Хакатон «Лидеры цифровой трансформации 2023»

Задача 4: Сервис обработки обратной связи пользователей «Московского постамата» от Агентства инноваций Москвы и Московского постамата

РЕШЕНИЕ КОМАНДЫ «ДАШБОРДИЗАЦИЯ»

Оглавление

[1. Задача 3](#_Toc137334169)

[2. Решение 4](#_Toc137334170)

[3. Основной сервис 5](#_Toc137334171)

[1. WEB-сайт 5](#_Toc137334172)

[1. Раздел «Навигация» 6](#_Toc137334173)

[2. Раздел «Статистика» 6](#_Toc137334174)

[3. Раздел «Отзывы» 8](#_Toc137334175)

[4. Раздел «Тикеты» 9](#_Toc137334176)

[5. Раздел «Постаматы» 11](#_Toc137334177)

[6. Раздел «Партнёры» 13](#_Toc137334178)

[7. Раздел «API» 14](#_Toc137334179)

[2. Запуск сервиса 16](#_Toc137334180)

[4. Нейросеть 17](#_Toc137334181)

[5. Сервис сбора статистики 18](#_Toc137334182)

[6. Чат-бот в Telegram 19](#_Toc137334183)

[7. Приложение для постамата 20](#_Toc137334184)

[8. Планы по развитию системы 22](#_Toc137334185)

# Задача

Разработать сервис сбора и обработки обратной связи от пользователей «Московского постамата» для управления качеством обслуживания на всех этапах: от оформления интернет-заказа до получения в постамате.

# Решение

Для решения задачи была выбрана микросервисная архитектура.

Микросервисная архитектура позволяет собрать общую систему из разных независимых модулей. Это позволяет не только вносить изменения в каждый модуль, не затрагивая другие, но и оптимизировать работу, предоставив каждому модулю или группам модулей персональные машины с наиболее подходящими параметрами.

В нашей задаче используются следующие компоненты:

* Сервис, обеспечивающий связь со всеми компонентами системы, предоставляющий API и графический интерфейс пользователя
* Нейросеть, обученная распознавать тональность сообщений и определять тип проблем
* Сервис сбора статистики
* Чат-бот в Telegram для сбора обратной связи
* Приложение для сбора обратной связи непосредственно через постамат

Для хранения данных используется СУБД «PostgreSQL»

# Основной сервис

Особенности реализации:

* Язык программирования Java 11
* Фреймворк Apache Maven
* Фреймворк Spring Framework
* REST API
* Язык разметки Thymeleaf

Сервис предоставляет графический пользовательский интерфейс для взаимодействия со всеми методами.

Сервис предоставляет четыре основных контроллера:

* Для работы с сообщениями (отзывами)
* Для работы с заявками (тикетами)
* Для работы с постаматами
* Для работы с партнёрами

В сервисе реализованы следующие подсервисы:

* Сервис обработки сообщений
* Сервис обработки тикетов
* Сервис обработки постаматов
* Сервис обработки партнёров
* Сервис чтения документов

## WEB-сайт

Сайт состоит из семи основных разделов:

* Навигация
* Статистика
* Отзывы
* Тикеты
* Постаматы
* Партнёры
* API

Также в верхнем меню имеется кнопка перехода в чат Telegram, где бот поможет ответить на важные вопросы.

Переход между разделами осуществляется нажатием на необходимую кнопку в верхней навигационной панели.

### Раздел «Навигация»

Главная страница содержит краткую информацию о ресурсе.

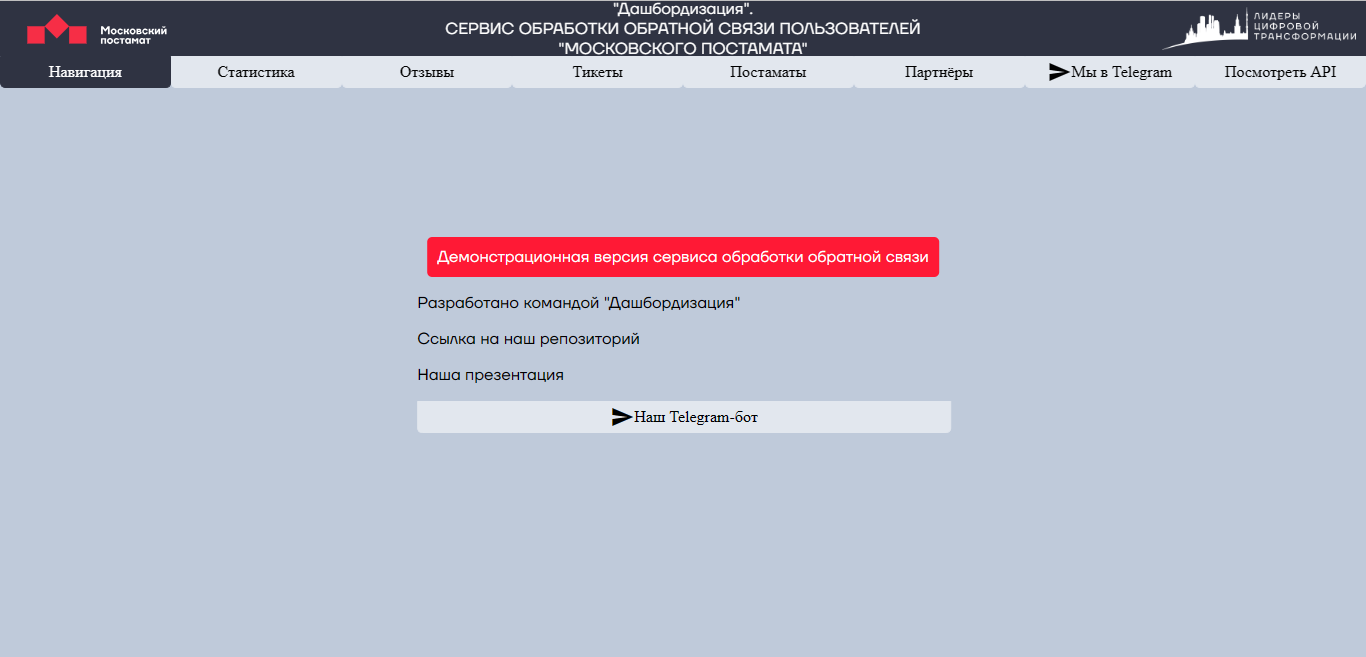


Рисунок Страница "Навигация"

### Раздел «Статистика»

В данном разделе отображаются разные графики и диаграммы, также имеется возможность скачивать отчёты.

Доступ в раздел осуществляется по логину и паролю. Необходимые для входа логин и пароль указаны на странице.

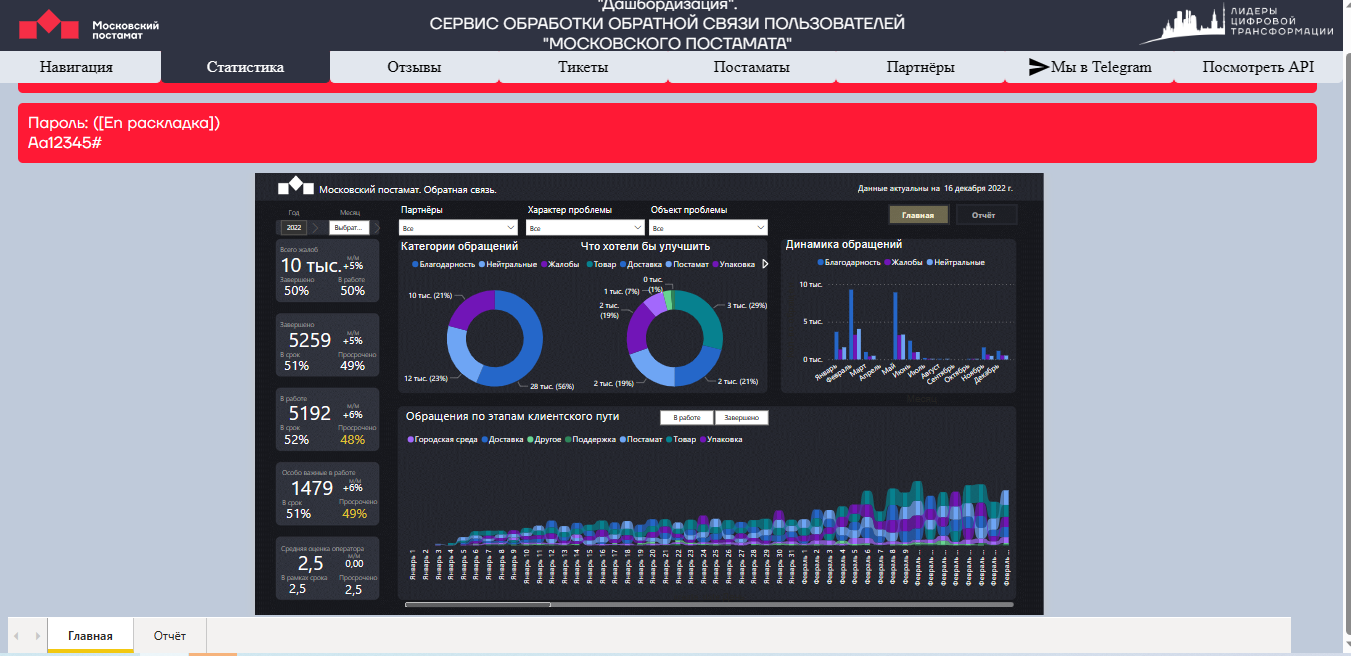


Рисунок Страница "Статистика"

Имеется возможность фильтровать данные по любым полям, выбирать конкретные временные промежутки, настраивать режим отображения.

Также есть окно отчётов.

Генерируемые отчёты можно выгружать. В сгенерированном отчёте также доступна фильтрация данных.

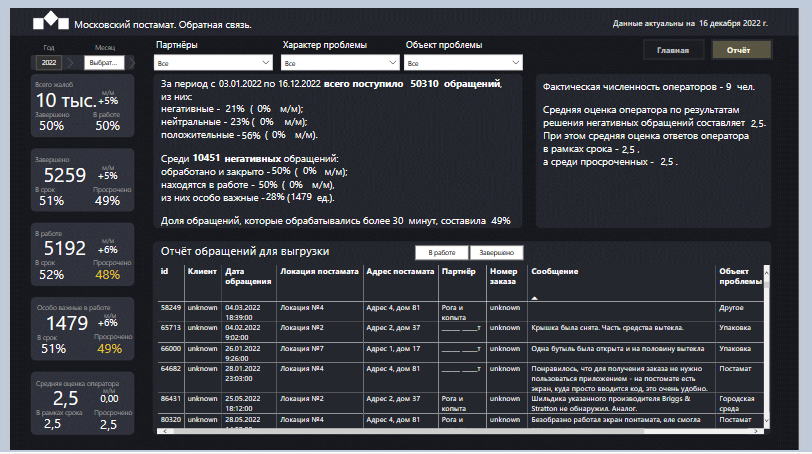


Рисунок Страница "Статистика", раздел "Отчёт"

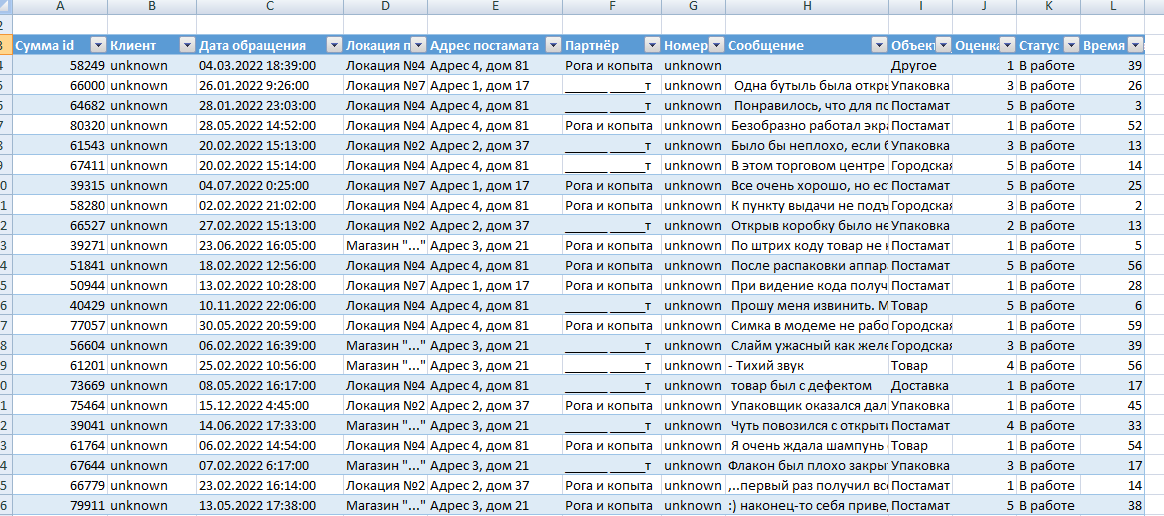


Рисунок Вид сгенерированного отчёта

### Раздел «Отзывы»

Позволяет просматривать отзывы, связанные с ними тикеты, постаматы. Возможно добавить как один отзыв, так и импортировать множество отзывов через загрузку csv-файла.

Максимальный размер загружаемого файла определяется при запуске сервиса.

Доступно ручное изменение категории проблемы.

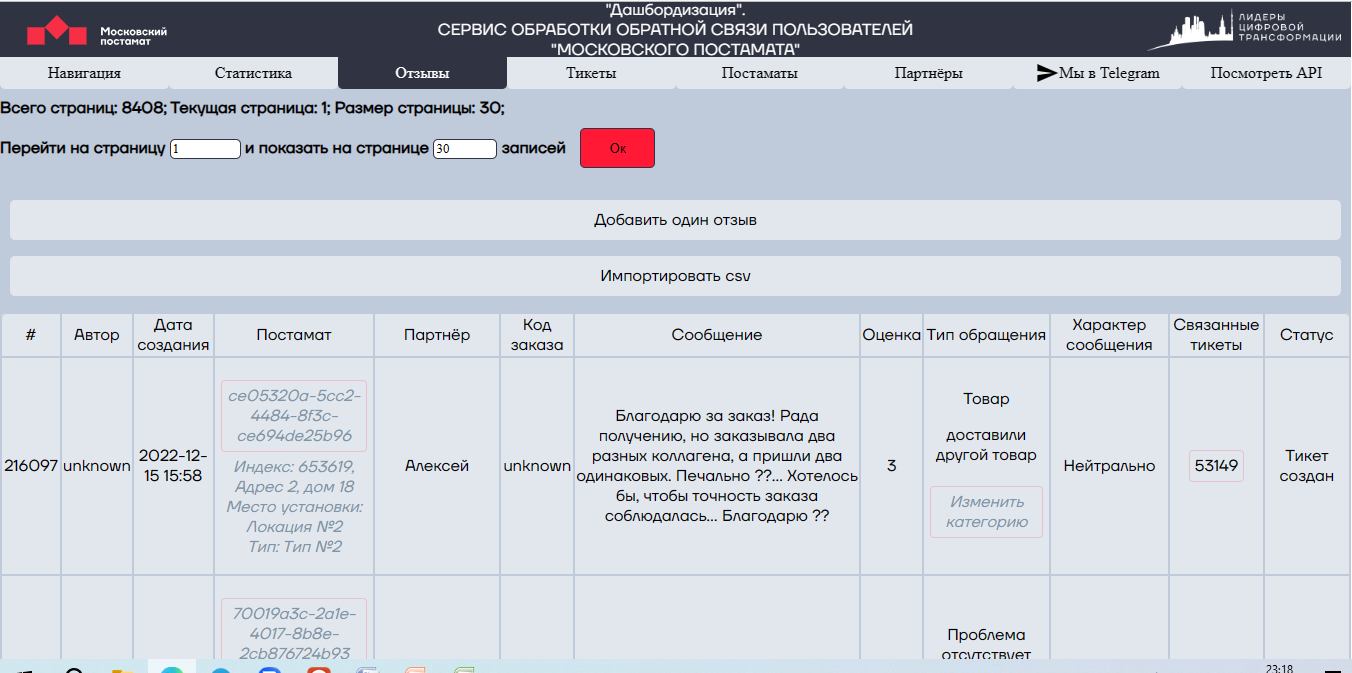


Рисунок Страница "Отзывы"

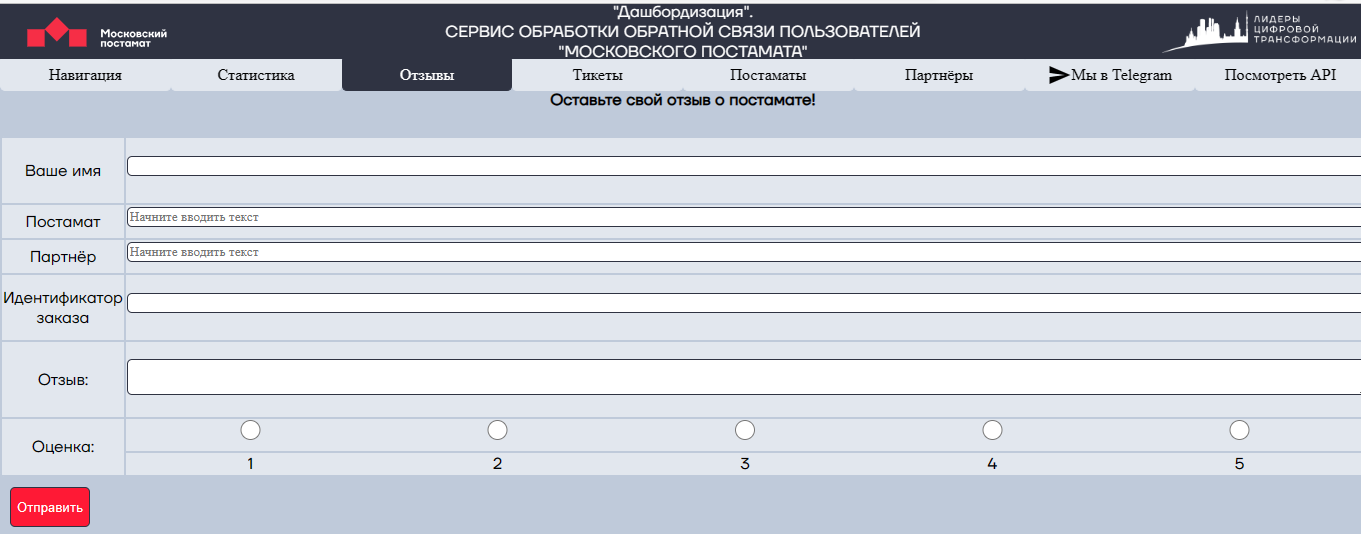


Рисунок Страница "Добавить один отзыв"

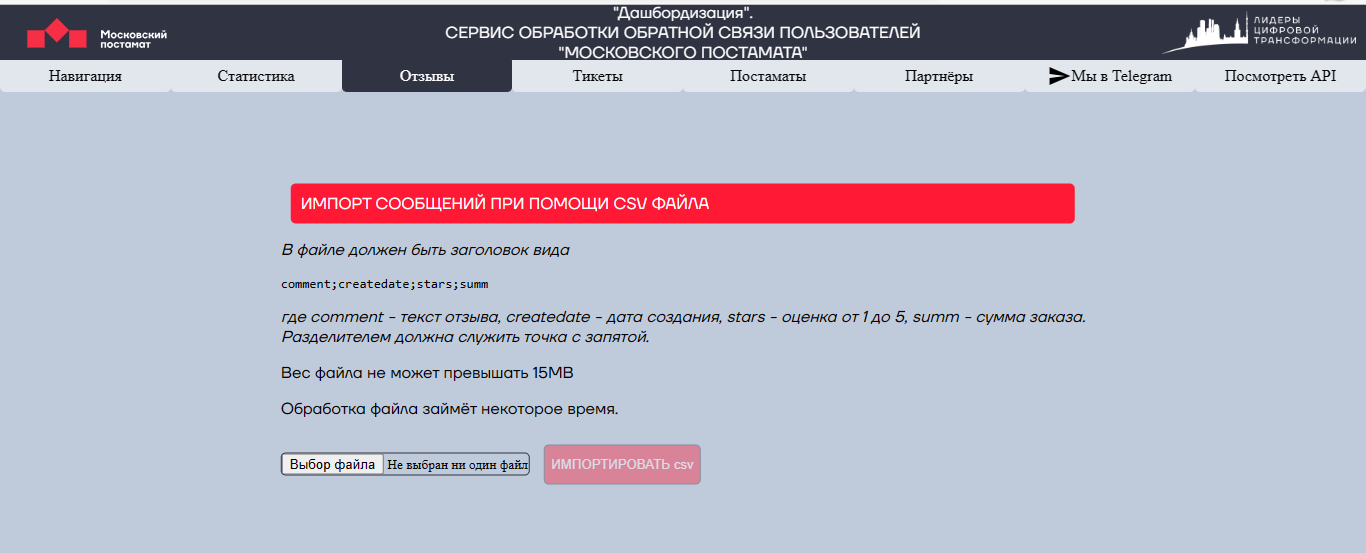


Рисунок Страница "Импортировать отзывы"

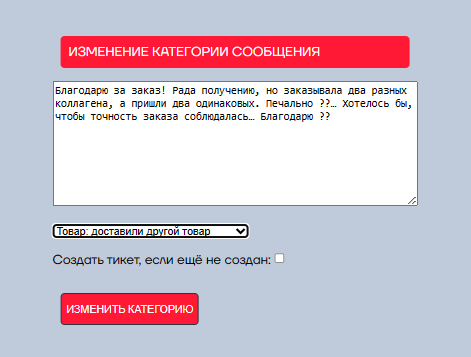


Рисунок Окно изменения категории проблемы

### Раздел «Тикеты»

Доступен просмотр заведённых в системе тикетов, перход к партнёрам или сообщениям, с которыми тикеты связаны.

Имеется возможность добавлять к тикетам какие-то комментарии, рабочие заметки.

Имеется возможность закрывать тикеты.

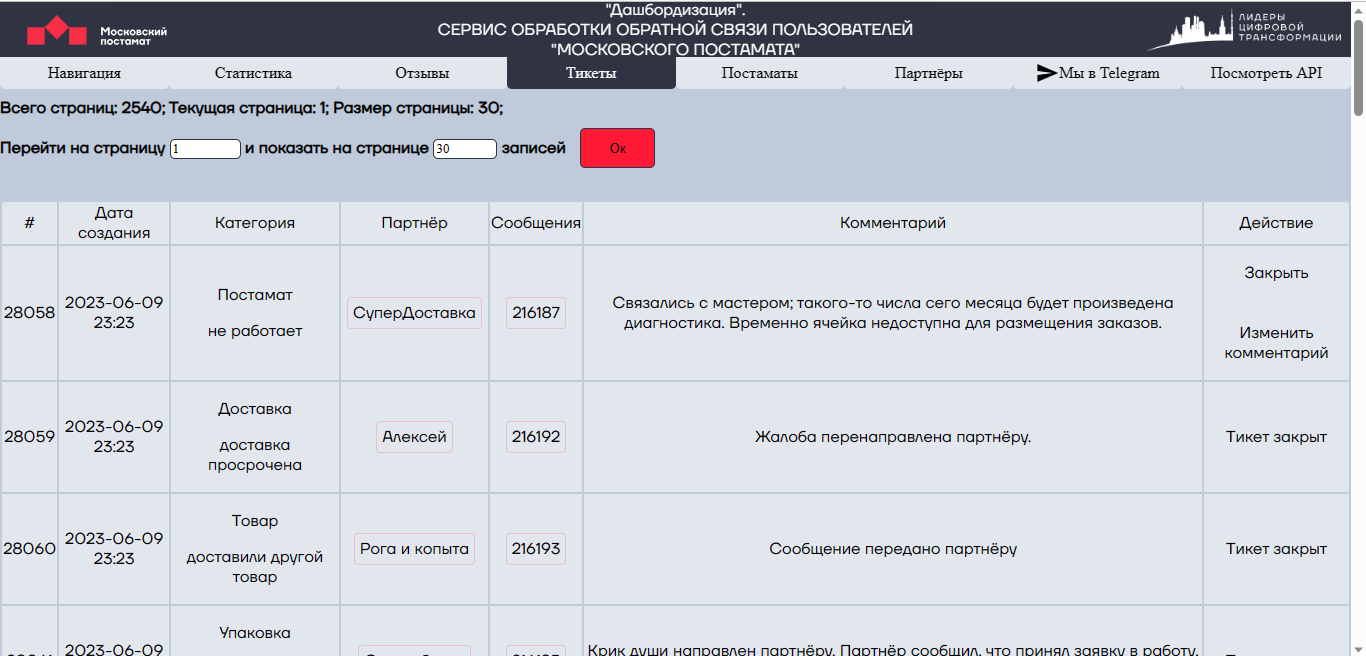


Рисунок Страница "Тикеты"

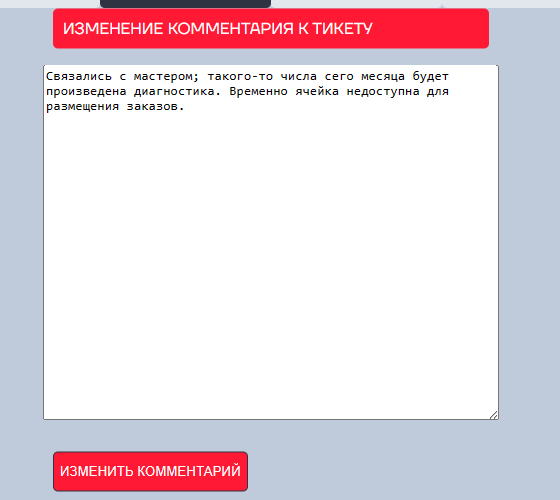


Рисунок Страница изменния комментария к тикету

### Раздел «Постаматы»

Позволяет просматривать список постаматов, добавлять новые, изменять существующие.

При удалении постамата во всех сообщениях, где он упоминался, будет стоять пометка «Работа постамата приостановлена».

Для каждого постамата выводится процент недовольства. Он равен проценту негативных отзывов, связанных с конкретным постаматом, от общего числа отзывов на этот постамат.

Доступен просмотр постаматов на карте.

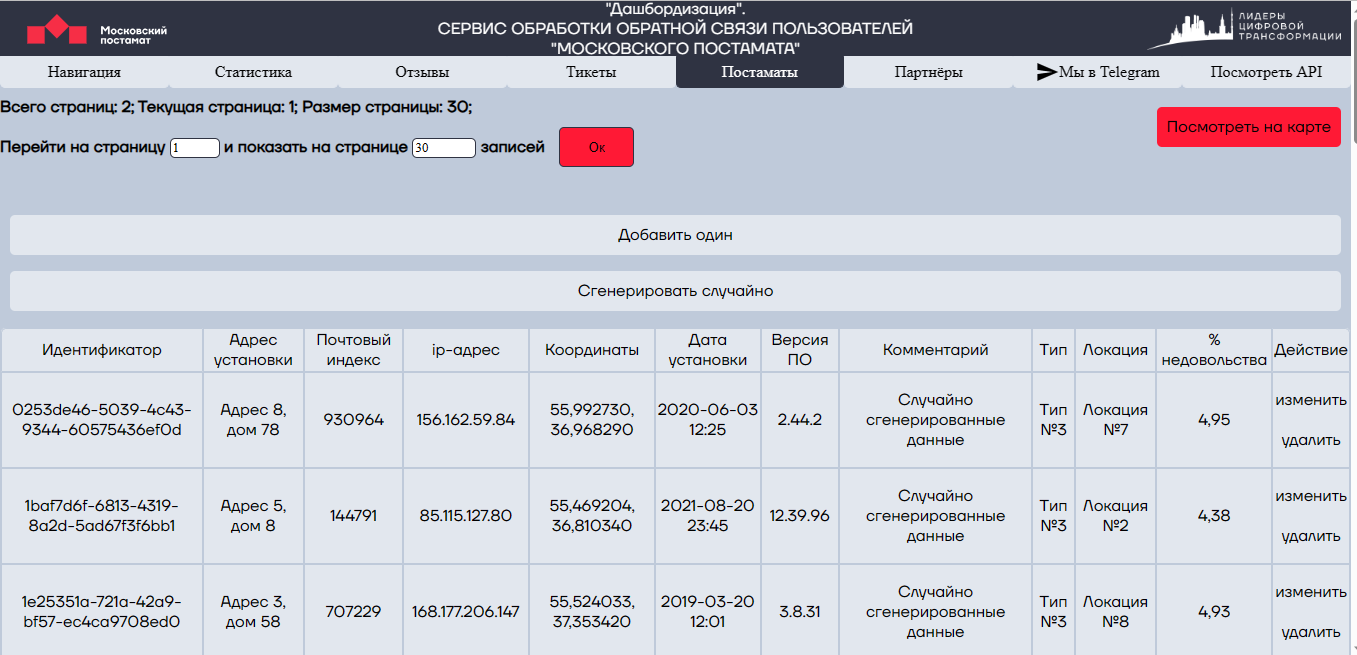


Рисунок Страница "Постаматы"

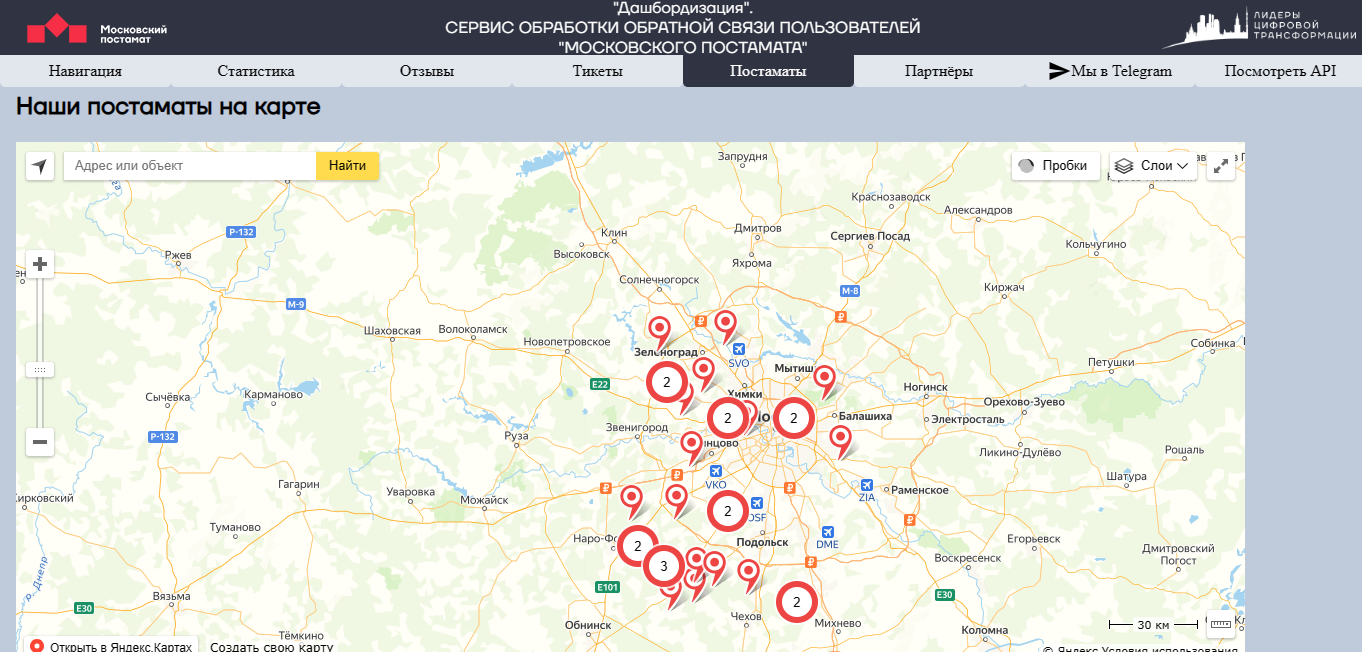


Рисунок Постаматы на карте

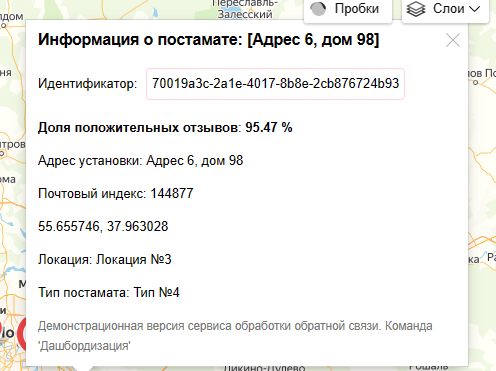


Рисунок Информация о постамате на карте

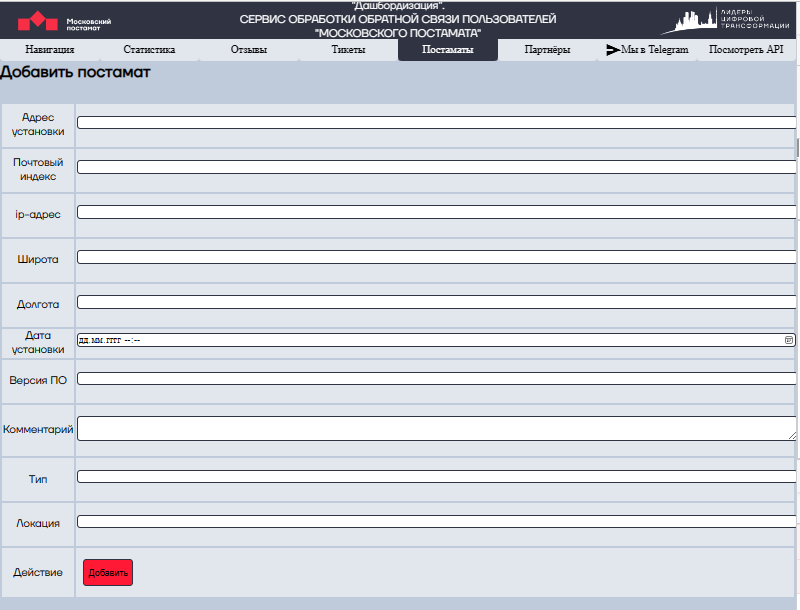


Рисунок Страница добавления одного постамата

### Раздел «Партнёры»

Позволяет просматривать список партнёров, добавлять новых, изменять существующих.

При удалении партнёра во всех сообщениях, где он упоминался, будет стоять пометка «Сотрудничество прекращено».

Для каждого партнёра выводится процент недовольства. Он равен проценту негативных отзывов, связанных с конкретным партнёром, от общего числа отзывов на этого партнёра.

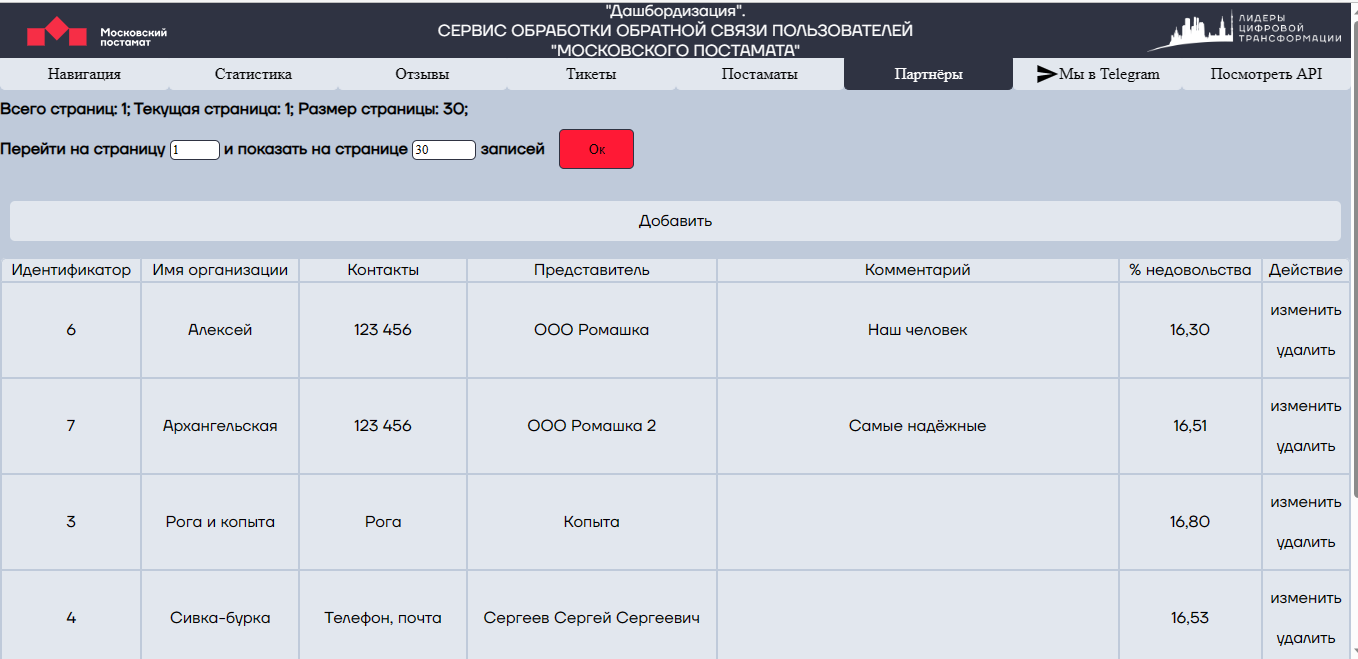


Рисунок Страница "Партнёры"

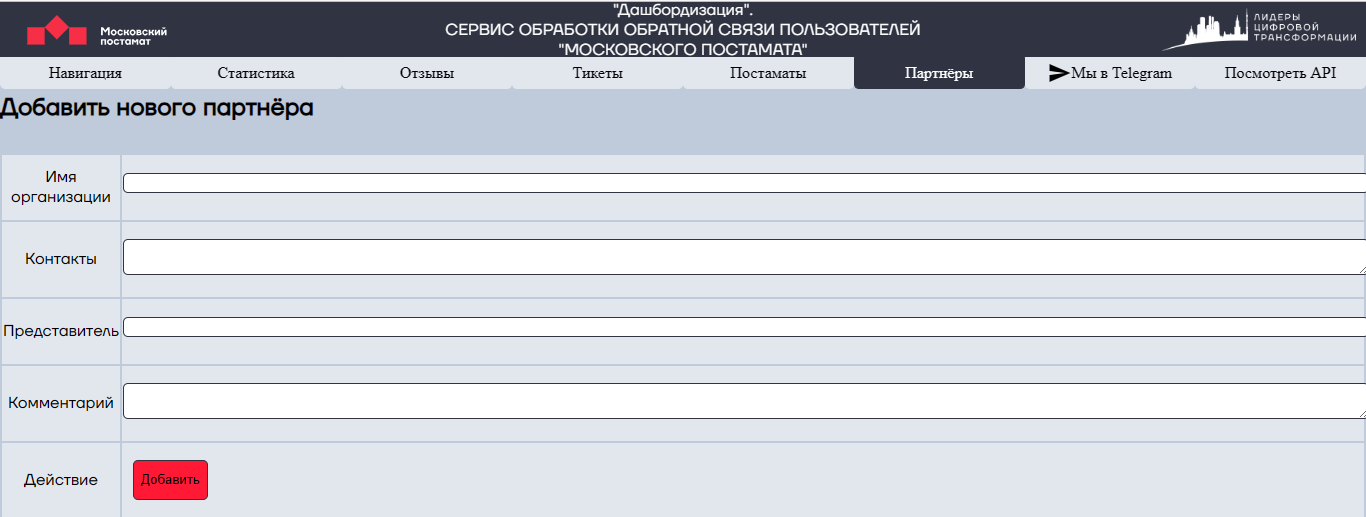


Рисунок Страница добавления партнёра

### Раздел «API»

Данная страница генерируется автоматически плагином Swagger и представляет собой удобный набор API-методов. Удобство заключается в наличии графической оболочки, валидации полей и возможности добавлять комментарии.

Swagger умеет генерировать документацию по API и защищать определённые методы от пользователей согласно ролям.

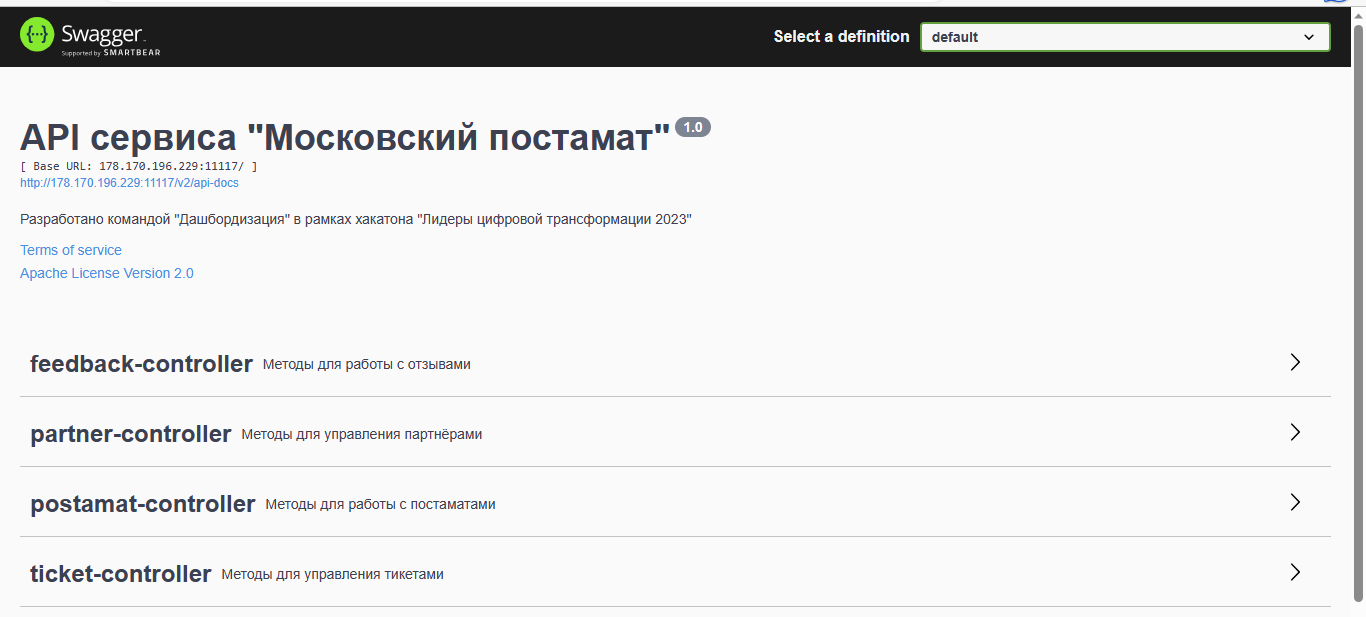


Рисунок Swagger: список контроллеров

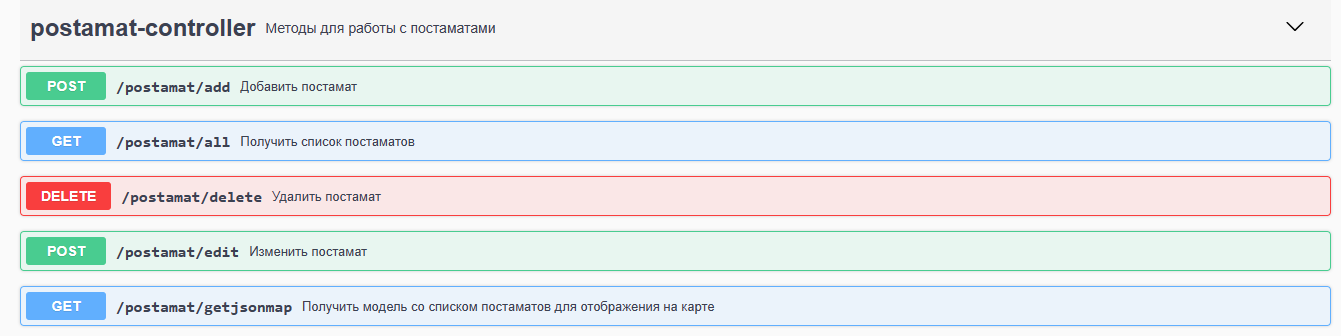


Рисунок Swagger: Методы в контроллере

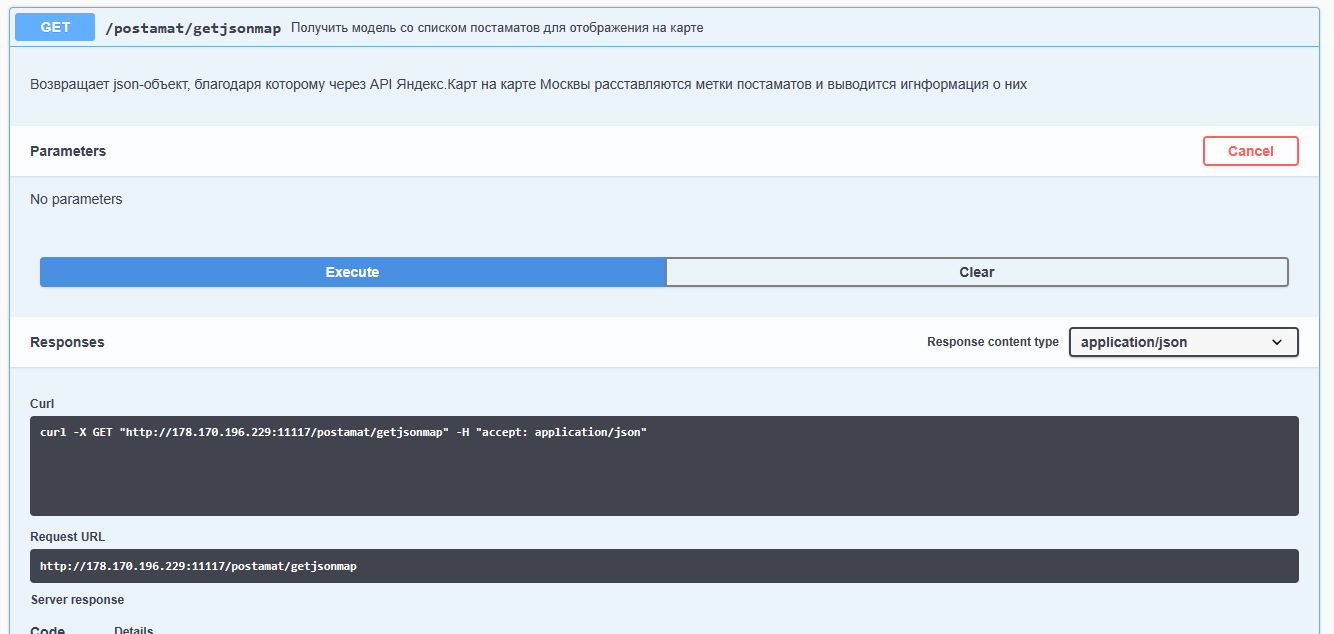


Рисунок Swagger: пример отправки запроса

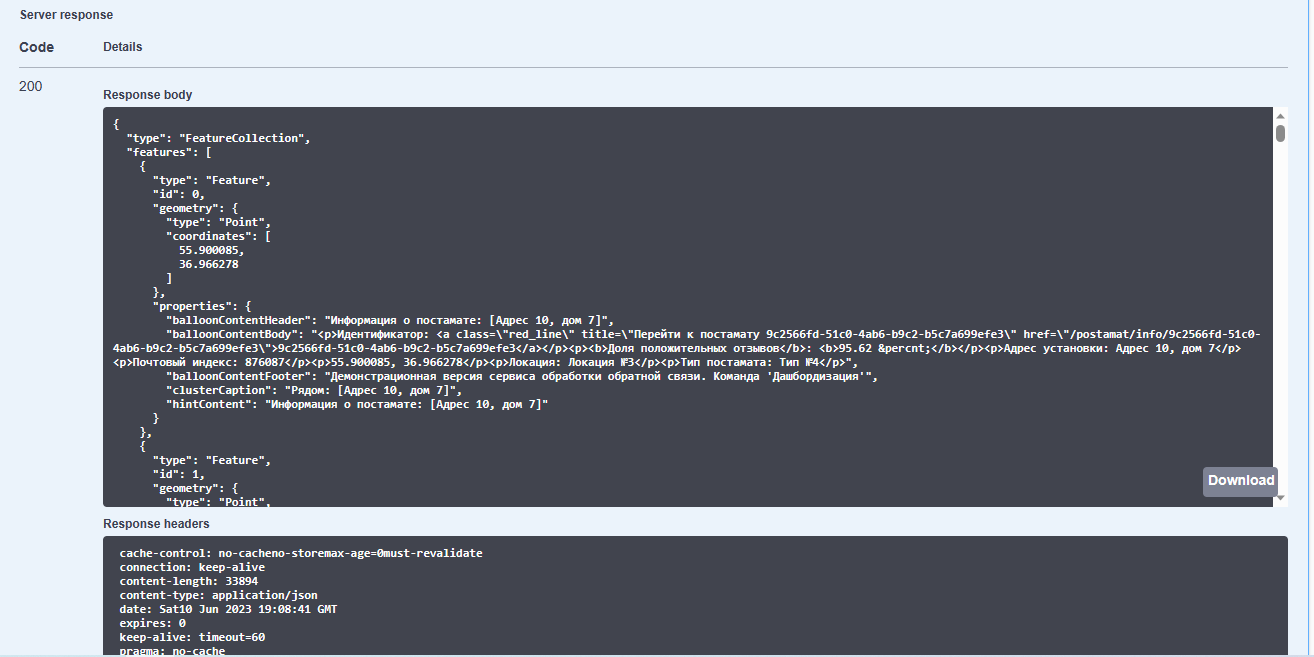


Рисунок Swagger: Пример ответа на запрос

## Запуск сервиса

Для запуска сервиса на сервере необходимо сперва собрать образ. После этого возможно как ручное развёртывание образа, так и запуск с помощью docker-compose.

|  |  |  |
| --- | --- | --- |
| Ключ | Значение по умолчанию | Описание |
| SERVER\_PORT | 11111 | Порт, на котором запустится сервис |
| DB\_HOST | localhost | Хост БД |
| DB\_USERNAME | root | Имя пользователя БД |
| DB\_PASSWORD | root | Пароль пользователя БД |
| PY\_HOST | localhost | Хост сервиса с нейросетью |
| MAX\_REQUEST\_SIZE | 15MB | Максимальный размер загружаемого файла |
| YA\_MAP\_API\_KEY | - | API-ключ для работы с API Яндекс.Карт |
| FSPAM\_SEC | 120 | Раз во сколько секунд выводить в лог состояние памяти сервиса |
| RECHECK\_STATUSES\_SEC | 5 | Раз во сколько секунд проверять сообщения и передавать их на обработку далее |
| RECHECK\_NEGATIVE\_MIN | 120 | Раз во сколько минут пересчитывать % негативных отзывов для партнёров и постаматов |

# Нейросеть

Реализована на языке Python

В качестве модели для ML используется Логистическая регрессия.

Предварительно массив отзывов очищается от специальных символов и векторизируется по методу Tfidf

С помощью кроссвалидации моделей определяется наиболее релевантные параметры для обучения модели

Оценка модели проводится по показателю Accuracy.

Результаты моделей на обучающих выборках

* модель определения тональности accuracy = 0,867
* модель определения типа проблемы accuracy = 0.818

# Сервис сбора статистики

Для сбора статистики и формирования дашбордов используется платформа Power BI.

Данные для анализа платформа получает напрямую из базы данных.

В текущей версии сервиса обработки обратной связи дашборды со статистикой выводятся на страницу «Статистика».

Платформа позволяет генерировать отчёты разных типов в зависимости от потребностей пользователя. В демонстрационных целях в данной версии сервиса для ознакомления с возможностями системы предлагается отчёт по поступившим от пользователей сообщениям.

# Чат-бот в Telegram

В качестве дополнительного средства сбора обратной связи предлагается чат-бот Telegram. Он может ответить на разные вопросы пользователя или по необходимости перенаправить к оператору.

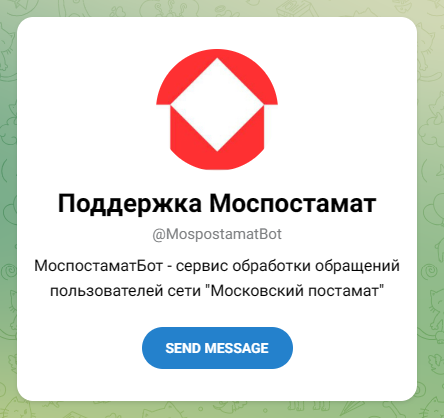


Рисунок Предложение перейти в чат

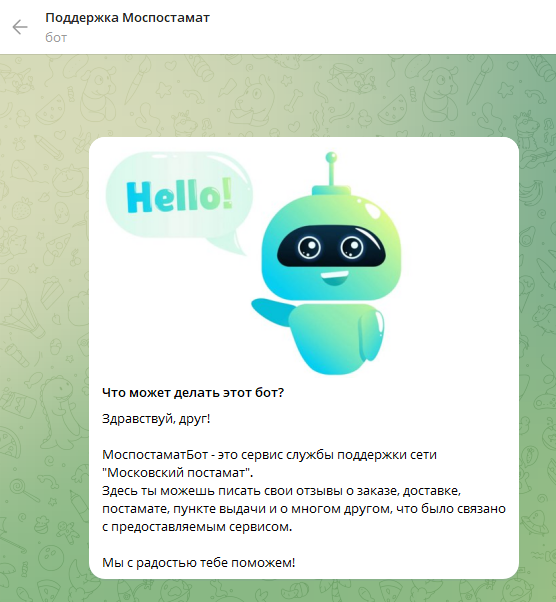


Рисунок Приветственное сообщение

# Приложение для постамата

Реализация – HTML, JS, CSS;

Для сборки приложения – Node.js, Electron

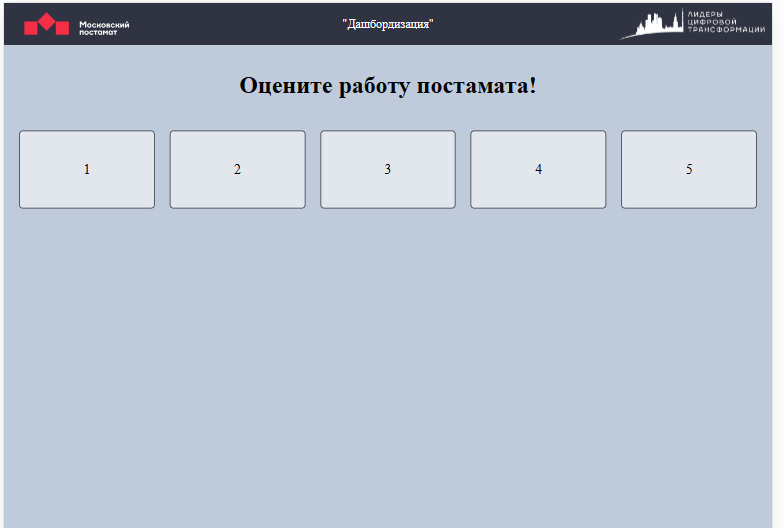


Рисунок Приложение для постамата. Главная страница

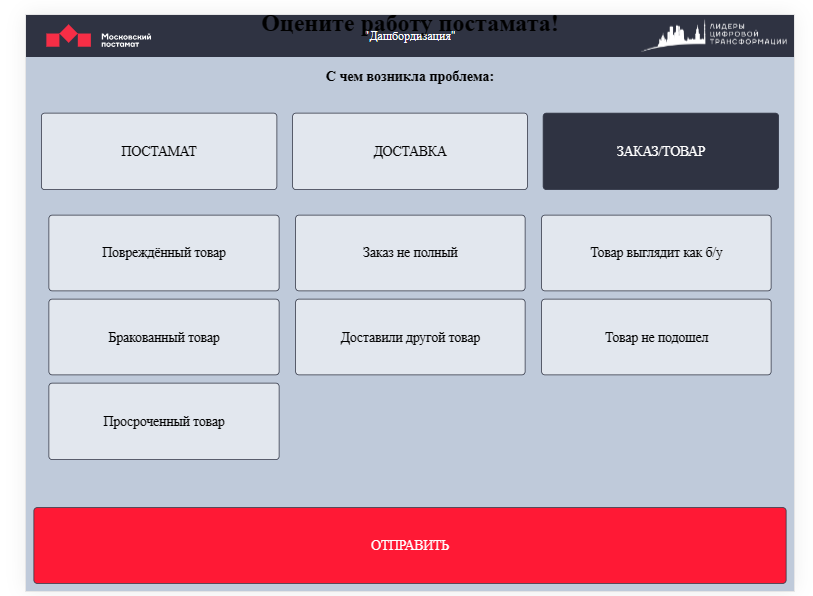


Рисунок Приложение для постамата. Выбор проблемы из категории "Заказ/Товар"

Пользователю предлагается выбрать оценку. Если оценка низкая, то предлагается выбрать, с чем именно возникли трудности.

При нажатии на кнопку с общей категорией проблем появляется выбор с расширенным перечнем проблем.

При нажатии на кнопку «Отправить» сообщение отправляется в общий сервис обработки обратной связи.

# Планы по развитию системы

В рамках развития системы предлагается:

1. Произвести интеграцию с сервисом авторизации, чтобы сервисы имели общую информацию о пользователях системы.
2. Произвести интеграцию с существующими сервисами, которые на данный момент содержат информацию о постаматах, партнёрах и текущих заказах.
3. Произвести интеграцию с партнёрскими сервисами типа HelpDesk для передачи заявок, не имеющих отношения к Московскому Постамату.
4. Добавить функцию массового анализа проблем для генерации групповых тикетов. Например, выявить постоянные проблемы с каким-то пунктом выдачи.
5. Добавить функцию, которая из сообщений, где категория проблемы была изменена оператором вручную, будет формировать отдельные документы для дообучения нейросети.
6. Расширить API