 16.03.2022 |
| Участники варианта использования | Администратор системы. |
| Предусловие | 1. Пользователь авторизован в системе и имеет роль администратора.  2. Пользователь находится на странице «Проекты». |
| Триггер | Пользователь нажал на кнопку «Создать проект». |
| Основной сценарий | 1. Открылась панель создания проекта.  2. Пользователь заполняет все обязательные поля и, опционально, дополнительные.  3. Пользователь нажимает кнопку «Сохранить».  4. Проект создан и отображается в списке проектов. 6. Система закрывает панель создания проекта. |
| Альтернативный сценарий 1 | 1. Открылась панель создания проекта.  2. Пользователь заполняет все обязательные поля и, опционально, дополнительные.  3. Пользователь нажимает кнопку «Сохранить».  4. Проект создан и отображается в списке проектов. 6. Система закрывает панель создания проекта. |
| Альтернативный сценарий 2 | 1. Пользователь заполняет не все обязательные поля.  2. Пользователь нажимает кнопку «Сохранить».  3. Система выводит сообщение об ошибке. 4. Незаполненные обязательные поля подсвечиваются красным цветом. 5. Работа возобновляется с шага 2 основного сценария. |
| Приоритет | Критический. |

### 

### 3.2.3 ФТ-3 Просмотр списка задач

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 16.03.2022 |
| Участники варианта использования | Авторизованный пользователь системы. |
| Предусловие | 1. Пользователь авторизован в системе. 2. Открыта любая веб-страница системы. |
| Триггер | Пользователь нажал на кнопку «Моя страница», расположенную в горизонтальном верхнем меню, которое доступно на любой странице. |
| Основной сценарий | Произошел переход на страницу пользователя, где должны отображаться все его задачи в виде списка. По умолчанию система должна отображать: номер задачи, проект, название задачи, приоритет, готовность. |
| Приоритет | Критический. |

### 

### 3.2.4 ФТ-4 Создание задачи

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 16.03.2022 |
| Участники варианта использования | Авторизованный пользователь системы. |
| Предусловие | 1. Пользователь авторизован в системе. 2. Открыта любая веб-страница системы. |
| Триггер | Пользователь нажал на кнопку «Новая задача», расположенную в горизонтальном верхнем меню, которое доступно на любой странице. |
| Основной сценарий | 1. Открылась панель создания задачи.  2. Пользователь заполняет все обязательные поля и, опционально, дополнительные.  3. Пользователь нажимает кнопку «Сохранить».  4. Система проверяет, что у пользователя есть право на создание задачи в выбранном проекте. 5. Задача создана и отображается в списке задач пользователя, на которого она назначена. 6. Система закрывает панель создания задачи. |
| Альтернативный сценарий 1 | 1. Пользователь заполняет не все обязательные поля.  2. Пользователь нажимает кнопку «Сохранить».  3. Система выводит сообщение об ошибке. 4. Незаполненные обязательные поля подсвечиваются красным цветом. 5. Работа возобновляется с шага 2 основного сценария. |
| Альтернативный сценарий 2 | 1. Пользователь заполняет не все обязательные поля.  2. Пользователь нажимает кнопку «Сохранить».  3. Система выводит сообщение об ошибке. 4. Незаполненные обязательные поля подсвечиваются красным цветом. 5. Работа возобновляется с шага 2 основного сценария. |
| Исключения | 1. У пользователя нет прав на создание задачи в выбранном проекте.  2. Система выводит сообщение об ошибке . 3. Система закрывает панель создания задачи. |
| Приоритет | Критический. |

### 3.2.5 ФТ-5 Просмотр задачи

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 16.03.2022 |
| Участники варианта использования | Авторизованный пользователь системы. |
| Предусловие | 1. Пользователь авторизован в системе. 2. Открыта любая веб-страница системы, содержащая ссылку на задачу. |
| Триггер | Пользователь нажал на номер или тему задачи. |
| Основной сценарий | 1. Система проверяет, что у пользователя есть право на просмотр задачи. 2. Открылась страница задачи, содержащая всю информацию о ней, которую заполнили во время создания или редактирования задачи, а также комментарии. |
| Альтернативный сценарий | 1. Пользователь перешел по ссылке на задачу из стороннего источника, например, почта. 2. Система проверяет, что пользователь авторизован в системе. Если нет, то открывается страница авторизации и пользователь вводит необходимые данные. 3. Система проверяет, что у пользователя есть право на просмотр задачи. 4. Открылась страница задачи, содержащая всю информацию о ней, которую заполнили во время создания или редактирования задачи, а также комментарии. |
| Исключения | 1. У пользователя нет прав на просмотр веб-страницы с задачей.  2. Система отображает сообщение: «У Вас нет прав для посещения данной страницы». |
| Приоритет | Критический. |

### 3.2.6 ФТ-6 Редактирование задачи

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 16.03.2022 |
| Участники варианта использования | Авторизованный пользователь системы. |
| Предусловие | 1. Пользователь авторизован в системе. 2. Открыта веб-страница задачи. |
| Триггер | Пользователь нажал на кнопку «Редактировать». |
| Основной сценарий | 1. Открывается панель задачи в режиме редактирования.  2. Пользователь редактирует поля по выбору.  3. Пользователь нажимает кнопку «Сохранить». 4. Система закрывает панель редактирования задачи. 5. Пользователь видит задачу с внесенными им изменениями.  6. Все изменения записаны в журнал изменения задач. |
| Альтернативный сценарий 1 | 1. Открывается панель задачи в режиме редактирования.  2. Пользователь редактирует поля по выбору.  3. Пользователь нажимает кнопку «Сохранить». 4. Система закрывает панель редактирования задачи. 5. Пользователь видит задачу с внесенными им изменениями.  6. Все изменения записаны в журнал изменения задач. |
| Альтернативный сценарий 2 | 1. Пользователь отказывается от выполнения операции.  2. Пользователь нажимает на кнопку «Отмена».  3. Система закрывает панель редактирования задачи. |
| Альтернативный сценарий 3 | 1. Во время редактирования задачи другой пользователь уже ее отредактировал.  2. Система должна уведомить об этом пользователя и предложить несколько вариантов решения: 2.1 Отменить все внесенные изменения. 2.2 Применить изменения, которые внес другой пользователь и продолжить редактирование. 2.3 Применить изменения, которые внес другой пользователь, поверх его изменений применить свои.  3. Система обновляет атрибуты задачи в зависимости от выбранного варианта. 4. Работа возобновляется с шага 2 основного сценария. |
| Исключения | 1. У пользователя нет прав на редактирование задачи.  2. Система отображает сообщение: «У Вас нет прав для редактирования данной задачи». |
| Приоритет | Критический. |

### 

### 3.2.7 ФТ-7 Удаление задачи

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 17.03.2022 |
| Участники варианта использования | Администратор системы. |
| Предусловие | 1. Пользователь авторизован в системе и имеет роль администратора.  2. Открыта страница задачи. |
| Триггер | Пользователь нажал на кнопку «Удалить». |
| Основной сценарий | 1. Система показывает диалоговое окно «Вы уверены, что хотите удалить данную задачу?» 2. Пользователь нажимает кнопку «Да». 3. Из системы удалена задача, все комментарии к ней и вложения.  4. Система закрывает диалоговое окно. |
| Альтернативный сценарий | 1. Пользователь отказывается от выполнения операции.  2. Пользователь нажал на кнопку «Нет».  3. Система закрывает диалоговое окно. |
| Приоритет | Нормальный. |

### 

### 3.2.8 ФТ-8 Поиск задачи

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 18.03.2022 |
| Участники варианта использования | Авторизованный пользователь системы. |
| Предусловие | 1. Пользователь авторизован в системе. 2. Открыта любая веб-страница системы. |
| Триггер | Пользователь ввел данные в поле ввода текста, расположенное в горизонтальном верхнем меню рядом с заголовком «Поиск», которое доступно на любой странице. |
| Основной сценарий | 1. Пользователь вводит номер задачи. 2. Система проверяет, что задача с данным номером существует.  3. Происходит переход на страницу задачи. |
| Альтернативный сценарий | 1. Система не обнаружила задачу с введенным номером, либо у пользователя нет прав на просмотр веб-страницы задачи.  2. Происходит переход на страницу поиска, где можно добавить критерии поиска по проекту, пользователю, исполнителю, а также указать, где мы ищем введенные данные: в названии задачи, в комментариях к ней, в ее описании. 3. Пользователь выбирает необходимые критерии и нажимает на кнопку «Принять». 4. Система отображает найденные задачи доступные для просмотра пользователю в виде списка. |
| Приоритет | Высокий. |

### 

### 3.2.9 ФТ-9 Добавление комментариев

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 20.03.2022 |
| Участники варианта использования | Авторизованный пользователь системы. |
| Предусловие | 1. Пользователь авторизован в системе. 2. Открыта страница задачи. |
| Триггер | Пользователь нажал кнопку «Комментарий». |
| Основной сценарий | 1. Система отображает текстовое поле. 2. Пользователь вводит комментарий в появившемся поле. 3. Пользователь нажимает на кнопку «Сохранить». 4. Система проверяет, что у пользователя есть право на добавление комментария к задаче. 5. Система закрывает текстовое поле. 6. Созданный комментарий отображается на странице задачи. |
| Альтернативный сценарий | 1. Пользователь отказывается от выполнения операции.  2. Пользователь нажимает на кнопку «Отмена».  3. Текстовое поле закрывается. |
| Исключения | 1. У пользователя нет прав на добавление комментариев к задаче.  2. Система отображает сообщение: «У Вас нет прав для добавления комментариев к данной задаче». 3. Система закрывает текстовое поле. |
| Приоритет | Критический. |

### 3.2.10 ФТ-10 Редактирование комментариев

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 20.03.2022 |
| Участники варианта использования | Авторизованный пользователь системы. |
| Предусловие | 1. Пользователь авторизован в системе. 2. Открыта страница задачи. |
| Триггер | Пользователь нажал кнопку «Редактировать» справа от собственного комментария. |
| Основной сценарий | 1. Система отображает текстовое поле с уже существующим комментарием. 2. Пользователь редактирует информацию в появившемся поле.  3. Пользователь нажимает на кнопку «Сохранить». 4. Отредактированный комментарий отображается на странице задачи.  5. Система закрывает текстовое поле. |
| Альтернативный сценарий | 1. Пользователь отказывается от выполнения операции.  2. Пользователь нажимает на кнопку «Отмена».  3. Система закрывает текстовое поле. |
| Приоритет | Нормальный. |

# 

# **4 НЕФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ**

## 4.1 Приоритеты

Для каждого пункта функциональных требований определен приоритет, который может быть:

1. Критический.
2. Высокий.
3. Нормальный.
4. Низкий.

## 4.2 Требования

### 4.2.1 НФТ1 - Создание учетной записи

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | В системе обеспечивается создание уникальной учетной записи для каждого пользователя. |
| Обоснование | Необходима аутентификация и авторизация пользователей системы. Пользователи должны иметь доступ к личным данным. |
| Способ тестирования | Ручное/автоматическое тестирование специалистом QA. |
| Приоритет | Критический. |

### **4.2.2 НФТ2** - **Аутентификация**

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Механизмы аутентификации реализуются с использованием защищенных протоколов аутентификации. При хранении и передаче конфиденциальность паролей обеспечивается шифрованием или хешированием с применением стойких криптографических алгоритмов. |
| Обоснование | Необходимо ограничить доступ к системе сторонних пользователей. |
| Способ тестирования | Тестирование специалистом QA по OWASP testing guide. |
| Приоритет | Критический. |

### **4.**2**.3 НФТ3** - **Авторизация**

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Обеспечивается реализация механизмов авторизации пользователей с разделением прав доступа к информации/данным и функциям внутри системы. |
| Обоснование | Пользователи должны иметь ограниченный доступ к определенным функциям сайта в зависимости от их ролей и того, какие функции назначил им владелец или администратор системы. |
| Способ тестирования | Ручное/автоматическое тестирование специалистом QA. |
| Приоритет | Критический. |

### **4.**2**.4 НФТ4** - **Взаимодействие компонентов**

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Взаимодействие компонентов одной или нескольких ИС должны осуществляется с использованием защищенных протоколов передачи данных. |
| Обоснование | Система должна быть защищена от взлома и утечки данных. |
| Способ тестирования | Тестирование специалистом QA по OWASP testing guide. |
| Приоритет | Критический. |

### **4.**2**.5 НФТ5** - **Регистрация событий безопасности**

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | В системе обеспечивается возможность регистрации следующих событий безопасности (как минимум, но не ограничиваясь):   1. Факты или попытки идентификации и аутентификации субъектов доступа. 2. Факты изменения полномочий, модификация профиля пользователей. 3. Факты создания, изменения или блокирования учетных записей. 4. Действия привилегированных пользователей и администраторов по настройке и изменению конфигурации ИС (в том числе изменение настроек аудита). 5. Факты доступа к защищаемым объектам доступа (включая журналы регистрации событий и параметры конфигурирования). |
| Обоснование | Система должна быть защищена от взлома и утечки данных. |
| Способ тестирования | Тестирование специалистом QA. |
| Приоритет | Нормальный. |

### **4.**2**.6 НФТ6** - **Доступ к журналу регистрации событий безопасности**

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Обеспечивается ограничение доступа к журналам регистрации событий безопасности только уполномоченным пользователям. |
| Обоснование | Система должна быть защищена от взлома и утечки данных. |
| Способ тестирования | Тестирование специалистом QA. |
| Приоритет | Нормальный. |

### **4.2.7 НФТ7** -Установка пароля пользователем

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Пользователям обеспечивается возможность самостоятельной установки и смены пароля. |
| Обоснование | Пользователь должен иметь возможность установить или сменить пароль от своей учетной записи для предотвращения несанкционированного доступа к ней. |
| Способ тестирования | Тестирование специалистом QA. |
| Приоритет | Высокий. |

### **4.**2**.8 НФТ8** - **Смена пароля пользователем при первичной аутентификации**

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Обязательная смена пароля пользователем при первичной аутентификации в системе. |
| Обоснование | Регистрация новых пользователей будет осуществляться администраторами. Поэтому при при первичной аутентификации в системе пользователь должен сменить первичный пароль на собственный для предотвращения несанкционированного доступа к учетной записи. |
| Способ тестирования | Тестирование специалистом QA. |
| Приоритет | Критический. |

### **4.**2**.**9 **НФТ9** - **Принудительная смена пароля**

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Обеспечивается принудительная смена пароля через установленный промежуток времени. Время жизни ссылки по сбросу пароля равно 24 часа. |
| Обоснование | Аккаунт для пользователя системы может создать по запросу только пользователь с ролью администратора. В дальнейшем пользователь должен иметь возможность сменить пароль для защиты аккаунта. Принудительная смена нужна для исключения возможности утечки данных. |
| Способ тестирования | Тестирование специалистом QA. |
| Приоритет | Нормальный. |

### **4.**2**.10 НФТ10** - **Ограничение пароля**

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Обеспечивается возможность установки ограничений на длину и сложность пароля. |
| Обоснование | Система должна быть защищена от взлома и утечки данных. |
| Способ тестирования | Тестирование специалистом QA. |
| Приоритет | Низкий. |

### **4.2.11 НФТ11** - **Доступность системы**

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Система должна быть доступна 99.5% времени. Недоступность 1.83 дня в год (3.65 часа в месяц). Пользователи должны быть предупреждены о запланированных простоях не менее, чем за 8 часов. |
| Обоснование | Время простоя заложено под выпуск релизов, устранение критических багов и непредвиденных ситуаций. |
| Способ тестирования | Тестирование специалистом QA. Как вариант нагрузочное тестирование с помощью JMeter. |
| Приоритет | Высокий. |

### **4.2.12 НФТ12** - **Язык программирования**

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Серверная часть системы должна быть написана с использованием языка C# и платформы .Net. |
| Обоснование | Система будет разрабатываться разработчиками компании заказчика. В компании принято писать бэкенд на языке C#. Поэтому данное приложение также должно быть написано на данном языке программирования. |
| Способ тестирования | Перейти в систему управления репозиториями кода для Git и посмотреть на каком языке ведется проект. |
| Приоритет | Критический. |

### **4.2.13 НФТ13** - **Логирование системы**

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | В системе должно быть предусмотрено наличие логирования и оповещения о сбоях. |
| Обоснование | Для исправления багов необходимо логирование. |
| Способ тестирования | Как минимум разработчики должны писать тесты + тестирование специалистом QA определенных моментов поддерживаемый формат, максимальный размер каждой записи, тестирование сколько событий попало в лог и какие именно это были события и т.д. |
| Приоритет | Критический. |

### 4.2.14 НФТ14 - Темная тема

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Веб-приложение должно поддерживать темную тему. |
| Обоснование | Система разрабатывается для внутреннего использование. Наличие темной темы пожелание разработчиков. |
| Способ тестирования | Ручное тестирование специалистом QA. |
| Приоритет | Низкий. |

### **4.2.15 НФТ15** - **Время отклика загрузки страниц**

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Основные страницы должны загружаться не более 1 секунды с расчетом на 1000 пользователей, работающих в системе одновременно. |
| Обоснование | При более долгой загрузке работать будет не удобно. |
| Способ тестирования | Нагрузочное тестирование специалистом QA, например, с помощью JMeter. |
| Приоритет | Критический. |

### **4.2.16 НФТ16** - **Время отклика загрузки отчетов**

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Генерация отчетов в системе должна происходить максимум за 15 минут. |
| Обоснование | Система должна позволять выгружать различные отчеты. Время ориентировочное, заказчик хочет мгновенно, пока ограничили так. |
| Способ тестирования | Проверка специалистом QA выгрузки отчетов. |
| Приоритет | Высокий. |

### **4.2.17 НФТ17** - **Персональные данные**

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Система должна соответствовать 152-ФЗ «О персональных данных» и требованиям GDPR. |
| Обоснование | В системе планируется сбор персональных данных заказчиков. |
| Способ тестирования | Проверка специалистам ИБ сторонней компании (например, ITGLOBAL.COM). |
| Приоритет | Критический. |

### **4.2.18 НФТ18** - **Кроссбраузерность**

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Веб-приложение должно быть кроссбраузерным. Должны поддерживаться:   1. Google Chrome. 2. Яндекс.Браузер. 3. Opera. 4. Mozilla Firefox. 5. Safari.   Версии: последняя и предыдущая для всех браузеров. |
| Обоснование | Доступ к системе планируется давать заказчикам. Необходимо, чтобы ему была доступна полная функциональность вне зависимости от используемого браузера. |
| Способ тестирования | Ручное тестирование специалистом QA.  Тестирование с использованием сторонних сервисов (CrossBrowserTesting, Browsershots, IE NetRenderer, Spoon Browser Sandbox). |
| Приоритет | Нормальный. |

### 4.2.19 НФТ19 - Поддерживаемые языки

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Система должна поддерживать русский и английский языки. |
| Обоснование | Планируется предоставлять доступ к системе заказчикам. В 2024 году запланирован выход на международный рынок и соответственно на иностранных заказчиков. |
| Способ тестирования | Ручное тестирование специалистом QA.  Проверяем языковые особенности:   1. На всех страницах сайта или экранах приложения используется одинаковая терминология. 2. Отсутствуют грамматические ошибки. 3. Отсутствуют орфографические ошибки. 4. Соблюдены правила пунктуации. 5. Указаны правильные названия городов, мест, должностей и прочее. 6. Проверяем пользовательский интерфейс: 7. Все текстовые надписи на картинках локализованы. 8. Макет языковых версий совпадает с оригиналом. 9. Переносы и разрывы строк на страницах размещены правильно. 10. Диалоги, всплывающие окна и уведомления отображаются корректно. 11. Длина строк не превышает существующие ограничения и текст отображается правильно   Проверяем функциональность:   1. Функционал локализованного приложения или сайта. 2. Гиперссылки 3. Работу вводных функций. 4. Поддержку специальных символов для различных локалей и языков. 5. Поддержку различных разделителей формата.   Можно обратиться к сторонней компании, которая предоставляет спектр услуг по локализационному тестированию и оценке качества. |
| Приоритет | Низкий. |

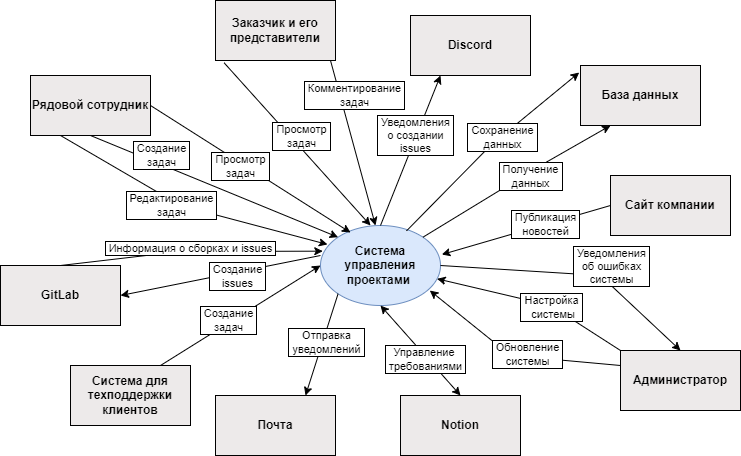
### 4.2.20 НФТ20 - Время и дата

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 14.03.2022 |
| Требование | Время и дата должны соответствовать тайм-зоне пользователя.  Формат дат: дд-мм-гггг.  Формат времени: чч:мм (24-часовой формат). |
| Обоснование | Планируется перевод сотрудников компании на удаленную работу. В связи с этим набор на новые должности будет вестись по различным регионам РФ. Сотрудники из разных регионов должны видеть в системе корректное время(дедлайны, совещания и т.д.). |
| Способ тестирования | Ручное тестирование специалистом QA. |
| Приоритет | Нормальный. |

# 

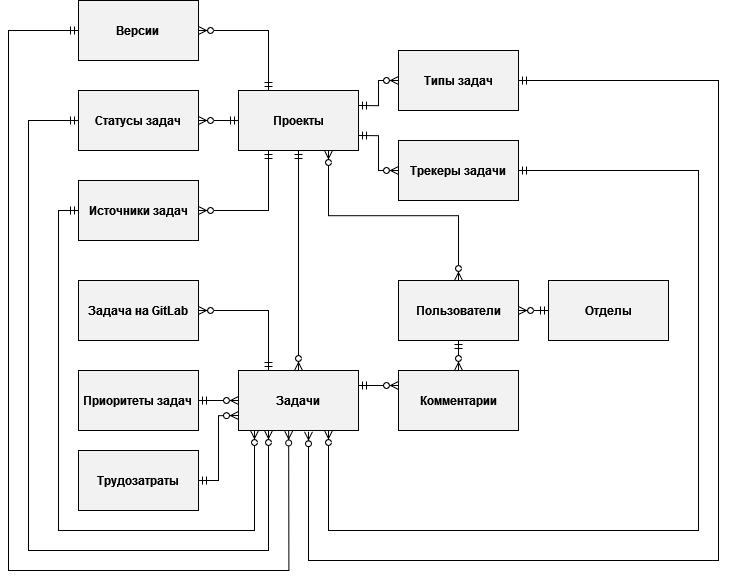
# **5 ДИАГРАММЫ К КУРСОВОМУ ПРОЕКТУ**

## **5.1 Контекстная диаграмма**



## 

## **5.2 Модель сущность-связь**



## 

## **5.3 Описание бизнес-пр**оцессов (**BPMN диаграммы)**

Ссылка на BMNP-диаграммы для создания нового функционала и работы с ошибками: <https://github.com/DashaShabalina/Tinkoff_course/tree/main/BPMN>

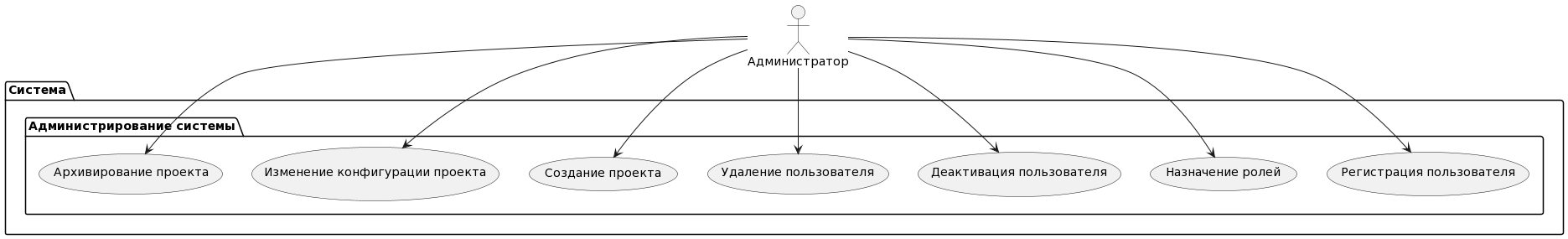
### 5.3.1 Описание процесса создания нового функционала

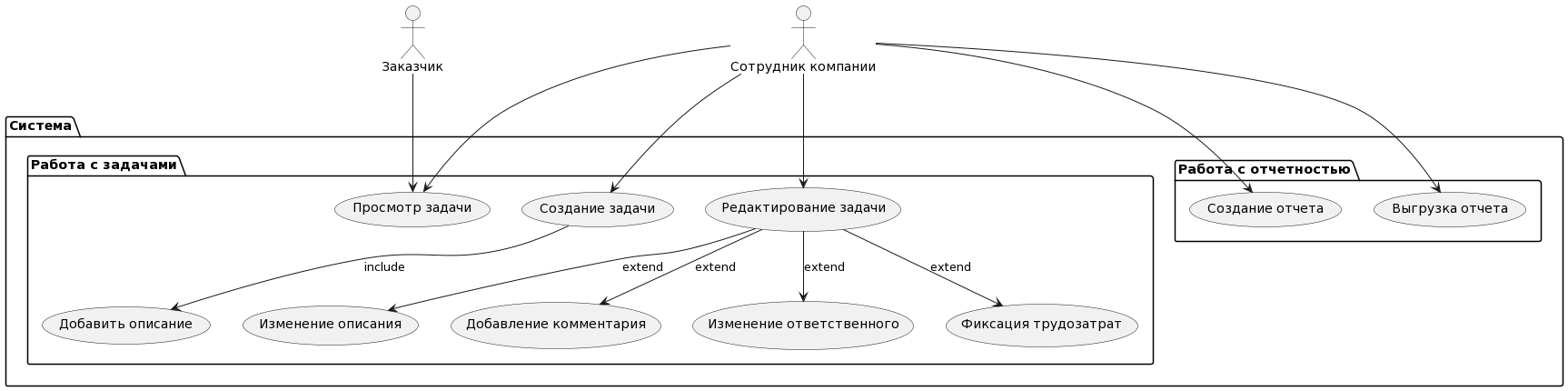
1. Задание поступает от заказчика сервисному инженеру ведущему проект.
2. Сервисный инженер проверяет, что аппаратная реализация данной задачи возможна (есть подходящее оборудование и т.д.).
   1. Если нет аппаратной возможности, то сообщаем заказчику, что задачу невозможно решить в данный момент и требуются дальнейшие обсуждения и поиски подходящего оборудования.
   2. Если да, пункт 3.
3. Сервисный инженер создает в системе задачу, описывает функциональность и назначает ее на менеджера проекта.
4. Менеджер проекта дает первоначальную оценку задаче.
   1. Если выставляют нереалистичные сроки сразу, требуют функциональность, которую сложно встроить в текущую архитектуру и т.д., то задача отправляется на доработку сервисному инженеру.
   2. Если менеджера устраивает первоначальная постановка, желаемые сроки и т.д., он создает подзадачу на аналитика для дальнейшей проработки.
5. Аналитик прорабатывает ТЗ и согласовывает его со всеми заинтересованными лицами.
6. После окончательного согласования менеджер проекта создает задачу на разработку и помещает ее в бэклог продукта.
7. Задача находится в бэклоге до тех пор пока на PBR не будет решено взять данную задачу в спринт.
8. Задача назначается на конкретного разработчика с уже четкими требованиями, дедлайнами и т.д., а также инициатор задачи дополнительно уведомляется об этом.
9. Идет процесс разработки.
10. Тестирование (при необходимости возврат к шагу 9).
11. Написание пользовательской и сервисной документации.
12. Слияние с мастер-веткой.
13. Сбор обратной связи сотрудником техподдержки.

### 5.3.2 Описание процесса работы с ошибками

1. Задача поступает от клиента в техподдержку.
2. Сотрудник фиксирует задачу в системе.
3. Если не хватает данных (Пришла заявка ”все сломалось, ничего не работает”) сотрудник техподдержки запрашивает необходимые дополнительные данные.
4. Сотрудник техподдержки проводит первичный анализ.
   1. Если проблема верхнеуровневая и не требует работы сотрудников других отделов (например, ошибка в неправильных настройках), то уведомляем клиента о причине проблемы и способах решения.
   2. Если проблема глубже, то создает задачу на старшего тестировщика.
5. Старший тестировщик назначает ответственного на задачу.
6. Ответственный за задачу сотрудник пытается воспроизвести описанный баг и анализирует другие источники для решения проблем (смотрит логи, тестирует запросы в postman и т.д.)
   1. Если получилось воспроизвести баг, то тестировщик описывает в кейсе шаги для воспроизведения.
   2. Если найдена причина проблемы, то также отписывается о причине проблемы в кейсе.
7. Если тестировщик может самостоятельно решить проблему, то исправляет ее и уведомляет сотрудника техподдержки.
8. Если тестировщик не может сам решить проблему, то переводит задачу на старшего разработчика.
9. Старший разработчик назначает ответственного на задачу.
10. Идет работа над исправлением бага
    1. Если обнаруживается, что проблема аппаратная(например, по логам видно, что какое-то оборудование не работало), то передает задачу сервисному отделу.
    2. Если проблема программная, то решает ее и передает в тестирование по итогам работы.
11. Тестирование исправления бага(при необходимости возврат в разработку).
12. 12. Сбор обратной связи сотрудником техподдержки.

## **5.4** Сценарии использования(MVP)

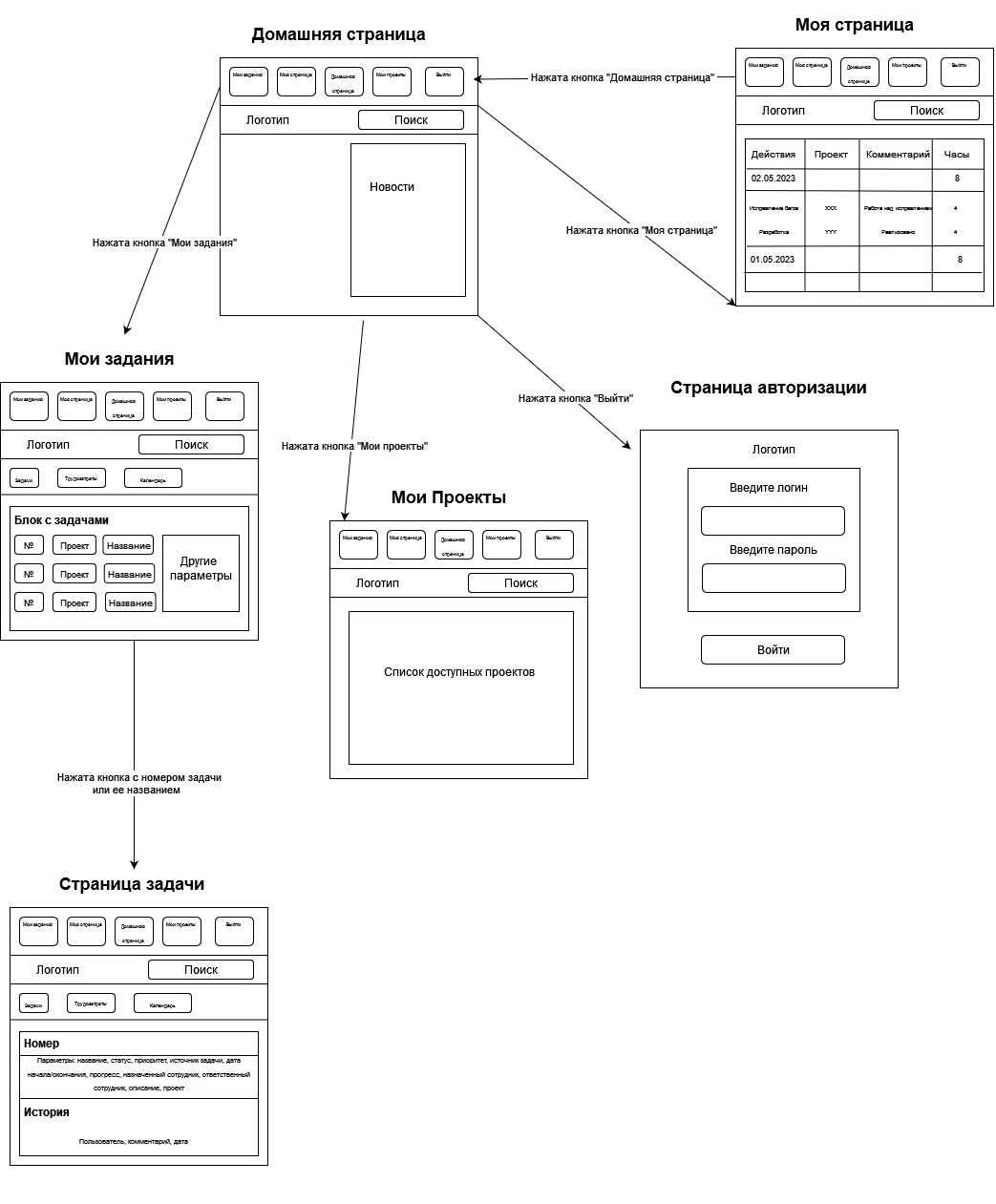




## 

# **6 API**

## **6.1 Экраны**



## **6.1 API**

Ссылка на файл с описанием API:

<https://github.com/DashaShabalina/Tinkoff_course/tree/main/API>

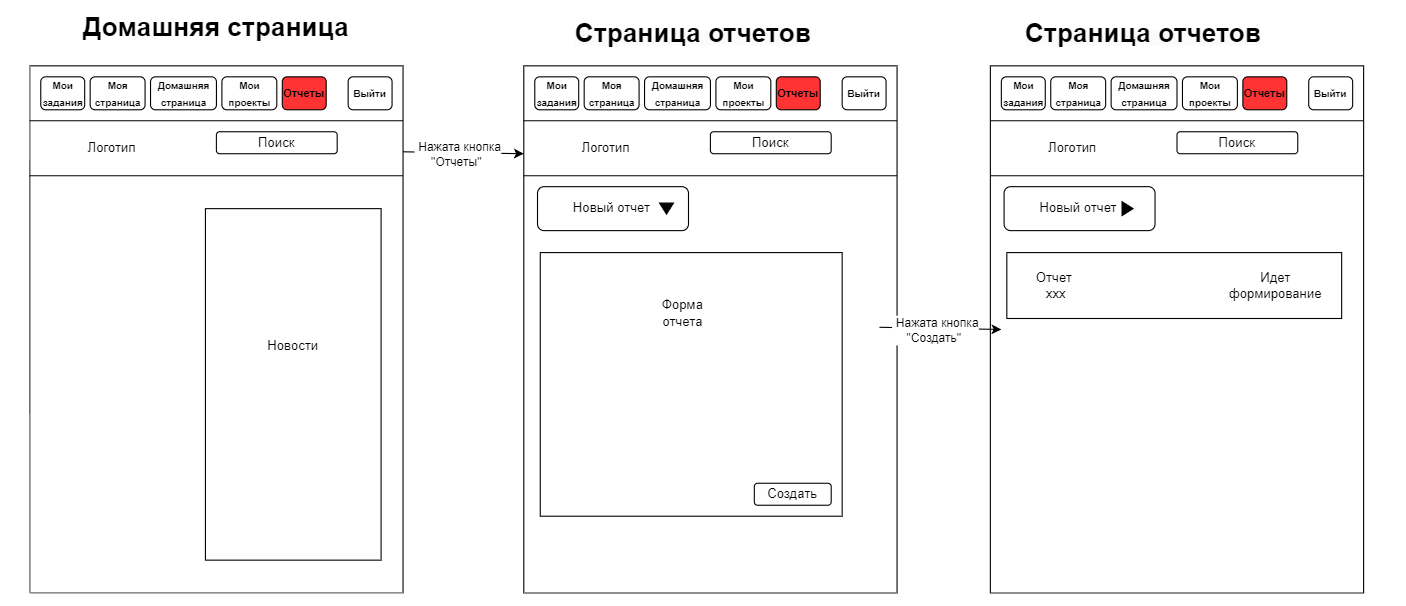
## **6.**2 **Проектирование асинхронного взаимодействия**

### 6.2.1 Описание функциональности

В приложении должна быть возможность сформировать отчет по затратам времени. В отчет попадают только задачи, для которых ведется учет времени. Чтобы сформировать отчет:

1. Клиент нажимает кнопку **Отчет по затратам времени**.
2. Выбирает тип отчета:
   1. **Отчет по исполнителям** — для каждого исполнителя в таблице приводится список задач, над которыми он работал, их тип, очередь, приоритет и данные по затраченному времени.
   2. **Отчет по трудозатратам** — отчет, сгруппированный по задачам. Для каждой задачи в таблице приводится список исполнителей, работавших над задачей, ее тип, приоритет и данные по затраченному времени.
3. Указывается период, за который будет сформирован отчет.
4. Клиент нажимает кнопку **Создать отчет**.
5. На сайте появляется уведомление, информирующее, что идет формирование отчета.
6. Как только от сервера будет получены все данные, клиенту становится доступна кнопка Выгрузить в xlsx.

**Макеты:**

****

**Обоснование решения:**

Предлагается использовать асинхронный REST - Polling и throttling запросов (чтобы не положить сервер).

Исходя из концепции система создается для небольшого количества пользователей и для внутреннего использования. Предполагается, что MVP будет монолитом. Отчеты предположительно будут выгружаться раз в спринт(две недели), поэтому нет смысла держать только для них WS или делать очередь.

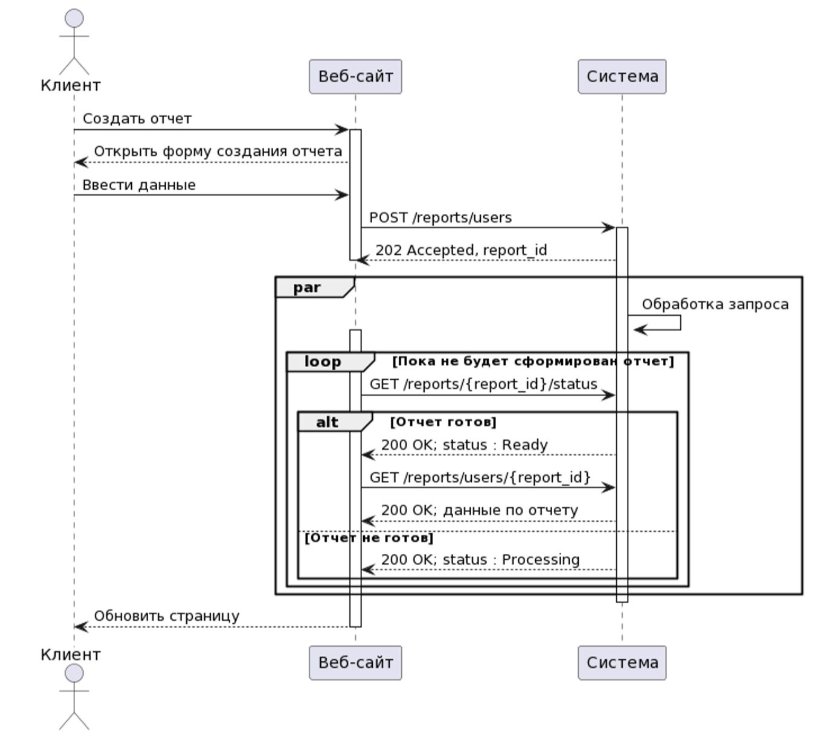
### 6.2.2 ФТ-11 отчет по исполнителям

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 06.05.2023 |
| Участники варианта использования | Авторизованный пользователь системы. |
| Предусловие | Открыта веб-страница «Отчеты». |
| Триггер | Пользователь нажимает кнопку «Новый отчет». |
| Основной сценарий | 1. Пользователь нажимает кнопку «Новый отчет».  2. Пользователь выбирает тип отчета «Отчет по исполнителям».  3. Пользователь выбирает (из доступных ему): 3.1. Отдел.  3.2. Проект.  3.3. Конкретных участников (есть чекбокс выбрать всех). 4. Пользователь нажимает кнопку «Создать». 5. Форма закрывается и начинает формироваться отчет. 6. Как только отчет будет сформирован, пользователю становится доступна кнопка «Выгрузить в xlsx». |
| Альтернативный сценарий | 1. Пользователь заполняет не все обязательные поля.  2. Пользователь нажимает кнопку «Создать».  3. Система выводит сообщение об ошибке. 4. Незаполненные обязательные поля подсвечиваются красным цветом. 5. Работа возобновляется с шага 2 основного сценария. |
| Приоритет | Нормальный. |

**Код:**

| @startuml  actor "Клиент" as emp  participant "Веб-сайт" as web  participant "Система" as system  emp -> web: Создать отчет  activate web  web --> emp: Открыть форму создания отчета  emp -> web: Ввести данные  web -> system: POST /reports/users  activate system  system --> web : 202 Accepted, report\_id  deactivate web  par  system -> system: Обработка запроса  loop Пока не будет сформирован отчет  activate web  web -> system: GET /reports/{report\_id}/status  alt Отчет готов  system --> web : 200 OK; status : Ready  web -> system: GET /reports/users/{report\_id}  system --> web : 200 OK; данные по отчету  else Отчет не готов  system --> web : 200 OK; status : Processing  end  end  end  deactivate system  web --> emp : Обновить страницу  deactivate web  @enduml |
| --- |

**Диаграмма:**



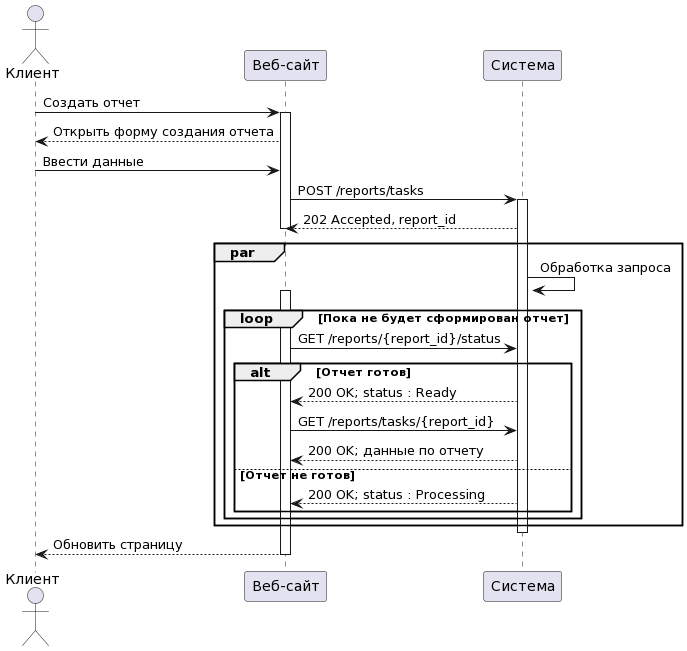
### 6.2.3 ФТ-12 отчет по трудозатратам

| Автор | Шабалина Дарья |
| --- | --- |
| Дата создания | 06.05.2023 |
| Участники варианта использования | Авторизованный пользователь системы. |
| Предусловие | Открыта веб-страница «Отчеты». |
| Триггер | Пользователь нажимает кнопку «Новый отчет». |
| Основной сценарий | 1. Пользователь нажимает кнопку «Новый отчет».  2. Пользователь выбирает тип отчета «Отчет по трудозатратам».  3. Пользователь выбирает (из доступных ему): 3.1. Отдел.  3.2. Проект.  3.3. Конкретные задачи (есть чекбокс выбрать все). 4. Пользователь нажимает кнопку «Создать». 5. Форма закрывается и начинает формироваться отчет. 6. Как только отчет будет сформирован, пользователю становится доступна кнопка «Выгрузить в xlsx». |
| Альтернативный сценарий | 1. Пользователь заполняет не все обязательные поля.  2. Пользователь нажимает кнопку «Создать».  3. Система выводит сообщение об ошибке. 4. Незаполненные обязательные поля подсвечиваются красным цветом. 5. Работа возобновляется с шага 2 основного сценария. |
| Приоритет | Нормальный. |

**Код:**

| @startuml  actor "Клиент" as emp  participant "Веб-сайт" as web  participant "Система" as system  emp -> web: Создать отчет  activate web  web --> emp: Открыть форму создания отчета  emp -> web: Ввести данные  web -> system: POST /reports/tasks  activate system  system --> web : 202 Accepted, report\_id  deactivate web  par  system -> system: Обработка запроса  loop Пока не будет сформирован отчет  activate web  web -> system: GET /reports/{report\_id}/status  alt Отчет готов  system --> web : 200 OK; status : Ready  web -> system: GET /reports/tasks/{report\_id}  system --> web : 200 OK; данные по отчету  else Отчет не готов  system --> web : 200 OK; status : Processing  end  end  end  deactivate system  web --> emp : Обновить страницу  deactivate web  @enduml |
| --- |

**Диаграмма:**



# **7 ПРОЕКТИРОВАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ**

## **7.1 Описание сущностей**

**1. Сущность Проект**

| Атрибут | Описание | Тип данных | Значение по умолчанию | Ограничение | Обязательность | Пример |
| --- | --- | --- | --- | --- | --- | --- |
| Идентификатор | UUID проекта | Текстовый |  |  | Да | 87a3eea4-2858-47fb-9712-cb8a2b564923 |
| Название | Название проекта | Текстовый |  | 50 символов | Да | MARS |
| Описание | Подробное описание проекта | Текстовый |  | 1000 символов | Нет | Проект посвящен созданию автоматизированной системы управления буровзрывными работами |
| Дата начала | Плановая или фактическая дата начала проекта | Дата |  |  | Нет | 2009-06-04 |
| Дата завершения | Плановая или фактическая дата завершения проекта | Дата |  |  | Нет | 2009-06-04 |

**2. Сущность Задача**

| Атрибут | Описание | Тип данных | Значение по умолчанию | Ограничение | Обязательность | Пример |
| --- | --- | --- | --- | --- | --- | --- |
| Идентификатор | UUID задачи | Текстовый |  |  | Да | 87a3eea4-2858-47fb-9712-cb8a2b564923 |
| Номер задачи | Номер задачи в системе | Текстовый |  | 200 символов | Да | 40370 |
| Проект | UUID проекта, к которому относится задача | Текстовый |  |  | Да | 97a3eea4-2858-47fb-9712-cb8a2b564923 |
| Название | Краткое описание задачи | Текстовый |  | 200 символов | Да | Корректировка глубины при начале вдавливания |
| Описание | Подробное описание задачи | Текстовый |  | 1000 символов | Да | Исправить глубину …. |
| Автор задачи | Сотрудник, который создал задачу | Текстовый |  |  | Да | 54a3eea4-2858-47fb-9712-cb8a2b564923 |
| Приоритет | Идентификатор приоритета | Числовой |  |  | Да | 1 |
| Назначенный сотрудник | Сотрудник, на которого в данный момент назначена задача | Текстовый |  |  | Да | 54a3eea4-2858-47fb-9712-cb8a2b564923 |
| Ответственный сотрудник | Сотрудник, отвечающий за выполнение задачи | Текстовый |  |  | Нет | 13a3eea4-2858-47fb-9712-cb8a2b564923 |
| Дата начала | Сотрудник, отвечающий за выполнение задачи | Дата |  |  | Да | 2011-04-15 00:03:20 |
| Дата завершения | Плановая дата завершения задачи | Дата |  |  | Нет | 2011-04-15 00:03:20 |
| Оценка временных затрат | Оценочное время на реализацию задачи в часах | Числовой |  |  | Нет | 16 |
| Готовность | Готовность решения в процентах | Числовой | 0 |  | Да | 10 |
| Трекер | UUID трекера |  |  |  | Да | 17a3eea4-2858-47fb-9712-cb8a2b564923 |
| Источник задачи | UUID источника задачи | Текстовый |  |  | Да | 14a3eea4-2858-47fb-9712-cb8a2b564923 |
| Тип задачи | UUID типа задачи | Текстовый |  |  | Да | 16a3eea4-2858-47fb-9712-cb8a2b564923 |
| Статус задачи | UUID статуса задачи | Текстовый |  |  | Да | 15a3eea4-2858-47fb-9712-cb8a2b564923 |
| Версия | UUID версии | Текстовый |  |  | Да | 17a3eea4-2858-47fb-9712-cb8a2b564923 |
| Затраченные часы | Время, которое сотрудник потратил на выполнение задачи | Числовой |  |  | Нет | 10 |

### 

**3. Сущность Комментарий**

| Атрибут | Описание | Тип данных | Значение по умолчанию | Ограничение | Обязательность | Пример |
| --- | --- | --- | --- | --- | --- | --- |
| Идентификатор | UUID комментария | Текстовый |  |  | Да | 87a3eea4-2858-47fb-9712-cb8a2b564923 |
| Задача | UUID задачи, к которой написан комментарий | Текстовый |  |  | Да | 77a3eea4-2858-47fb-9712-cb8a2b564923 |
| Пользователь | UUID пользователя, оставившего комментарий | Текстовый |  |  | Да | 67a3eea4-2858-47fb-9712-cb8a2b564923 |
| Текст комментария | Текст комментария | Текстовый |  | 500 символов | Да | Прошу проверить кейс повторно |
| Дата | Дата создания комментария | Дата |  |  | Да | 2011-04-15 00:03:20 |
| Дата обновления | Дата обновления комментария | Дата |  |  | Нет | 2011-04-16 00:03:20 |

### 

**4. Сущность Приоритет задачи**

| Атрибут | Описание | Тип данных | Значение по умолчанию | Ограничение | Обязательность | Пример |
| --- | --- | --- | --- | --- | --- | --- |
| Идентификатор | ID приоритета | Числовой |  |  | Да | 1 |
| Приоритет | Уровень важности задачи | Текстовый |  | 50 символов | Да | Срочный |

### 

**5. Сущность Задача на GitLab**

| Атрибут | Описание | Тип данных | Значение по умолчанию | Ограничение | Обязательность | Пример |
| --- | --- | --- | --- | --- | --- | --- |
| Идентификатор | UUID задачи | Текстовый |  |  | Да | 67a3eea4-2858-47fb-9712-cb8a2b564923 |
| Идентификатор задачи | Ссылка на задачу в системе | Текстовый |  |  | Да | 77a3eea4-2858-47fb-9712-cb8a2b564923 |
| URL | Ссылка на задачу в GitLab | Текстовый |  |  | Да | http://172.29.10.19/xxx/yyy/-/issues/4254 |
| Статус | Состояние Issues на GitLab (Открыта, Закрыта) | Логический |  |  | Да | true |

### 

**6. Сущность Источник задачи**

| Атрибут | Описание | Тип данных | Значение по умолчанию | Ограничение | Обязательность | Пример |
| --- | --- | --- | --- | --- | --- | --- |
| Идентификатор | UUID источника | Текстовый |  |  | Да | 47a3eea4-2858-47fb-9712-cb8a2b564923 |
| Источник | Постановщик задачи | Текстовый |  | 50 символов | Да | Тестировщик |
| Проект | UUID проекта | Текстовый |  |  | Да | 67a3eea4-2858-47fb-9712-cb8a2b564923 |

**7. Сущность Тип задачи**

| Атрибут | Описание | Тип данных | Значение по умолчанию | Ограничение | Обязательность | Пример |
| --- | --- | --- | --- | --- | --- | --- |
| Идентификатор | UUID  типа задачи | Текстовый |  |  | Да | 81a3eea4-2858-47fb-9712-cb8a2b564923 |
| Тип | Тип задачи | Текстовый |  | 50 символов | Да | Bug |
| Проект | UUID  проекта | Текстовый |  |  | Да | 67a3eea4-2858-47fb-9712-cb8a2b564923 |

### 

**8. Сущность Статус задачи**

| Атрибут | Описание | Тип данных | Значение по умолчанию | Ограничение | Обязательность | Пример |
| --- | --- | --- | --- | --- | --- | --- |
| Идентификатор | UUID статуса | Текстовый |  |  | Да | 14a3eea4-2858-47fb-9712-cb8a2b564923 |
| Статус | Название статуса | Текстовый |  | 50 символов | Да | Новая |
| Проект | UUID проекта | Текстовый |  |  | Да | 67a3eea4-2858-47fb-9712-cb8a2b564923 |

### 

**9. Сущность Версия**

| Атрибут | Описание | Тип данных | Значение по умолчанию | Ограничение | Обязательность | Пример |
| --- | --- | --- | --- | --- | --- | --- |
| Идентификатор | UUID версии (спринта) | Текстовый |  |  | Да | 14a3eea4-2858-47fb-9712-cb8a2b564923 |
| Название | Название | Текстовый |  | 50 символов | Да | 2.20.0 спринт 2 |
| Проект | UUID проекта | Текстовый |  |  | Да | 67a3eea4-2858-47fb-9712-cb8a2b564923 |
| Дата начала | Дата начала спринта | Дата |  |  | Нет | 2011-04-15 |
| Дата завершения | Дата окончания спринта | Дата |  |  | Нет | 2011-04-15 |

### 

**10. Сущность Трекер задачи**

| Атрибут | Описание | Тип данных | Значение по умолчанию | Ограничение | Обязательность | Пример |
| --- | --- | --- | --- | --- | --- | --- |
| Идентификатор | UUID трекера | Текстовый |  |  | Да | 67a3eea4-2858-47fb-9712-cb8a2b564923 |
| Трекер | Название трекера | Текстовый |  | 50 символов | Да | Тестирование |
| Проект | UUID проекта | Текстовый |  |  | Да | 67a3eea4-2858-47fb-9712-cb8a2b564923 |

### 

**11. Сущность Отдел**

| Атрибут | Описание | Тип данных | Значение по умолчанию | Ограничение | Обязательность | Пример |
| --- | --- | --- | --- | --- | --- | --- |
| Идентификатор | UUID отдела | Текстовый |  |  | Да | 14a3eea4-2858-47fb-9712-cb8a2b564923 |
| Название | Название отдела | Текстовый |  | 50 символов | Да | Разработки ПО |
| Проект | Краткое описание функций отдела | Текстовый |  | 1000 символов | Нет | Программисты |

### 

**12. Сущность Пользователь**

| Атрибут | Описание | Тип данных | Значение по умолчанию | Ограничение | Обязательность | Пример |
| --- | --- | --- | --- | --- | --- | --- |
| Идентификатор | UUID пользователя | Текстовый |  |  | Да | 14a3eea4-2858-47fb-9712-cb8a2b564923 |
| Имя | Имя пользователя | Текстовый |  | 50 символов | Да | Дарья |
| Фамилия | Фамилия пользователя | Текстовый |  | 50 символов | Да | Шабалина |
| Активность | Показатель работает ли пользователь в компании или уволен | Логический |  |  | Да | true |
| Дата регистрации | Дата регистрации пользователя в системе | Дата |  |  | Нет | 2011-04-15 00:03:20 |
| Дата последнего подключения | Дата последнего подключения пользователя к системе | Дата |  |  | Нет | 2011-04-16 00:03:20 |
| Отдел | UUID отдела, в котором работает пользователь | Текстовый |  |  | Да | 24a3eea4-2858-47fb-9712-cb8a2b564923 |

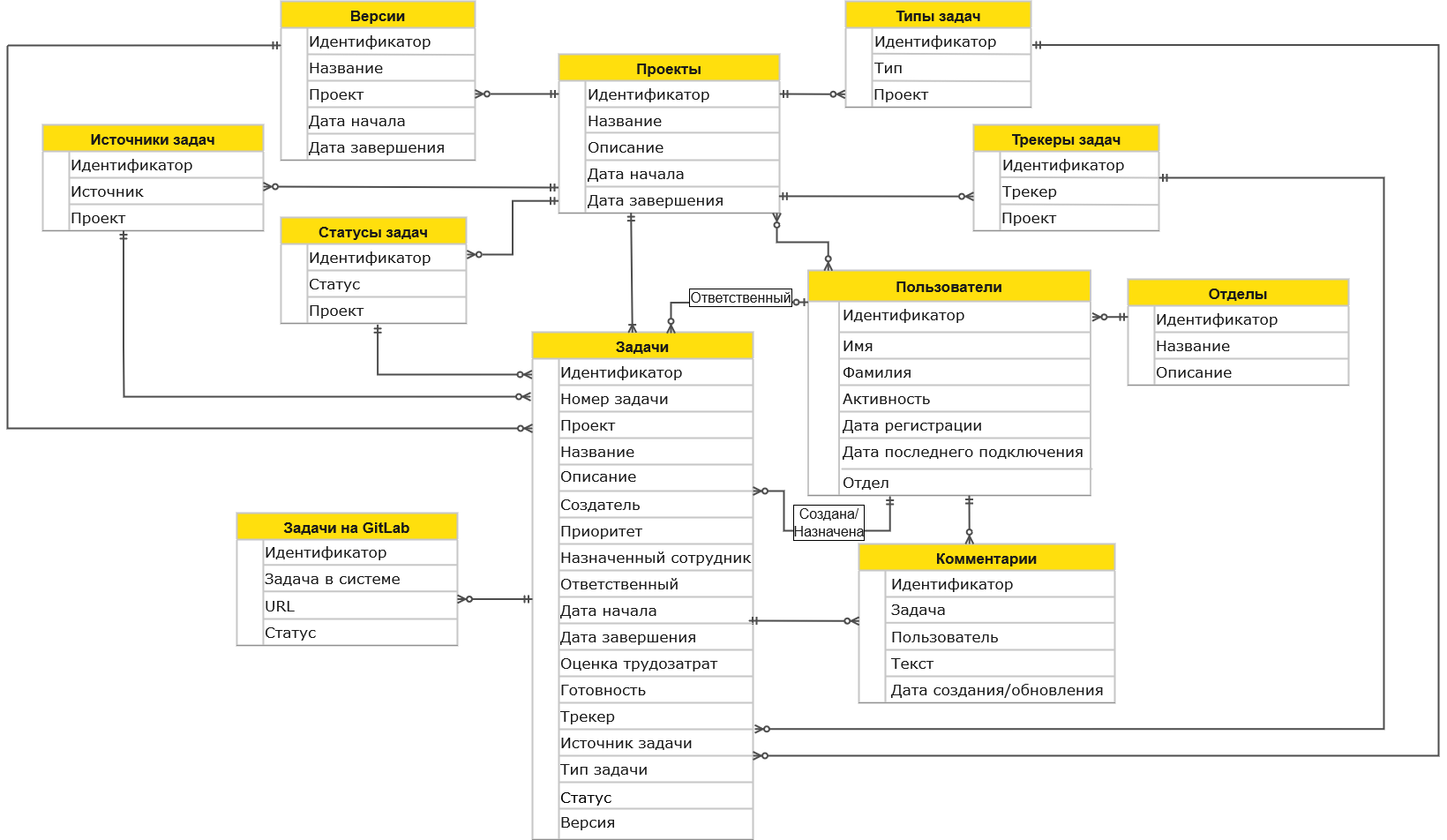
### 

**13. Сущность Трудозатраты**

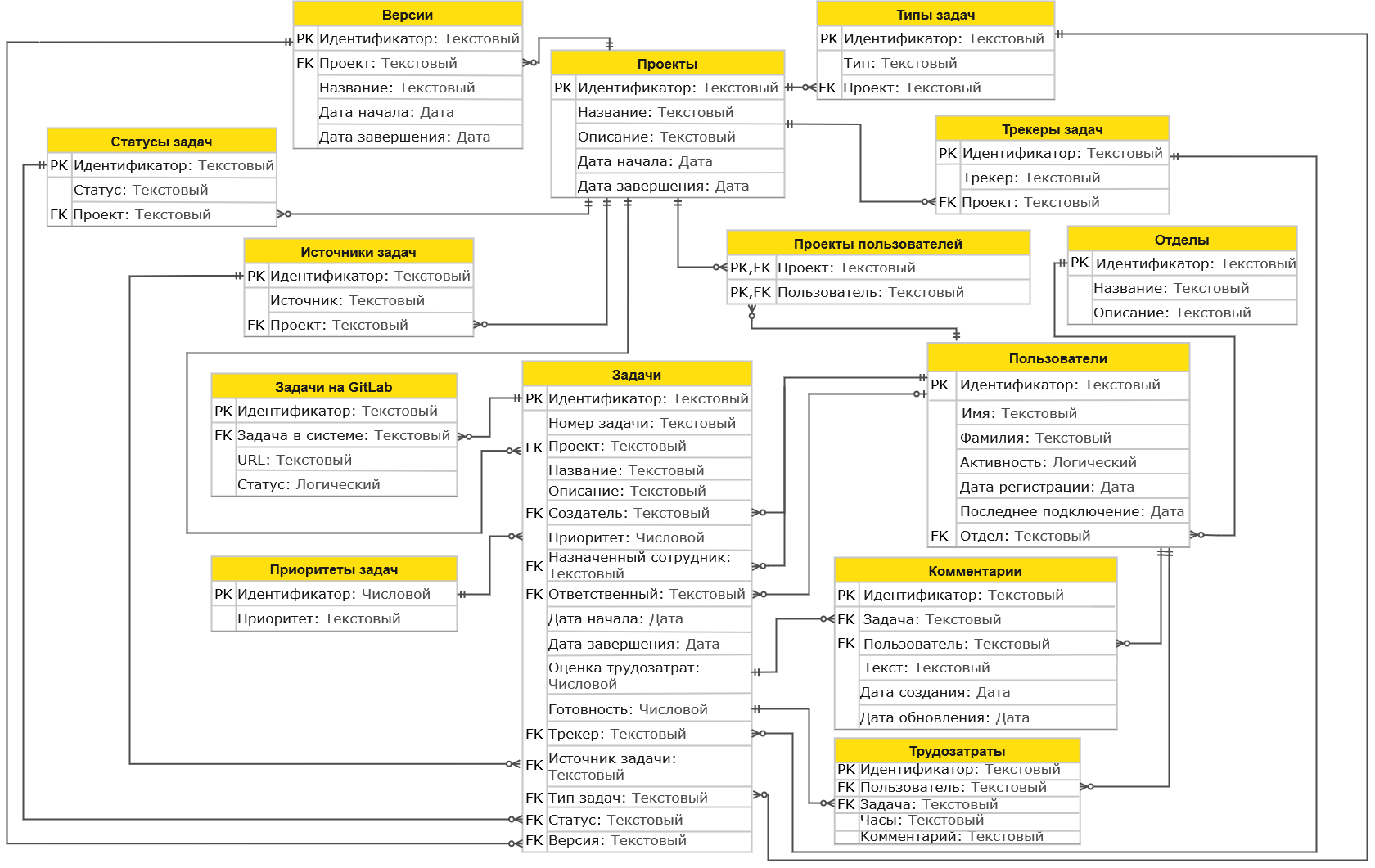
| Атрибут | Описание | Тип данных | Значение по умолчанию | Ограничение | Обязательность | Пример |
| --- | --- | --- | --- | --- | --- | --- |
| Идентификатор | UUID трудозатраты | Текстовый |  |  | Да | 24a3eea4-2858-47fb-9712-cb8a2b564923 |
| Пользователь | UUID пользователя | Текстовый |  |  | Да | 25a3eea4-2858-47fb-9712-cb8a2b564923 |
| Задача | UUID задачи | Текстовый |  |  | Да | 26a3eea4-2858-47fb-9712-cb8a2b564923 |
| Часы | Часы, потраченные на определенное действие | Числовой |  |  | Да | 16.5 |
| Комментарий | Комментарий к трудозатрате | Текстовый |  | 200 символов | Да | Исправлено часть замечаний |

### 

## 7.2 Концептуальная модель

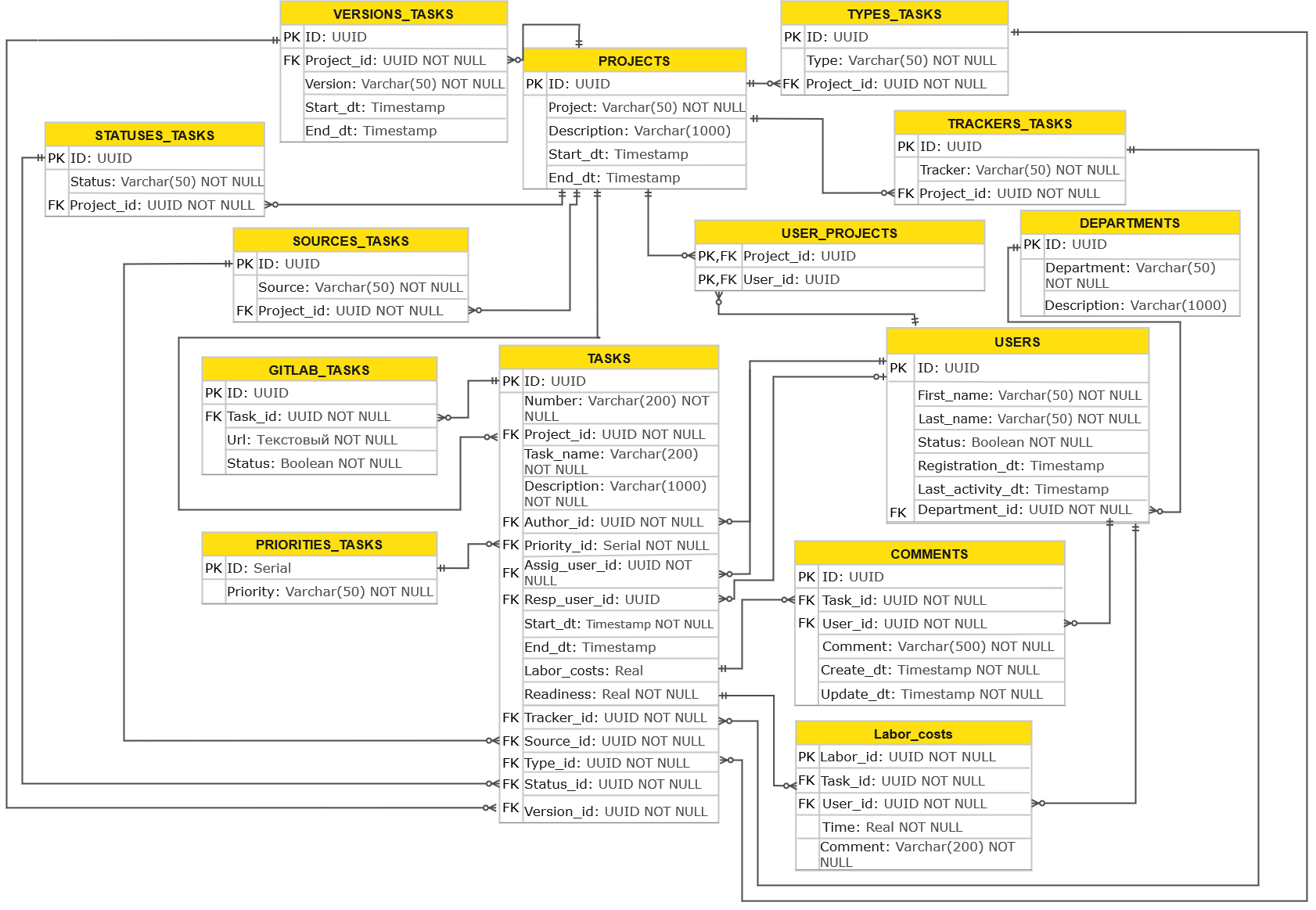


## 7.3 Логическая модель



## 7.4 Физическое представление модели

База данных - PostgreSQL v14.

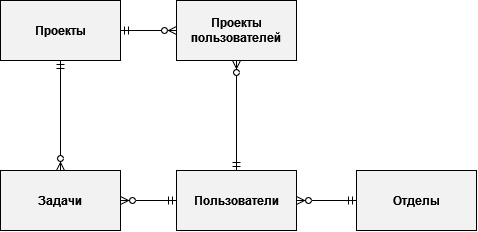


# **8 Практика SQL**

## **8.1 Инструменты**

Используемая база данных - PostgreSQL v14. Проверка работоспособности выполнялась с помощью DB Fiddle.

Для выполнения работы и наполнения БД экспериментальными данными использовалась упрощенная схема базы данных:



Ссылка на файлы со скриптами, описанными в разделах 8.2-8.5: <https://github.com/DashaShabalina/Tinkoff_course/tree/main/DB%20scripts>

## **8.2 Скрипт на создание таблиц**

| --Таблица проектов  -- ID - UUID проекта  -- Name - название  -- Description - описание  -- Start\_date - дата начала  -- End\_date - дата завершения  CREATE TABLE PROJECTS (  ID varchar(200) NOT NULL,  Name varchar(50) NOT NULL UNIQUE,  Description varchar(1000),  Start\_date TIMESTAMP,  End\_date TIMESTAMP,  CONSTRAINT PROJECTS\_pk PRIMARY KEY (ID)  );  --Таблица отделов  -- ID - UUID отдела  -- Name - название  -- Description - описание  CREATE TABLE DEPARTMENTS (  ID varchar(200) NOT NULL,  Name varchar(50) NOT NULL,  Description varchar(1000),  CONSTRAINT DEPARTMENTS\_pk PRIMARY KEY (ID)  );  --Таблица пользователей  -- ID - UUID пользователя  -- First\_name - имя  -- Last\_name - фамилия  -- Reg\_date - дата регистрации в системе  -- Last\_connection\_date - дата последнего подключения в систему  -- Department\_id - UUID отдела, в котором работает пользователь  -- Active - маркер работает ли пользователь в компании(true) или уволен(false)  CREATE TABLE USERS (  ID varchar(200) NOT NULL,  First\_name varchar(50) NOT NULL,  Last\_name varchar(50) NOT NULL,  Reg\_date TIMESTAMP,  Last\_connection\_date TIMESTAMP NOT NULL,  Department\_id varchar(255) NOT NULL,  Active boolean NOT NULL,  CONSTRAINT USERS\_pk PRIMARY KEY (ID)  );  --Таблица для взаимосвязи пользователей с проектами  -- User\_id - UUID пользователя  -- Project\_id - UUID проекта  CREATE TABLE USER\_PROJECTS (  User\_id varchar(200) NOT NULL,  Project\_id varchar(200) NOT NULL,  CONSTRAINT USER\_PROJECTS\_pk PRIMARY KEY (User\_id,Project\_id)  );  --Таблица задач  -- ID - UUID задачи  -- Project\_id - UUID проекта, к которому относится задача  -- Number - номер задачи  -- Name - название  -- Description - описание  -- Priority - приоритет (срочный, высокий, низкий и т.д.)  -- Desig\_emp - UUID пользователя, на которого назначена задача  -- Resp\_emp - UUID пользователя ответственного за задачу  -- Start\_date - дата начала  -- End\_date - дата завершения  -- Labor\_costs - оценка трудозатрат  -- Readness - готовность задачи в процентах  -- Source - источник задачи (тестировщик, заказчик, разработчик и т.д.)  -- Type - тип задачи (баг, фича и т.д.)  CREATE TABLE TASKS (  ID varchar(200) NOT NULL,  Project\_id varchar(200) NOT NULL,  Number varchar(200) NOT NULL UNIQUE,  Name varchar(200) NOT NULL,  Description varchar(1000) NOT NULL,  Priority varchar(50) NOT NULL,  Desig\_emp varchar(200) NOT NULL,  Resp\_emp varchar(200),  Start\_date TIMESTAMP NOT NULL,  End\_date TIMESTAMP,  Labor\_costs real,  Readness real,  Source varchar(50) NOT NULL,  Type varchar(50) NOT NULL,  CONSTRAINT TASKS\_pk PRIMARY KEY (ID)  );  ALTER TABLE USERS ADD CONSTRAINT USERS\_fk0 FOREIGN KEY (Department\_id) REFERENCES DEPARTMENTS(ID);  ALTER TABLE USER\_PROJECTS ADD CONSTRAINT USER\_PROJECTS\_fk0 FOREIGN KEY (User\_id) REFERENCES USERS(ID);  ALTER TABLE USER\_PROJECTS ADD CONSTRAINT USER\_PROJECTS\_fk1 FOREIGN KEY (Project\_id) REFERENCES PROJECTS(ID);  ALTER TABLE TASKS ADD CONSTRAINT TASKS\_fk0 FOREIGN KEY (Project\_id) REFERENCES PROJECTS(ID);  ALTER TABLE TASKS ADD CONSTRAINT TASKS\_fk1 FOREIGN KEY (Desig\_emp) REFERENCES USERS(ID);  ALTER TABLE TASKS ADD CONSTRAINT TASKS\_fk2 FOREIGN KEY (Resp\_emp) REFERENCES USERS(ID); |
| --- |

## **8.3 Скрипт на** наполнение бд **данны**ми

| --заполнение таблицы проектов  do  '  begin  FOR i in 1..30 LOOP  INSERT INTO PROJECTS(ID, Name) VALUES (CAST(i AS varchar), CONCAT(''project'',i));  END LOOP;  end;  ';  --заполнение таблицы отделов  do  '  begin  FOR i in 1..10 LOOP  INSERT INTO DEPARTMENTS(ID, Name) VALUES (CAST(i AS varchar), CONCAT(''department'',i));  END LOOP;  end;  ';  --заполнение таблицы пользователей  CREATE TEMP TABLE Girls (  id serial PRIMARY KEY,  name VARCHAR(40)  );  CREATE TEMP TABLE Men (  id serial PRIMARY KEY,  name VARCHAR(40)  );  CREATE TEMP TABLE Surnames\_Girls (  id serial PRIMARY KEY,  surname VARCHAR(40)  );  CREATE TEMP TABLE Surnames\_Men (  id serial PRIMARY KEY,  surname VARCHAR(40)  );  INSERT INTO Girls(name) VALUES  ('Дарья'), ('Марья'), ('Александра'), ('Валентина'),('Ксения'),  ('София'),('Валерия'),('Виктория'),('Галина'),('Татьяна');  INSERT INTO Men(name) VALUES  ('Юрий'),('Иван'),('Олег'),('Тимофей'),('Михаил'),  ('Руслан'),('Роман'),('Александр'),('Лев'),('Евгений');  INSERT INTO Surnames\_Girls(surname) VALUES  ('Бабурина'),('Секрет'),('Петрова'),('Иванова'),('Тимошко');  INSERT INTO Surnames\_Men(surname) VALUES  ('Комлев'),('Семенов'),('Иванов'),('Петров'),('Данилкин');  do  '  declare  begin  INSERT INTO USERS(ID, First\_name, Last\_name,Last\_connection\_date, Department\_id, Active)  SELECT ROW\_NUMBER() over() as number, name, surname, now(), department, true FROM  (  SELECT Girls.name as name, surname, DEPARTMENTS.ID as department  FROM Girls CROSS JOIN Surnames\_Girls CROSS JOIN DEPARTMENTS  UNION ALL  SELECT Men.name as name, surname, DEPARTMENTS.ID as department  FROM Men CROSS JOIN Surnames\_Men CROSS JOIN DEPARTMENTS  )t  ORDER BY RANDOM();  end;  ';  DROP TABLE Girls;  DROP TABLE Men;  DROP TABLE Surnames\_Girls;  DROP TABLE Surnames\_Men;  --Заполнение таблицы для взаимосвязи пользователей с проектами  INSERT INTO USER\_PROJECTS(User\_id, Project\_id)  Select USERS.ID, PROJECTS.ID FROM USERS CROSS JOIN PROJECTS  ORDER BY RANDOM()  LIMIT 50;  --Заполнение таблицы задач  CREATE TEMP TABLE Priorities (  id serial PRIMARY KEY,  priority VARCHAR(40)  );  INSERT INTO Priorities(priority) VALUES  ('Низкий'), ('Нормальный'), ('Высокий'), ('Срочный'),('Немедленный');  CREATE TEMP TABLE Sources (  id serial PRIMARY KEY,  source VARCHAR(40)  );  INSERT INTO Sources(source) VALUES  ('Заказчик'), ('Тестировщик'), ('Разработчик'), ('Сервисный отдел');  CREATE TEMP TABLE Types (  id serial PRIMARY KEY,  type VARCHAR(40)  );  INSERT INTO Types(type) VALUES  ('Bug'), ('Feature'), ('Документация'), ('Общие вопросы');  INSERT INTO TASKS(ID, Project\_id, Number, Name, Description, Priority, Desig\_emp, Resp\_emp, Start\_date, Source, Type)  Select ROW\_NUMBER() over() as ID,  PROJECTS.ID,  ROW\_NUMBER() over() as Number,  CONCAT('task', ROW\_NUMBER() over()) as Name,  '...' as Description,  priority,  USERS.ID,  USERS.ID,  now(),  source,  type FROM USERS CROSS JOIN Priorities CROSS JOIN Sources CROSS JOIN Types CROSS JOIN PROJECTS  ORDER BY RANDOM()  LIMIT 300;  DROP TABLE Priorities;  DROP TABLE Sources;  DROP TABLE Types; |
| --- |

## **8.4 Скрипт на получение данных**

| --Запросы для проверки заполнения таблиц  --SELECT \* FROM PROJECTS;  --SELECT \* FROM DEPARTMENTS;  --SELECT \* FROM USERS;  --SELECT \* FROM USER\_PROJECTS;  --SELECT \* FROM TASKS;  --Получение всех задач назначенных на сотрудника  --Из-за рандомного заполнения тяжело найти сотрудника наугад, поэтому для проверки взяла сотрудника, у которого есть минимум одна задача  SELECT Number, Name FROM USERS JOIN TASKS ON USERS.ID = TASKS.Desig\_emp  WHERE USERS.ID = (SELECT Desig\_emp FROM TASKS ORDER BY RANDOM() LIMIT 1);  --Задачи, которые пришли от заказчика  SELECT \* FROM TASKS  WHERE Source = 'Заказчик';  --Количество задач каждого сотрудника  SELECT USERS.ID, First\_name, Last\_name, COUNT(TASKS.ID) as count FROM USERS LEFT JOIN TASKS ON USERS.ID = TASKS.Desig\_emp  GROUP BY USERS.ID, First\_name, Last\_name  ORDER BY count DESC  --Сотрудники, у которых нет задач  SELECT USERS.ID, First\_name, Last\_name FROM USERS LEFT JOIN TASKS ON USERS.ID = TASKS.Desig\_emp  WHERE TASKS.ID is NULL; |
| --- |

## **8.5 Скрипт на изменение данных**

| --Назначить ответственного сотрудника за задачу  UPDATE TASKS SET Resp\_emp = (SELECT ID FROM USERS ORDER BY RANDOM() LIMIT 1)  WHERE ID = (SELECT ID FROM TASKS ORDER BY ID DESC LIMIT 1);  --Проверка  --SELECT \* FROM TASKS WHERE ID = (SELECT ID FROM TASKS ORDER BY ID DESC LIMIT 1);  --Всем задачам от заказчика поставить приоритет 'Срочный'  UPDATE TASKS SET Priority = 'Срочный'  WHERE Source = 'Заказчик';  --Проверка  --SELECT \* FROM TASKS WHERE Source = 'Заказчик';  --Удалить задачу из системы  DELETE FROM TASKS  WHERE ID = (SELECT ID FROM TASKS ORDER BY ID ASC LIMIT 1); |
| --- |