

# Спецификация требований к Telegram-боту для взаимодействия с ЛК ФГУП «РЧЦ ЦФО»

## Оглавление

Оглавление.....	1
Общее описание проекта.....	3
Цель проекта.....	3
Основные этапы проекта.....	4
Требования.....	5
User Stories.....	5
Функциональные требования.....	6
1. Управление данными учетной записи пользователя.....	6
2. Управление запросами.....	6
3. Просмотр информации (VIEW).....	7
4. Доступ к файлам.....	7
5. Поддержка.....	7
Модель AS IS.....	8
Модель TO BE.....	8
Варианты использования.....	11
UC:Подать запрос.....	11
UC:Получить отчет.....	12
UC:Посмотреть информацию о запросах.....	14
UC:Скачать отчет.....	15
UC:Добавить учетную запись.....	15
UC:Изменить учетную запись.....	17
UC:Удалить учетную запись.....	18
UC:Получить информацию о боте.....	19
UC:Написать отзыв разработчику.....	19
Нефункциональные требования.....	20
Права и роли.....	23
Метрики.....	23
Получение доступа к сайту.....	24
Основные кнопки действий, доступные пользователю.....	25

Кнопка "/start" -> "Создать заявку" .....	25
Кнопка "/start" -> "Получить заявку" .....	25
Кнопка "/settings" .....	25
Кнопка "/about" .....	26
Кнопка "/feedback" .....	26
Запросы, отправляемые на сервер .....	26
Приложения .....	27
Приложение 1. Алгоритм оформления комментария к полученной заявке .....	27

## Общее описание проекта

Проект “Telegram-бот для ЛК ФГУП «РЧЦ ЦФО»” направлен на разработку и внедрение Telegram-бота, который обеспечит пользователям удобный, быстрый и безопасный доступ к функционалу личного кабинета ФГУП «РЧЦ ЦФО» через мессенджер Telegram. Целью проекта является оптимизация рабочих процессов, повышение мобильности и эффективности взаимодействия пользователей с системой за счет предоставления интуитивно понятного и автоматизированного интерфейса.

В рамках данного проекта будет разработан Telegram-бот, который позволит пользователям личного кабинета ФГУП «РЧЦ ЦФО» выполнять следующие действия непосредственно из мессенджера:

- Просмотр информации: получение оперативной информации о статусе заявок, истории созданных заявок, хранящихся в личном кабинете.
- Управление заявками: создание новых заявок, отслеживание их статуса.
- Доступ к документам: получение необходимых документов, связанных с заявками.
- Автоматизация: автоматизация рутинных операций, таких как отправка запросов и получение отчетов.

## Цель проекта

### Основная цель проекта

Обеспечить удобный, быстрый и мобильный доступ к ключевым функциям личного кабинета ФГУП «РЧЦ ЦФО» посредством Telegram-бота, с целью оптимизации рабочих процессов и повышения эффективности взаимодействия пользователей с системой.

### Развернутое описание цели

**Мобильный доступ:** предоставить возможность доступа к информации и функциям личного кабинета с мобильных устройств через удобный интерфейс Telegram-бота, что особенно важно для сотрудников, работающих вне офиса.

**Упрощение взаимодействия:** сделать процесс взаимодействия с личным кабинетом более простым и интуитивно понятным, не требующим от пользователей постоянного обращения к веб-интерфейсу.

**Повышение эффективности:** увеличить общую эффективность взаимодействия пользователей с личным кабинетом за счет ускорения процессов, снижения вероятности ошибок и экономии времени.

**Техническая интеграция:** обеспечить надежную и бесперебойную интеграцию Telegram-бота с личным кабинетом ФГУП «РЧЦ ЦФО» посредством API или других подходящих методов.

Поставленная цель направлена на:

1. Повышение эффективности работы сотрудников:

- Сократить время, затрачиваемое сотрудниками на выполнение рутинных операций в ЛК ФГУП «РЧЦ ЦФО»  
<https://portal.rfc-revizor.ru/>
- Оптимизировать процессы взаимодействия с личным кабинетом.
- Увеличить производительность труда сотрудников.

2. Обеспечение мобильности доступа для сотрудников:

- Предоставить сотрудникам возможность доступа к ключевым функциям личного кабинета с мобильных устройств в любое время и в любом месте.
- Улучшить взаимодействие с системой для сотрудников, работающих вне офиса.

## Основные этапы проекта

1. Анализ и проектирование:

- Сбор и анализ требований к функционалу бота.
- Проектирование архитектуры бота и интерфейса пользователя.
- Выбор необходимых технологий и инструментов разработки.

2. Разработка:

- Написание кода бота и его интеграция с личным кабинетом.
- Реализация функций просмотра информации, управления заявками, доступа к документам и автоматизации процессов.
- Разработка системы аутентификации пользователей.

3. Тестирование:
  - Проведение тестирования бота на всех этапах разработки.
  - Исправление ошибок и внесение необходимых изменений.
4. Внедрение и развертывание:
  - Развертывание бота на сервере.
  - Обеспечение доступа к боту для пользователей.
5. Поддержка и сопровождение:
  - Предоставление технической поддержки пользователям.
  - Регулярное обновление функционала бота на основе обратной связи от пользователей.

## Требования

### User Stories

1. “Как пользователь, я хочу быстро авторизоваться без ввода пары логин/пароль при каждом входе в Telegram-бот, чтобы сократить время на создание отчета”.
2. “Как пользователь, я хочу иметь возможность создавать запросы, выбирая несколько дат одновременно, чтобы не нажимать снова “создать запрос” и выбирать дату, когда необходимо создать запросы на несколько дат”.
3. “Как пользователь, я хочу иметь возможность просмотреть список созданных запросов и их статус, чтобы узнать, когда отчет перешел в статус “готов” и его можно скачать”
4. “Как пользователь, я хочу иметь возможность скачать готовый отчет только в формате .pdf, чтобы не тратить время на извлечение данных из файла формата .zip”
5. “Как пользователь, я хочу видеть готовый комментарий по отчету, который я пишу, анализируя отчет по специальному алгоритму, чтобы не тратить время на постоянное описание отчетов по выработанному алгоритму”

# Функциональные требования

## 1. Управление данными учетной записи пользователя

- **Локальное хранение данных аутентификации:**
  - Бот должен сохранять данные учетной записи пользователя от ЛК ФГУП «РЧЦ ЦФО».
  - Данные должны храниться в безопасном виде.
- **Управление профилем пользователя:**
  - Бот должен предоставлять возможность добавлять учетную запись.
  - Бот должен предоставлять возможность изменять данные учетной записи.
  - Бот должен предоставлять возможность удалять данные учетной записи.
- **Авторизация:**
  - Бот должен автоматически отправлять данные учетной записи пользователя для авторизации в ЛК ФГУП «РЧЦ ЦФО» (логин и пароль).
  - Для авторизации бот должен пересылать капчу, сгенерированную на ФГУП «РЧЦ ЦФО».
- **Сохранение сессии:**
  - Бот должен автоматически запоминать и поддерживать пользовательскую сессию в течении ~ 25 минут для упрощения доступа к функционалу.
- **Ролевая модель:**
  - Бот должен проверять права и разрешения пользователя для доступа к определенным функциям и данным в соответствии с ролевой моделью.

## 2. Управление запросами

- **Создание нового запроса на формирование отчета:**
  - Бот должен предоставлять возможность пользователю создавать новые запросы на формирование отчетов.

- Бот должен предоставлять список дат (текущая дата и 6 предыдущих от текущей даты), на которые необходимо сформировать запросы.
- Бот должен иметь возможность отправлять одновременно несколько запросов на выбранные даты.
- Бот должен автоматически ставить галочку на условие “Отчет только по <моим> агентам”.

### 3. Просмотр информации (VIEW)

- **История поданных запросов:**
  - Бот должен предоставлять возможность пользователю просматривать список запросов (последние 10 запросов) с указанием статуса отчета, даты/времени подачи запроса и даты, на которую формируется отчет.
- **Предоставление сформированного отчета:**
  - Бот должен предоставлять отчет в виде файлов в формате .pdf и .CSV, а также составлять описание готового отчета на основе специального [шаблона](#).

### 4. Доступ к файлам

- **Просмотр файлов:**
  - Бот должен предоставлять возможность пользователю просматривать файлы с готовым отчетом в формате .pdf и .csv.
- **Скачивание файлов:**
  - Бот должен предоставлять возможность пользователю скачивать файлы с готовым отчетом в формате .pdf и .csv.

### 5. Поддержка

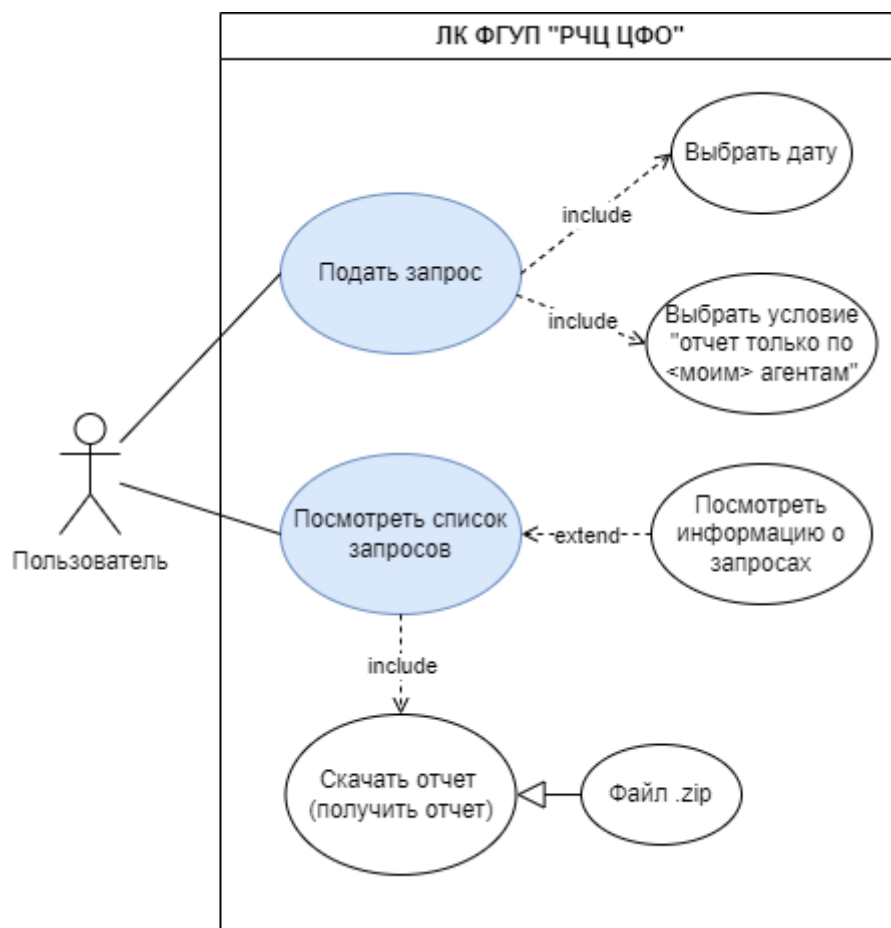
- **Описание бота и доступных команд:**
  - Бот должен предоставлять возможность пользователю ознакомиться с описанием бота и его доступных команд.
- **Feedback:**

- Бот должен предоставлять возможность оставлять отзывы.

## Модель AS IS

Текущий процесс взаимодействия пользователя с ЛК ФГУП «РЧЦ ЦФО» происходит посредством наличия следующего функционала на стороне пользователя:

1. Подать запрос на формирование отчета:
  - 1.1. Выбрать дату
  - 1.2. Выбрать условие “Отчет только по <моим> агентам”
2. Посмотреть список запросов:
  - 2.1. Скачать отчет в виде файла.zip
  - 2.2. Посмотреть информацию о запросах



## Модель TO BE

На основе потребностей пользователей и описанных функциональных требований процесс взаимодействия пользователя с ЛК ФГУП «РЧЦ ЦФО»



посредством Telegram-бота будет происходить при помощи наличия следующего функционала на стороне пользователя:

1. Подать запрос на отчет:
  - 1.1. Выбрать дату, на которую необходимо сформировать отчет
2. Получить отчет:
  - 2.1. Выбрать отчет из списка запросов
    - 2.1.1. Посмотреть информацию о запросах
  - 2.2. Скачать отчет в виде файла .csv или .pdf
3. Настроить учетную запись:
  - 3.1. Добавить учетную запись
    - 3.1.1. Ввести логин/пароль
  - 3.2. Изменить учетную запись
    - 3.2.1. Ввести логин/пароль
  - 3.3. Удалить учетную запись
4. Получить информацию о проекте:
  - 4.1. Посмотреть описание бота и доступные команды
5. Написать отзыва разработчику



## Варианты использования

### УС:Подать запрос

**Триггер:** пользователь хочет создать запрос на формирование отчета за определенную дату.

**Действующие лица:** Пользователь и Telegram-бот.

**Предусловия:**

→ Пользователь зашел в Telegram-бот.

**Постусловие:** пользователь успешно подал запрос на формирование отчета.

№ шага	Действующее лицо	Базовый сценарий (действия)	Альтернативный сценарий
1	Пользователь	<b>Нажать</b> “Меню”	
2	Telegram-бот	<b>Вызвать</b> метод <code>commands_show</code> (раскрывает список доступных команд)	
3	Пользователь	<b>Выбрать</b> пункт “Сформировать и получить отчет”	
4	Telegram-бот	<b>Вызвать</b> метод <code>command_start</code> (раскрывает список доступных команд)	
5	Пользователь	<b>Выбрать</b> пункт “Создать заявку”	<i>Сессия пользователя не активна (активная сессия ~25 минут).</i>  5.1 Telegram-бот отправляет GET-запрос скачивания картинки с капчей ( <a href="https://portal.rfc-revizor.ru/captcha/{secretcodeId}">https://portal.rfc-revizor.ru/captcha/{secretcodeId}</a> )

			<p>5.2 Пользователь вводит капчу</p> <p>5.3 Telegram-бот вызывает POST-запрос авторизации с капчей (<a href="https://portal.rfc-revizor.ru/login/">https://portal.rfc-revizor.ru/login/</a>)</p> <p>5.4 Шаги 5-9 базового сценария</p>
6	Telegram-бот	<b>Вызвать</b> GET-запрос получения данных для авторизации ( <a href="https://portal.rfc-revizor.ru/login/">https://portal.rfc-revizor.ru/login/</a> )	
7	Telegram-бот	<b>Вызвать</b> метод dates_show (раскрывает список дат)	
8	Пользователь	<b>Выбрать</b> дату отчета	
9	Telegram-бот	<b>Вызвать</b> POST-запрос создания заявки на дату ( <a href="https://portal.rfc-revizor.ru/cabinet/myclaims-reports/create">https://portal.rfc-revizor.ru/cabinet/myclaims-reports/create</a> )	

#### УС:Получить отчет

**Триггер:** пользователь хочет посмотреть готовый отчет за определенную дату.

**Действующие лица:** Пользователь и Telegram-бот.

**Предусловия:**

→ Пользователь зашел в Telegram-бот.

**Постусловие:** пользователь успешно просмотрел отчет за определенную дату.

№ шага	Действующее лицо	Базовый сценарий (действия)	Альтернативный сценарий
1	Пользователь	<b>Нажать</b> “Меню”	

2	Telegram-бот	<b>Вызвать</b> метод commands_show (раскрывает список доступных команд)	
3	Пользователь	<b>Выбрать</b> пункт “Сформировать получить отчет”	
4	Telegram-бот	<b>Вызвать</b> метод command_start (раскрывает список доступных команд)	
5	Пользователь	<b>Выбрать</b> пункт “Получить отчет”	<p>Сессия пользователя не активна (активная сессия ~25 минут).</p> <p>5.1 <b>Telegram-бот отправляет</b> GET-запрос скачивания картинки с капчей (<a href="https://portal.rfc-revizor.ru/captcha/{secretcode}">https://portal.rfc-revizor.ru/captcha/{secretcode}</a>)</p> <p>5.2 <b>Пользователь вводит</b> капчу</p> <p>5.3 <b>Telegram-бот вызывает</b> POST-запрос авторизации с капчей (<a href="https://portal.rfc-revizor.ru/login/">https://portal.rfc-revizor.ru/login/</a>)</p> <p>5.4 Шаги 5-10 базового сценария</p>
6	Telegram-бот	<b>Вызвать</b> GET-запрос получения данных для авторизации ( <a href="https://portal.rfc-revizor.ru/login/">https://portal.rfc-revizor.ru/login/</a> )	
7	Telegram-бот	<b>Вызвать</b> GET-запрос на получение списка заявок	

		(https://portal.rfc-revizor.ru/cabinet/myclaims-reports/)	
8	Пользователь	<b>Выбрать</b> отчет из списка запросов	
9	Telegram-бот	<b>Вызвать</b> GET-запрос на скачивание заявки (https://portal.rfc-revizor.ru/cabinet/claims-reports/download/{archive_id})	
10	Пользователь	<b>Получить</b> отчет в двух форматах .pdf и .csv и комментарий к отчету	<p>Сайт ФГУП «РЧЦ ЦФО» временно недоступен</p> <p>10.1 Пользователь получает файлы в формате .pdf и .csv и комментарий: «Данные обрабатываются, повторите запрос позже.»</p>

#### UC:Посмотреть информацию о запросах

**Триггер:** пользователь хочет посмотреть информацию о запросе.

**Действующие лица:** Пользователь и Telegram-бот.

**Предусловия:**

→ Успешно выполнены шаги 1-7 базового сценария UC: Получить отчет.

**Постусловие:** пользователь успешно посмотрел информацию о дате запроса, дату, на которую создается отчет, а также статус готовности и дату/время изменения статуса готовности.

№ шага	Действующее лицо	Базовый сценарий (действия)	Альтернативный сценарий
1	Пользователь	<b>Получить</b> список запросов и <b>просмотреть</b> информацию о нужном запросе: дата запроса,	

		дата, на которую создавался отчет, а также статус готовности и дата/время изменения статуса готовности	
--	--	--	--

#### UC:Скачать отчет

**Триггер:** пользователь хочет скачать готовый отчет за определенную дату.

**Действующие лица:** Пользователь и Telegram-бот.

**Предусловия:**

→ Успешно выполнен базовый сценарий UC: Получить отчет.

**Постусловие:** пользователь успешно скачал отчет за определенную дату.

№ шага	Действующее лицо	Базовый сценарий (действия)	Альтернативный сценарий
1	Пользователь	<b>Выбрать</b> формат файла с отчетом: .pdf или .csv	
2	Telegram-бот	<b>Открыть</b> просмотр файла	
3	Пользователь	<b>Скачать</b> файл на свое устройство	

#### UC:Добавить учетную запись

**Триггер:** пользователь хочет добавить данные своей учетной записи от ЛК ФГУП “РЧЦ ЦФО”.

**Действующие лица:** Пользователь и Telegram-бот.

**Предусловия:**

→ Пользователь впервые работает с Telegram-ботом.

→ Пользователь зашел в Telegram-бот.

**Постусловие:** пользователь успешно добавил данные своей учетной записи.

№ шага	Действующее лицо	Базовый сценарий (действия)	Альтернативный сценарий
1	Пользователь	<b>Нажать</b> “Меню”	<i>Пользователь выбрал пункт</i>

			<p>меню  “Сформировать и получить отчет”  1.1 <b>Telegram-бот</b> сообщает пользователю:  “Необходимо настроить аккаунт! Для этого введите команду: /settings”  1.2 <b>Пользователь</b> выбирает предложенную команду ботом.  1.3 Шаги 2-9 базового сценария</p>
2	Telegram-бот	<b>Вызвать</b> метод commands_show (раскрывает список доступных команд)	
3	Пользователь	<b>Выбрать</b> команду “Настроить учетную запись”	
4	Telegram-бот	<b>Вызвать</b> метод command_settings (раскрывает список действий с учетной записью)	
5	Пользователь	<b>Выбрать</b> пункт “Добавить”	
6	Telegram-бот	<b>Вызвать</b> метод handler_setup_account (предлагает ввести логин и пароль)	
7	Пользователь	<b>Вводит</b> логин	
8	Telegram-бот	<b>Вызвать</b> метод handler_login_from_message	
7	Пользователь	<b>Ввести</b> пароль	
8	Telegram-бот	<b>Вызвать</b> метод handler_password_from_message	Пользователь ввел неверные логин и пароль: 8.1 Telegram-бот



			сообщает, присылает ответ 302 и информационное сообщением с сайта РКН
9	Telegram-бот	<b>Сообщить</b> пользователю: "Ваши данные успешно сохранены"	

#### УС:Изменить учетную запись

**Триггер:** пользователь хочет изменить данные своей учетной записи.

**Действующие лица:** Пользователь и Telegram-бот.

**Предусловия:**

- Пользователь зашел в Telegram-бот.
- Сессия пользователя активна.

**Постусловие:** пользователь успешно изменил данные своей учетной записи.

№ шага	Действующее лицо	Базовый сценарий (действия)	Альтернативный сценарий
1	Пользователь	<b>Нажать</b> "Меню"	
2	Telegram-бот	<b>Вызвать</b> метод <code>commands_show</code> (раскрывает список доступных команд)	
3	Пользователь	<b>Выбрать</b> команду "Настроить учетную запись"	
4	Telegram-бот	<b>Вызвать</b> метод <code>command_settings</code> (раскрывает список действий с учетной записью)	
5	Пользователь	<b>Выбрать</b> пункт "Изменить"	
6	Telegram-бот	<b>Вызвать</b> метод <code>handler_setup_account</code> (предлагает ввести логин и пароль)	
7	Пользователь	<b>Вводит</b> логин	

8	Telegram-бот	<b>Вызвать</b> метод handler_login_from_message	
7	Пользователь	<b>Ввести</b> пароль	
8	Telegram-бот	<b>Вызвать</b> метод handler_password_from_message	<i>Пользователь ввел неверные логин и пароль: 8.1 Telegram-бот сообщает, присылает ответ 302 и информационно е сообщением с сайта РКН</i>
9	Telegram-бот	<b>Сообщить</b> пользователю: "Ваши данные успешно сохранены"	

#### УС:Удалить учетную запись

**Триггер:** пользователь хочет удалить данные своей учетной записи.

**Действующие лица:** Пользователь и Telegram-бот.

**Предусловия:**

- Пользователь зашел в Telegram-бот.
- Сессия пользователя активна.

**Постусловие:** пользователь успешно удалил свою учетную запись.

№ шага	Действующее лицо	Базовый сценарий (действия)	Альтернативный сценарий
1	Пользователь	<b>Нажать</b> "Меню"	
2	Telegram-бот	<b>Вызвать</b> метод commands_show (раскрывает список доступных команд)	
3	Пользователь	<b>Выбрать</b> команду "Настроить учетную запись"	
4	Telegram-бот	<b>Вызвать</b> метод command_settings (раскрывает список действий)	

		с учетной записью)	
5	Пользователь	<b>Выбрать</b> пункт “Удалить”	
9	Telegram-бот	<b>Сообщить</b> пользователю: “Учетная запись успешно удалена из базы данных!”	

**UC:Получить информацию о боте**

**Триггер:** пользователь хочет посмотреть информацию о Telegram-боте и узнать доступные команды.

**Действующие лица:** Пользователь и Telegram-бот.

**Предусловия:**

→ Пользователь зашел в Telegram-бот.

**Постусловие:** пользователь успешно ознакомился с описанием Telegram-бота и доступными командами.

№ шага	Действующее лицо	Базовый сценарий (действия)	Альтернативный сценарий
1	Пользователь	<b>Нажать</b> “Меню”	
2	Telegram-бот	<b>Вызвать</b> метод commands_show (раскрывает список доступных команд)	
3	Пользователь	<b>Выбрать</b> команду “Информация о проекте”	
4	Telegram-бот	<b>Вызвать</b> метод command_about (раскрывает описание бота и доступные команды)	
5	Пользователь	<b>Посмотреть</b> информацию о боте и список поддерживаемых команд	

**UC:Написать отзыв разработчику**

**Триггер:** пользователь хочет оставить отзыв.

**Действующие лица:** Пользователь и Telegram-бот.

**Предусловия:**

→ Пользователь зашел в Telegram-бот.

**Постусловие:** пользователь успешно отправил отзыв о работе Telegram-бота.

№ шага	Действующее лицо	Базовый сценарий (действия)	Альтернативный сценарий
1	Пользователь	Нажать “Меню”	
2	Telegram-бот	<b>Вызвать</b> метод <code>commands_show</code> (раскрывает список доступных команд)	
3	Пользователь	<b>Выбрать</b> команду “Написать отзыв разработчику”	
4	Telegram-бот	<b>Вызвать</b> метод <code>command_feedback</code> (раскрывает форму для написания отзыва)	
5	Пользователь	<b>Написать и отправить</b> отзыв	

## Нефункциональные требования

### 1. Производительность

- **Время отклика:**
  - Время отклика бота на запросы пользователя (например, отображение списка запросов) должно быть не более 2-3 секунд в 95% случаев.
  - Время отклика на простые команды (например, основные команды) должно быть не более 5 секунд.
- **Асинхронная обработка запросов:**
  - Бот должен обрабатывать не менее 50 одновременных запросов пользователей без значительного снижения производительности.
  - Бот должен корректно обрабатывать пиковые нагрузки, связанные с большим количеством одновременных пользователей.
- **Скорость выгрузки файлов:**
  - Скорость скачивания файлов из бота должна быть высокой и соответствовать скорости интернет-соединения пользователя.
- **Оптимизация базы данных:**
  - Запросы к базе данных должны быть оптимизированы для обеспечения быстрого получения необходимой информации.

## 2. Безопасность

- **Аутентификация:**
  - Бот должен обеспечивать безопасную аутентификацию пользователей, используя данные учетной записи личного кабинета ФГУП «РЧЦ ЦФО».
- **Авторизация:**
  - Бот должен проверять права доступа пользователя к запрашиваемой информации и функциям, обеспечивая доступ только к разрешенным ресурсам.
- **Безопасное хранение данных пользователя:**
  - Данные учетных записей пользователей, а также информация о последней активности, используемая ботом, должны храниться в реляционной СУБД в защищенном виде.
  - Шифрование персональных данных должно осуществляться с помощью хэш-функций.
- **Защита передаваемых данных:**
  - Обмен данными между ботом и пользователем, а также между ботом и ЛК ФГУП «РЧЦ ЦФО» должен осуществляться по защищенному каналу связи с использованием SSL/TLS.
  - Пароли и другие конфиденциальные данные не должны храниться в открытом виде.
- **Защита от несанкционированного доступа:**
  - Бот должен иметь механизмы защиты от несанкционированного доступа и взлома.
  - Должны быть предусмотрены средства для обнаружения и предотвращения DoS-атак..
- **Защита от SQL-инъекций:**
  - Если используется база данных, должны быть предприняты меры для защиты от SQL-инъекций.

## 3. Надежность:

- **Стабильная работа:**
  - Бот должен работать стабильно и без сбоев.
  - Необходимо минимизировать количество ошибок и сбоев в работе бота.
- **Резервное копирование и восстановление:**
  - Необходимо предусмотреть резервное копирование данных бота и возможность их восстановления в случае сбоев.
- **Мониторинг и логирование:**
  - Должны быть предусмотрены механизмы мониторинга работы бота, а также система логирования для отслеживания ошибок и проблем.

- Бот должна осуществлять детальное логирование всех ключевых действий пользователей в боте для последующего сбора метрик и анализа поведения.
- Логи должны включать в себя информацию о времени действия, типе действия, пользовательской идентификации, использованных параметрах и результатах выполнения операции.
- Логи должны сохраняться в формате, пригодном для обработки.
- **Обработка ошибок:**
  - Бот должен корректно обрабатывать ошибки и предоставлять пользователю понятное сообщение об ошибке.
- **Отказоустойчивость:**
  - Архитектура бота должна обеспечивать отказоустойчивость и минимальное время восстановления работоспособности после сбоев.

#### **4. Масштабируемость:**

- **Поддержка большого количества пользователей:**
  - Бот должен быть масштабируемым и способным поддерживать большее количество пользователей без значительного снижения производительности.
- **Легкость расширения функционала:**
  - Архитектура бота должна обеспечивать возможность легкого добавления нового функционала в будущем.
- **Расширяемость базы данных:**
  - Если используется база данных, то должна быть предусмотрена возможность масштабирования и ее расширения.
- **Готовность к росту объема данных:**
  - Бот должен быть готов к росту объема данных и не терять производительность.

#### **5. Удобство использования:**

- **Интуитивно понятный интерфейс:**
  - Интерфейс бота должен быть интуитивно понятным и простым в использовании для пользователей с разным уровнем технической подготовки.
- **Единый стиль:**
  - Бот должен иметь единый стиль и понятную навигацию.
- **Доступность:**
  - Бот должен быть доступен 24/7.

#### **6. Сопровождаемость:**

- **Код:**

- Код должен быть хорошо документирован.
- Код должен быть легко читаемым и понятным для разработчиков.
- **Развертывание:**
  - Процесс развертывания бота должен быть простым и понятным.
  - Процесс обновления бота должен быть максимально автоматизирован.
- **Тестирование:**
  - Бот должен быть покрыт автоматическими тестами.

## Права и роли

Роль	Действие
Пользователь (авторизованный)	<ol style="list-style-type: none"> <li>1. Сформировать заявку</li> <li>2. Получить отчет</li> <li>3. Настроить учетную запись</li> <li>4. Получить информацию о проекте</li> <li>5. Написать отзыв разработчику</li> </ol>
Пользователь (неавторизованный)	<ol style="list-style-type: none"> <li>1. Получить информацию о проекте</li> <li>2. Написать отзыв разработчику</li> </ol>
Администратор	<ol style="list-style-type: none"> <li>1. Сформировать заявку</li> <li>2. Получить отчет</li> <li>3. Настроить учетную запись</li> <li>4. Получить информацию о проекте</li> <li>5. Получить фидбек</li> <li>6. Изменить имя, описание, изображение, команды, политику приватности.</li> </ol>

## Метрики

Необходимы следующие метрики:

1. Общее кол-во пользователей
2. Рейтинг. Пользователь - кол-во обращений к боту
3. Кол-во кликов кнопок (/start, /...)
4. Рейтинг. Ссылка РКН - кол-во запросов

### Взаимодействие с ботом

Метрика	Описание
rkn_bot_command	Текстовое сообщение (/start, /help, ...)
rkn_bot_callback	Действие callback (btn_crt, btn_get, ...)
rkn_server_	Взаимодействие с сервером РКН
rkn_db_	Взаимодействие с базой данных
rkn_common_	Общие характеристики

## Получение доступа к сайту

### Проверка данных в БД:

- Наличие данных учетной записи
  - ☒ В наличие: Перейти к следующему пункту
  - ☐ Отсутствуют: Перейти к шагу настройки УЗ.
- Дата и время последней активности куков
  - ☐ Активность больше 25 min: Перейти к шагу 2.
  - ☒ Активность меньше 25 min: Перейти к шагу "Основные кнопки действий".

### Авторизация:

- Отправить GET-запрос на получение данных авторизации и ссылки на капчу
  - ☒ Успешно: Данные получены и сохранены. Сформирован url капчи.
  - ☐ Ошибка: Вывод сообщения "Сайт не работает, повторите попытку позже"
- Отправить GET-запрос на скачивание картинки с капчей
  - ☒ Успешно: Картинка загружена на диск.
  - ☐ Ошибка: Вывод сообщения "Сайт не работает, повторите попытку позже"
- Пользователь ввел капчу с изображения
- Отправить POST-запрос авторизации
  - ☒ Успешно: Всё ОК. (данные в БД обновлены)
  - ☐ Ошибка: Вывод сообщения об ошибке с сайта (повтор ввода капчи / настройка УЗ / сайт не работает).



## Основные кнопки действий, доступные пользователю

### Кнопка **"/start"** -> **"Создать заявку"**

1. Проверка сессии
  - ☒ Успешно: Перейти к следующему шагу
  - ☐ Ошибка: Перейти к шагу авторизации
2. Вывод кнопок пользователю для выбора даты создания заявки.
3. Отправить POST-запрос с созданием заявки
  - ☒ Заявка создана: Всё ОК
  - ☐ Заявка не создана: Вывод сообщения об ошибке с сайта.  
(необходима авторизация *шаг 2* или сайт не работает)
4. Вывод информационного сообщения о создании заявки

### Кнопка **"/start"** -> **"Получить заявку"**

1. Проверка сессии
  - ☒ Успешно: Перейти к следующему шагу
  - ☐ Ошибка: Перейти к шагу авторизации
2. Отправить GET-запрос на получение статуса отчетов
  - ☒ Успешно: Получен список заявок со статусом
  - ☐ Ошибка: Вывод сообщения об ошибке с сайта. (необходима авторизация *шаг 2* или сайт не работает)
3. Вывод кнопок пользователю с доступными для скачивания отчетов
4. Отправить GET-запрос на скачивание отчета
  - ☒ Успешно: Отчет скачен и выгружен пользователю
  - ☐ Ошибка: Вывод сообщения об ошибке с сайта. (необходима авторизация *шаг 2* или сайт не работает)

### Кнопка **"/settings"**

1. Проверка наличия УЗ в БД
2. Варианты действий пользователя:
  - Добавить данные аккаунт РКН в бот -> Перейти к шагу 3
  - Изменить сохраненные данные об аккаунте РКН в боте -> Перейти к шагу 3
  - Удалить аккаунт - Полностью удалить всю информацию
3. Ввод "Логина" УЗ
  - ☒ Введен: перейти к следующему шагу
  - ☐ Отмена: завершить настройку
4. Ввод "Пароля" УЗ
  - ☒ Введен: перейти к следующему шагу
  - ☐ Отмена: завершить настройку

5. Сообщение "Данные аккаунта сохранены в БД"
6. Вывод кнопок: "Создать заявку", "Получить заявку".

## **Кнопка `"/about"`**

1. Краткое описание проекта и предназначение бота

## **Кнопка `"/feedback"`**

1. Возможность пользователю написать сообщение разработчику

## **Запросы, отправляемые на сервер**

1. Авторизация:
  - 1.1. GET-запрос получения данных для авторизации (<https://portal.rfc-revizor.ru/login/>)
  - 1.2. GET-запрос скачивания картинки с капчей (<https://portal.rfc-revizor.ru/captcha/{secretcodeId}>)
  - 1.3. POST-запрос авторизации с капчей (<https://portal.rfc-revizor.ru/login/>)
2. POST-запрос создания заявки на дату (<https://portal.rfc-revizor.ru/cabinet/myclaims-reports/create>)
3. GET-запрос на получение списка заявок (<https://portal.rfc-revizor.ru/cabinet/myclaims-reports/>)
4. GET-запрос на скачивание заявки ([https://portal.rfc-revizor.ru/cabinet/claims-reports/download/{archive\\_id}.zip](https://portal.rfc-revizor.ru/cabinet/claims-reports/download/{archive_id}.zip))

Приложения

Приложение 1. Алгоритм оформления комментария к полученной заявке.

Отчет для оператора связи по имеющимся нарушениям									
Отчет о доступности запрещенных сайтов									
по оператору: ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ "ТЕОРЕМА ТЕЛЕКОМ" (ИНН: 7811152127)									
На дату: 17 Дек, 2024									
Сформирован: 20 Дек, 2024 10:04:51									
Категория	Количество нарушений (записей реестра)	% от общего количества записей категории в реестре							
Первомайский районный суд г. Омска	1	100%							
Благовещенский городской суд Амурской области	1	1%							
Роскомнадзор	1	<1%							
Всего	3	-							
Время обнаружения нарушения по часовой поясу региона, где установлен агент	Зонд (узел)	Почтовый адрес места установки агента	Координаты места установки агента	ЮЗ записи реестра	Идентификатор (URL/domain/IP)	Тип информации	Время включения записи в реестр (UTC)	HTTP статус	Адрес перенаправления
17 Дек, 2024 16:54:44	СЗФО_Б_3_00_ООО Теорема Телеком	192019, Санкт-Петербург, Пискаревский пр-кт, д.2, к.2, лит.Щ	30°24'22"E 59°57'33"N	5188088	http://secretitaliaclub.net, 52.223.13.41, GET	Роскомнадзор	15 Дек, 2022 10:13:19	200	
17 Дек, 2024 22:29:53	СЗФО_Б_3_00_ООО Теорема Телеком	192019, Санкт-Петербург, Пискаревский пр-кт, д.2, к.2, лит.Щ	30°24'22"E 59°57'33"N	1469880	https://www.labyrics.com/Музыка-Belux-chords-lyrics-order, 104.21.2.150, GET	Первомайский районный суд г. Омска	22 Апр, 2019 11:17:26	404	
17 Дек, 2024 22:29:53	СЗФО_Б_3_00_ООО Теорема Телеком	192019, Санкт-Петербург, Пискаревский пр-кт, д.2, к.2, лит.Щ	30°24'22"E 59°57'33"N	1469880	https://www.labyrics.com/Музыка-Belux-chords-lyrics-order, 172.67.129.84, GET	Первомайский районный суд г. Омска	22 Апр, 2019 11:17:26	404	
17 Дек, 2024 22:36:14	СЗФО_Б_3_00_ООО Теорема Телеком	192019, Санкт-Петербург, Пискаревский пр-кт, д.2, к.2, лит.Щ	30°24'22"E 59°57'33"N	1675571	https://www.labyrics.com/%D0%9F%D1%80%D0%BE%D1%81%D1%88%D0%BF%D0%B0%D0%B9%D1%82%D0%B5%D1%81%D1%8C%21-chords-lyrics-25%252F17, 104.21.2.150, GET	Благовещенский городской суд Амурской области	08 Авг, 2019 15:41:52	404	
17 Дек, 2024 22:36:14	СЗФО_Б_3_00_ООО Теорема Телеком	192019, Санкт-Петербург, Пискаревский пр-кт, д.2, к.2, лит.Щ	30°24'22"E 59°57'33"N	1675571	https://www.labyrics.com/%D0%9F%D1%80%D0%BE%D1%81%D1%88%D0%BF%D0%B0%D0%B9%D1%82%D0%B5%D1%81%D1%8C%21-chords-lyrics-25%252F17, 172.67.129.84, GET	Благовещенский городской суд Амурской области	08 Авг, 2019 15:41:52	404	

На основе файла:

Информация	Комментарий	Пример
Если A>10	ДАТА A = X, где X - значение из поля Всего	12.04.2021 A=31
Если любое из B=>1%	ДАТА B = Y, где Y - максимальное значение B	16.05.2021 B=2%

Если видим надпись “Данные обрабатываются. Повторите запрос позже”	Данные обрабатываются. Повторите запрос позже.	
В остальных случаях	ОК.	23.12.2021 ОК.