|  | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |
| --- | --- |

**Факультет «Информатика и системы управления»**

**Кафедра ИУ5 «Системы обработки информации и управления»**

Дисциплина «Разработка интернет-приложений»

Расчетно-пояснительная записка

Тема: «Поставки комплектующий для производства электроавтомобилей»

Студент: Белозеров Д. С.

Группа ИУ5-54Б

Преподаватель: Канев А.И.

2024г.

# **ВВЕДЕНИЕ**

Поставка комплектующих для производства автомобилей является ключевым звеном для автомобильной отрасли.

Так по данным EV-volumes, который под термином «Электрокар» понимает как чисто электрические автомобили, так и гибриды, в 2014 году объем рынка «электрокаров» в период с 2014 по 2020 год вырос почти в 19 раз. Со 170 тыс. до 3.2 миллиона в год. При этом средний ежегодный рост на всем протяжении этого периода составлял 50+-10%, что говорит об экспоненциальном росте.

Сфера электромобилей в России стремительно развивается, выходя на рекордные значения. Так, только за март 2023 года автолюбители приобрели 702 новых электрокара – больше, чем за весь 2020 год.

Доля «зелёных» электромобилей в России также увеличивается. В уже упомянутом марте 2023 года она составила 1%. Для сравнения: в 2022 году доля рынка за аналогичный период составила 0,3%. Рост популярности электромобилей вызван развитием инфраструктуры, появлением новых брендов и законодательной поддержкой.

Целью работы является реализация системы для формирования поставок автозапчастей для электромобилей, включающую в себя веб-сервис, веб-приложение, десктопное приложение и выделенный сервис для переговоров с поставщиком.

Система предназначена для заказчиков, и совету директоров. В сайте предусмотрен ограниченный доступ к автозапчастям. Для получения доступа заказчику необходимо создать заявку на получение автозапчастей к выбранному поставщику. Система предоставляет автоматизированный способ создания, учета и ведения заявок. Также она позволяет совету директоров завершать или отклонять заявки, возможность редактировать существующие и добавлять новые автозапчасти.

Нефункциональные требования к разрабатываемой системе:

1. Должна поддерживаться кроссплатформенность.
2. Интерфейс системы и текст ошибок должны быть русифицируемы.

В ходе работы необходимо выполнить следующие задачи:

* 1. Разработать дизайн приложения.
  2. Создать базу данных в PostgreSQL.
  3. Создать веб-сервис на языке GoLang 1.20.
  4. Реализовать интерфейс гостя на технологии React.
  5. Развернуть веб-приложение React на Github Pages.
  6. Добавить авторизацию и аутентификацию в веб-сервис.
  7. Реализовать интерфейс заказчика в React.
  8. Реализовать интерфейс совета директоров React.
  9. Создать мобильное приложение на React-Native.
  10. Создать асинхронный сервис для рассмотрения тендера на Python.
  11. Подготовить набор документации, включающий РПЗ, ТЗ и набор диаграмм.

# **БИЗНЕС-ПРОЦЕСС**

Широкий ассортимент электромобилей в Китае превосходит все остальные страны. Это приводит к огромной конкуренции среди брендов внутри страны. Автопроизводители вынуждены придумывать способы, как выделяться из толпы. В ход идут самые современные инновации и самые необычные дизайны. Времена, когда китайские компании слепо копировали европейские и японские автомобили давно прошли.[[1](https://quto.ru/journal/articles/prichiny-po-kotorym-kitai-lider-po-proizvodstvu-elektromobilei.htm)]

В 2023 году в ЕС автоимпорт из Китая удвоился, и сейчас электрокары из Поднебесной занимают 8% рынка. По прогнозам Еврокомиссии, за пару лет удвоится и эта доля. Чтобы это предотвратить, в Европе планируют ввести дополнительные пошлины на ввоз собранных в Китае электромобилей.

В России же, по состоянию на 2023 год, китайский автоэкспорт вырос в восемь раз. Если в 2022 на российских дорогах лишь каждая пятая машина была китайской, то теперь — каждая вторая. Что касается электромобилей, то в прошлом году россияне купили 14 869 новых авто на электротяге, что на 290% больше в сравнении с 2022. Пока с таким положением дел никто не спорит. [[2](https://vc.ru/volts/994730-kitayskie-elektromobili-vytesnyayut-evropeyskie-i-zavoevyvayut-pervenstvo-na-rynke)]

В связи с таким огромным спросом был создан сервис, чтобы вручную не искать автозапчасти электромобилей, заказчики могут оставлять заявки на формирования поставщиков и выбирать наиболее подходящие автозапчасти. Каждая заявка вначале создается как черновик, в который можно добавлять и из которого можно удалять автозапчасти. В одной заявке можно указать сразу несколько автозапчастей.

Каждой автозапчасти можно выставить нужное количество. Когда заказчик определится с автозапчастями, которые будут участвовать в заявке, он формирует заявку и она отправляется на модерирование. После этого её уже нельзя редактировать. Можно также посмотреть историю своих заявок. Для того, чтобы автозапчасти, их количество совпадали c ожиданиями - совет директоров отклоняет и принимает заявки таким образом, чтобы все происходило корректно. На случай, если заказчик передумает формировать заявку, присутствует возможность ее отменить.

За рассмотрение заявки отвечает выделенный сервис, который проверяет информацию о поставщиках и её возможности поставлять заданные автозапчасти. Функции пользователей с разными ролями описаны на диаграммах прецедентов (рис. 1).

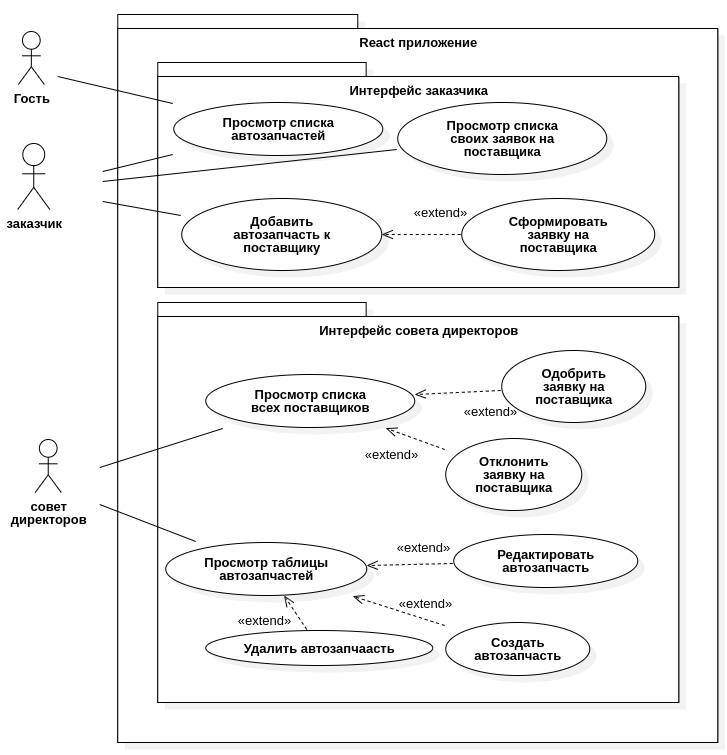


Рисунок 1 - Диаграмма прецедентов

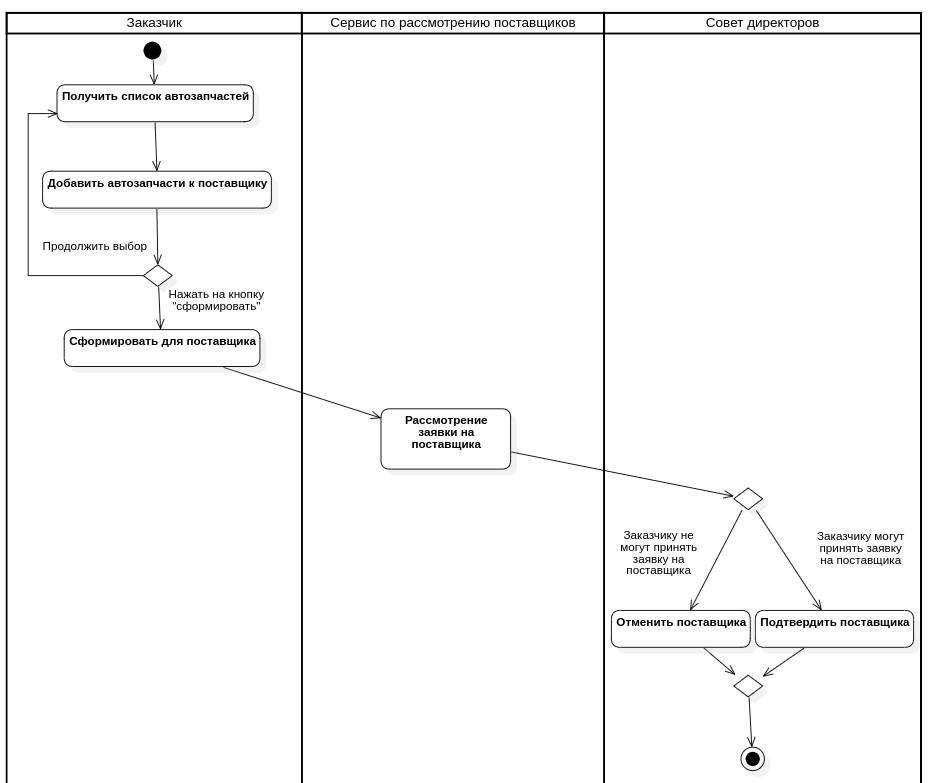
Гостям доступен просмотр автозапчастей. Зарегистрированные гости – заказчики. Они могут добавлять автозапчасти в заявку, просматривать список своих заявок и сформировывать текущую заявку. Заявки обрабатываются советом директоров. В результате обработки заявки её либо одобряют, либо отклоняют. Совету директоров также доступны уникальные функции для работы с автозапчастями, а именно: просмотр всех автозапчастей, редактирование, создание и удаление автозапчастей, а также просмотр списка всех поставщиков в табличном виде. Процесс оформления заявки отражен на диаграмме деятельности (рис. 2).

Рисунок 2 - Диаграмма деятельности

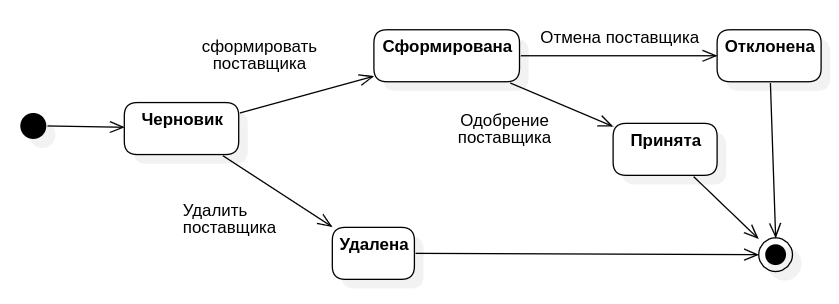
Заказчик выбирает автозапчасти, затем формирует на основе выбранных автозапчастей заявку. Эту заявку обрабатывает совет директоров. Возможные состояния заявки отражены на диаграмме состояний (рис. 3).

Рисунок 3 - Диаграмма состояний уведомления

При выборе первой автозапчасти формируется черновик. Последующие выбранные автозапчасти добавляются в этот черновик. Заказчик затем формирует заявку, удаляет её или выходит из приложения. Сформированную заявку обрабатывает совет директоров. Они могут одобрить или отклонить её.

# **АРХИТЕКТУРА**

Архитектура системы отображена на диаграмме развертывания (рис. 4). Бэкенд и база данных разворачивается на самом устройстве, а s3 хранилище изображений, Redis в докер контейнерах [6]. Такое решение было принято в связи с тем, что технология докер контейнеров позволяет быстро и удобно разворачивать целые системы на любых устройствах, поддерживающих докер. Еще одним неоспоримым преимуществом докер контейнеров является то, что их контейнеры представляют собой виртуальные машины, а значит их работа не зависит от платформы устройствах. Сервис рассмотрение поставщика [3] разворачиваются отдельно и необязательно на том же компьютере, что и основной сервис.

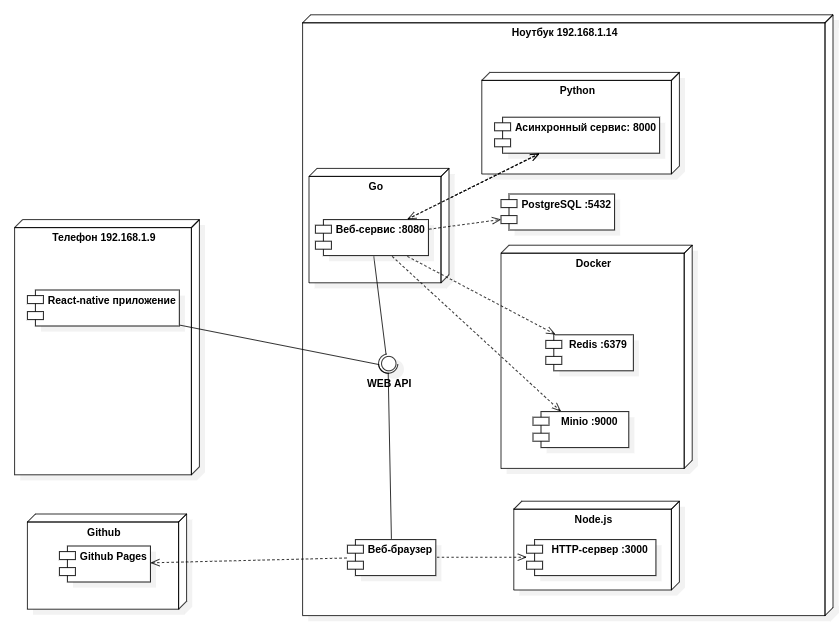


Рис. 4 - Диаграмма развертывания

Мобильное [10] и браузерное приложения [7, 8, 9] обращаются к веб-сервису на базе языке Golang 1.20 [3, 4, 5] через REST Web-API. Использование Golang обосновано его проверенной эффективностью, признанием в индустрии и современностью языка.

Данные хранятся в СУБД PostgreSQL [11], их структура отражена на ER диаграмме (рис. 5). СУБД PostgreSQL является одним из стандартов индустрии, поэтому было решено использовать её. Структура данных довольна проста. Помимо базовых полей, тендера также имеет поле status\_check, отражающее переговоры с поставщиком, полученные от асинхронного сервиса. Для хранения в одной заявке нескольких автозапчастей используется промежуточная таблица Assembly\_Autopart, реализующая связь М-М. Устройство бекенда приложения изображено на диаграмме классов бекенда.

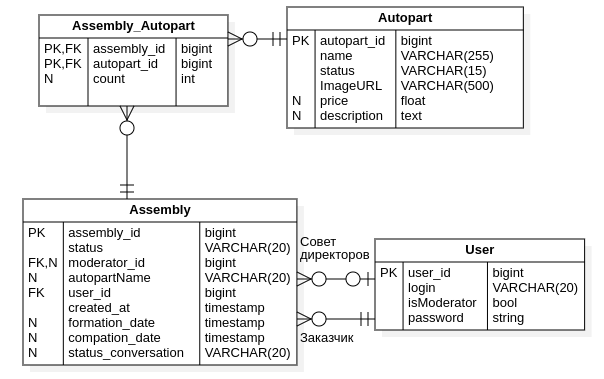


Рисунок 5 - ER диаграмма

Устройство бекенда приложения изображено на диаграмме классов бекенда (рис 6.). Модели имеют связи с таблицами в базе данных. Также некоторые модели имеют связи с внешними сервисами. В частности,автозапчасти имеют связь с сервером статических файлов, т.к. в данных получателя хранится ссылка на их изображение, хранимое на сервере статических файлов.

Изображение выглядит как текст, диаграмма, линия, План

Автоматически созданное описание

Рисунок *6* *-* Диаграмма классов бекенда

Связь фронтенда и бекенда отражена на диаграмме классов фронтенда (рис. 7). Ключевые страницы имеют связь с API аутентификации, т.к. доступ к ним осуществляется только для авторизированных пользователей с определенными правами (ролями).

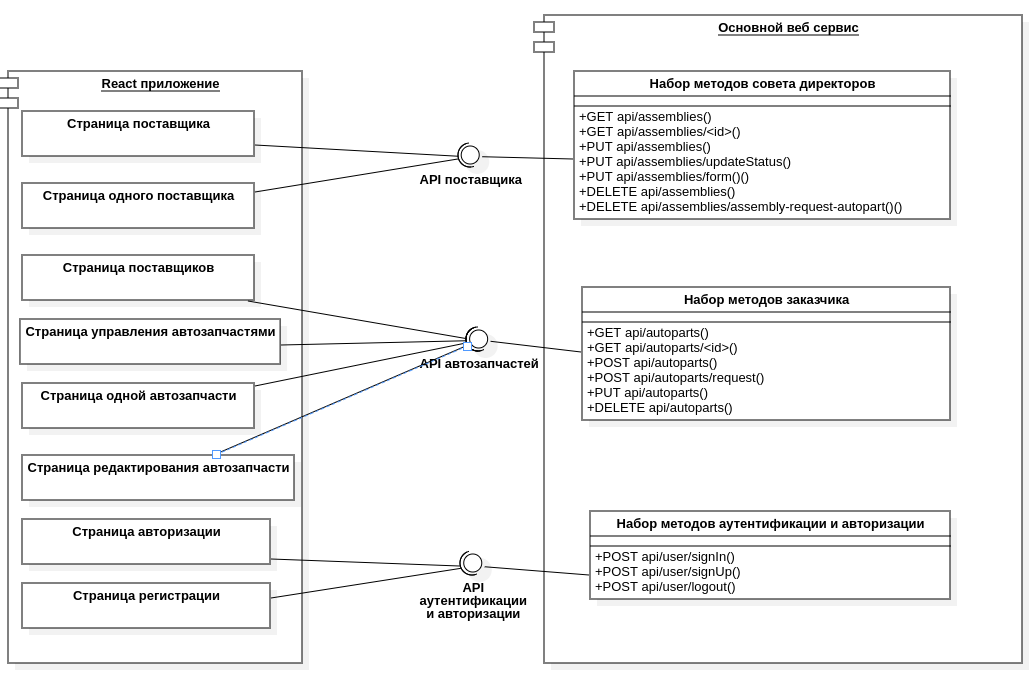


Рисунок 7 *-* Диаграмма классов фронтенда

# **АЛГОРИТМЫ**

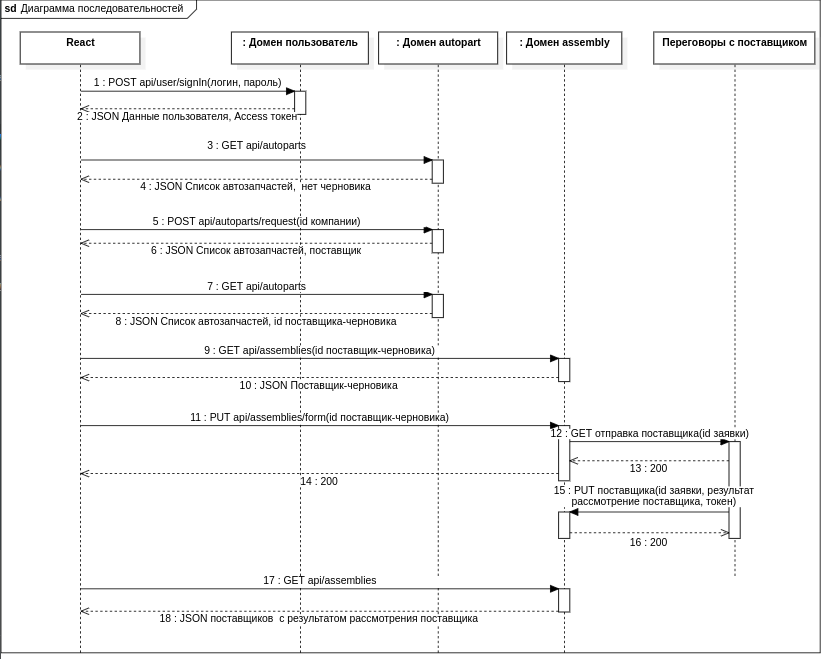
Алгоритм работы системы отображен на диаграмме последовательности (рис. 8). В основе системы лежит веб-сервис, реализующий внутри себя всю бизнес-логику. Он предоставляет доступ к методам из следующих доменов: поставщик, автозапчасти, и аутентификация. Методы следуют правилам REST API.

Рисунок 8 - Диаграмма последовательности

В начале бизнес-процесса происходит аутентификация заказчика. Для этого он отправляет через графический интерфейс запрос, передавая в нем логин и пароль. Если аккаунт с такими данными существует, то заказчик получает JWT в ответном запросе. Если же такого аккаунта не существует, или пароль введен неверно, заказчик получит ошибку. В таком случае ему надо либо пройти регистрацию, либо ввести пароль верно. Затем графический интерфейс заказчика запрашивает у веб-сервиса список поставщиков, которые возвращаются в JSON формате. Заказчик выбирает автозапчасть, которую хочет добавить в поставщика, и, нажимая на кнопку «плюс» в графическом интерфейсе, отправляет запрос на добавление автозапчасти в свою черновую заявку. Этот процесс может продолжаться несколько раз.

Когда закзачик определится с выбором, он нажимает на кнопку «сформировать» в графическом интерфейсе. После этого приложение запрашивает id черновой заявки поставщика и затем отправляет запрос на формирование этой заявки. В этот момент основной веб-сервис выполняет асинхронный запрос к другому сервису, чтобы он рассмотрел заявку на переговоры с поставщиком и вернул статус с решением в основной сервис. Через некоторое количество времени уведомитель может увидеть, что статус его переговоров с поставщиком изменился.

Процесс рассмотрения поставщика происходит также через графический интерфейс. Совет директоров могут просматривать списки всех поставщиков и, нажимая на соответствующие кнопки, отправлять запросы на подтверждение или отклонение поставщика в основной веб-сервис. В эти запросы также можно включить фильтры по имени создателя поставщика, одному из статусов поставщика и диапазону дат, в которых эти поставщики были сформированы. Также совету директоров могут управлять поставщиками через графический интерфейс. Им доступны такие функции, как создание и редактирование поставщиков, просмотр списка поставщиков в виде таблицы и удаление их. Для каждой из этих функций присутствует свой метод, отправляемый на основной веб-сервис.

# **ОПИСАНИЕ ИНТЕРФЕЙСА**

Главное меню приложения включает пункты, которые доступны в зависимости от роли заказчика (рис. 9, 10).

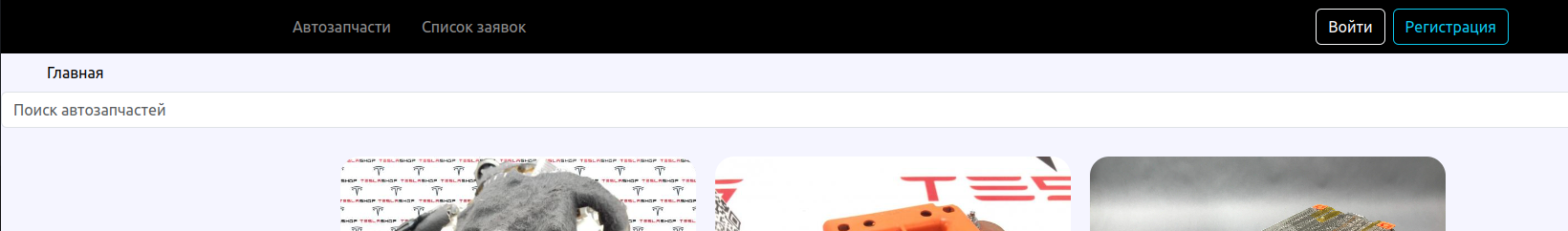


Рисунок 9 - Главное меню (неавторизированный пользователь)

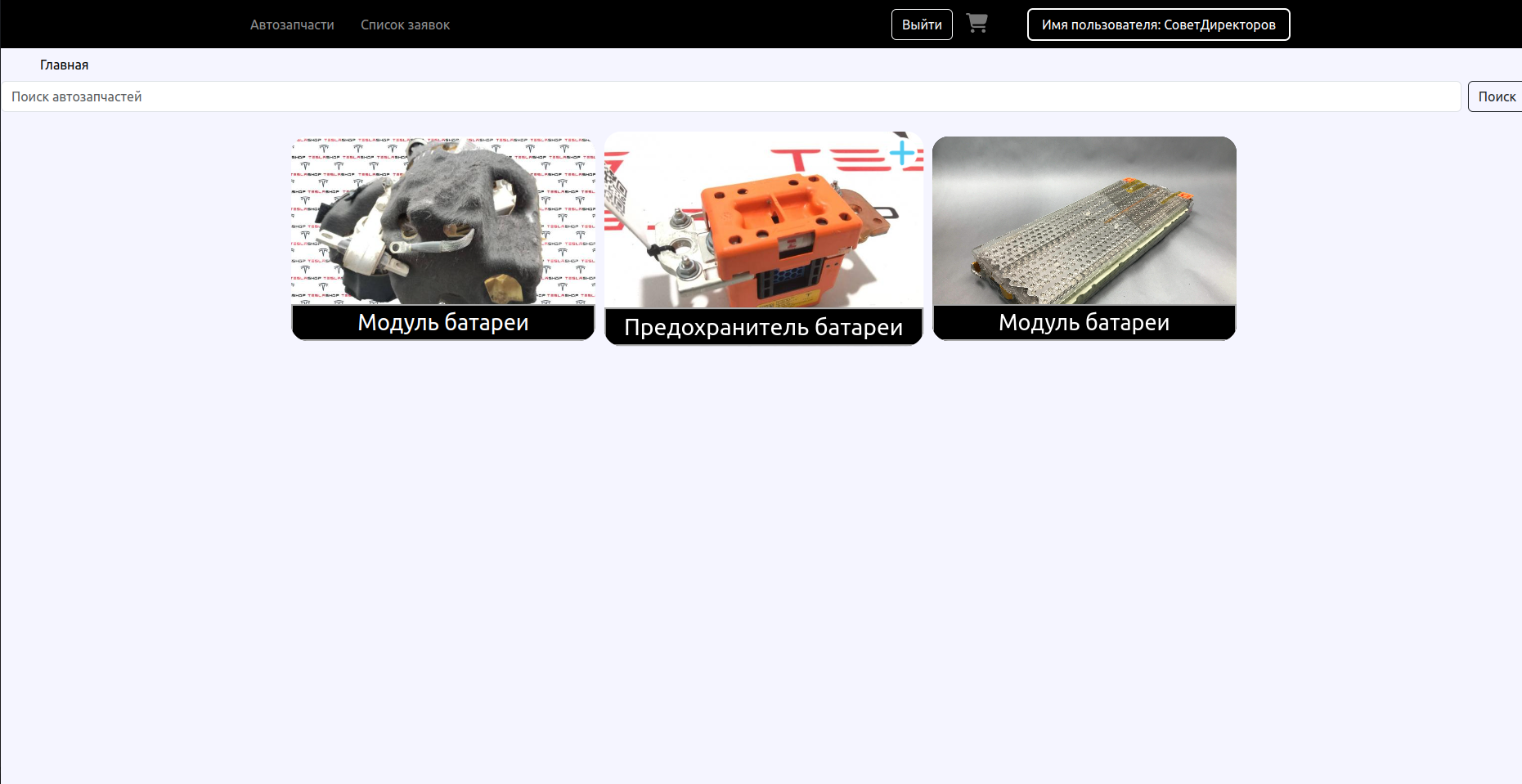


Рисунок 10 - Главное меню (совет директоров)

Первоначальная страница для всех пользователей и гостей. В зависимости от типа пользователя её содержимое меняется. Гости, например, могут посмотреть список автозапчастей и подробную информацию о них, а также нажать на кнопку «Войти», либо «Регистрация». А у заказчика уже есть кнопки, «Корзина» и «Выход».

На странице с формой авторизации (рис. 11) отображается форма, через которую гость входит в свой аккаунт. При успешном вводе логина и пароля гость получает JWT, который сохраняется в cookie и используется при отправлении запросов.

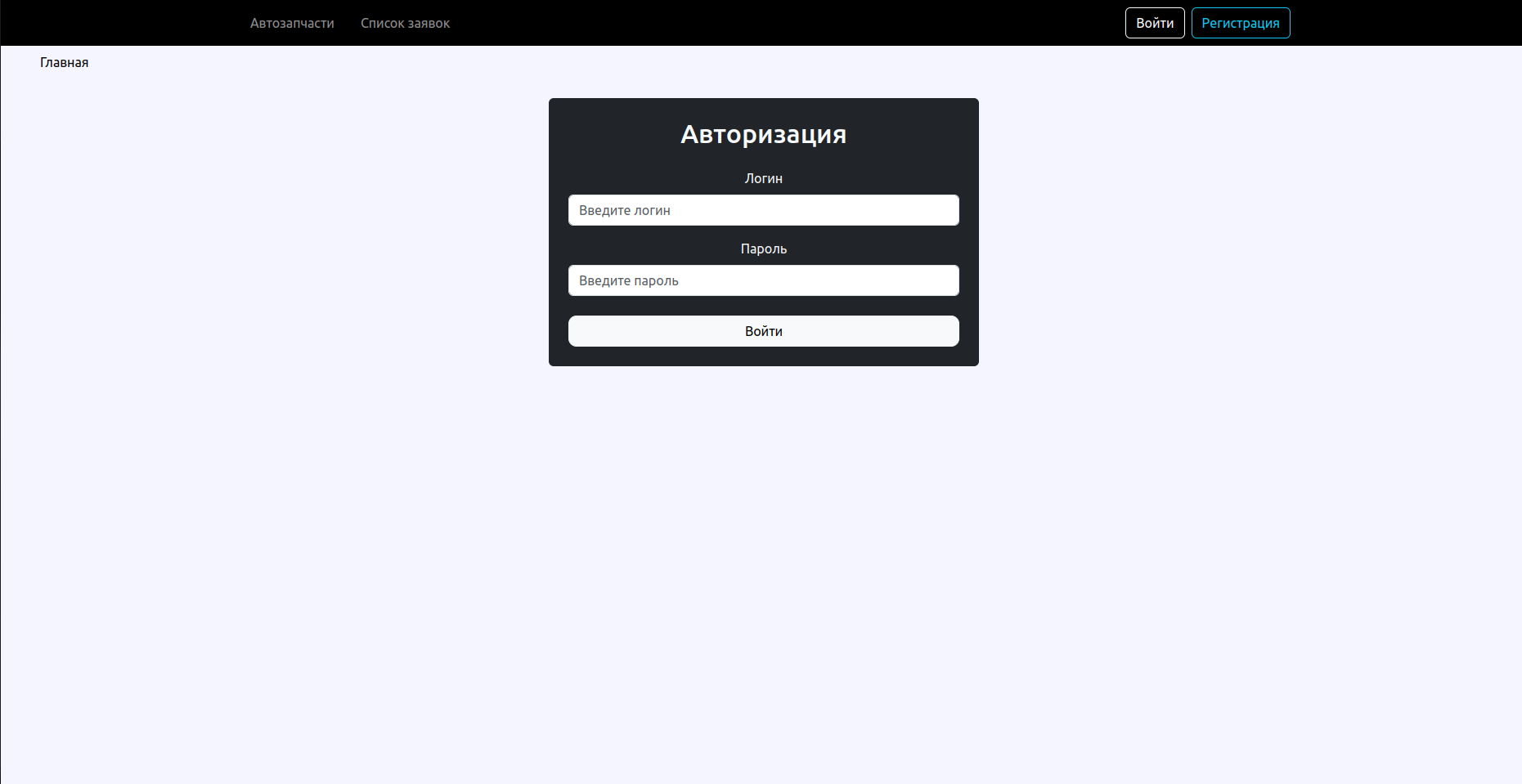


Рисунок 11 - Страница авторизации

На странице с формой регистрации (рис. 12) гости могут завести аккаунт. Для этого нужно указать логин, и пароль. Если введенный логин уже занят система попросит пользователя сменить его.

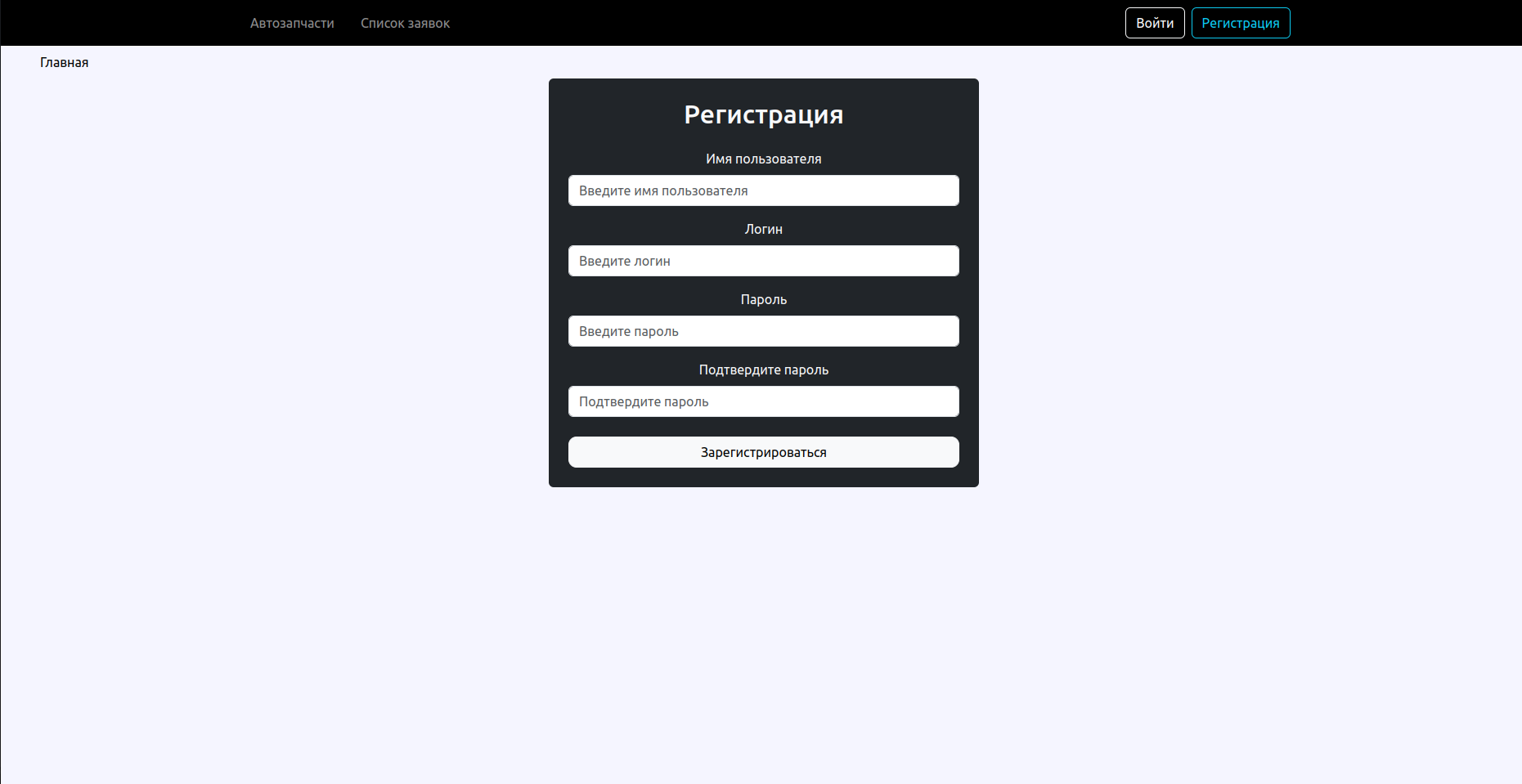


Рисунок 12 - Страница регистрации

На странице со списком автозапчастей (рис. 13) отображается список автозапчастей в виде карточек. У каждой автозапчасти есть хендлер нажатия для перехода в режим «Подробнее», переносящая заказчика на страницу с подробной информацией о автозапчасти.

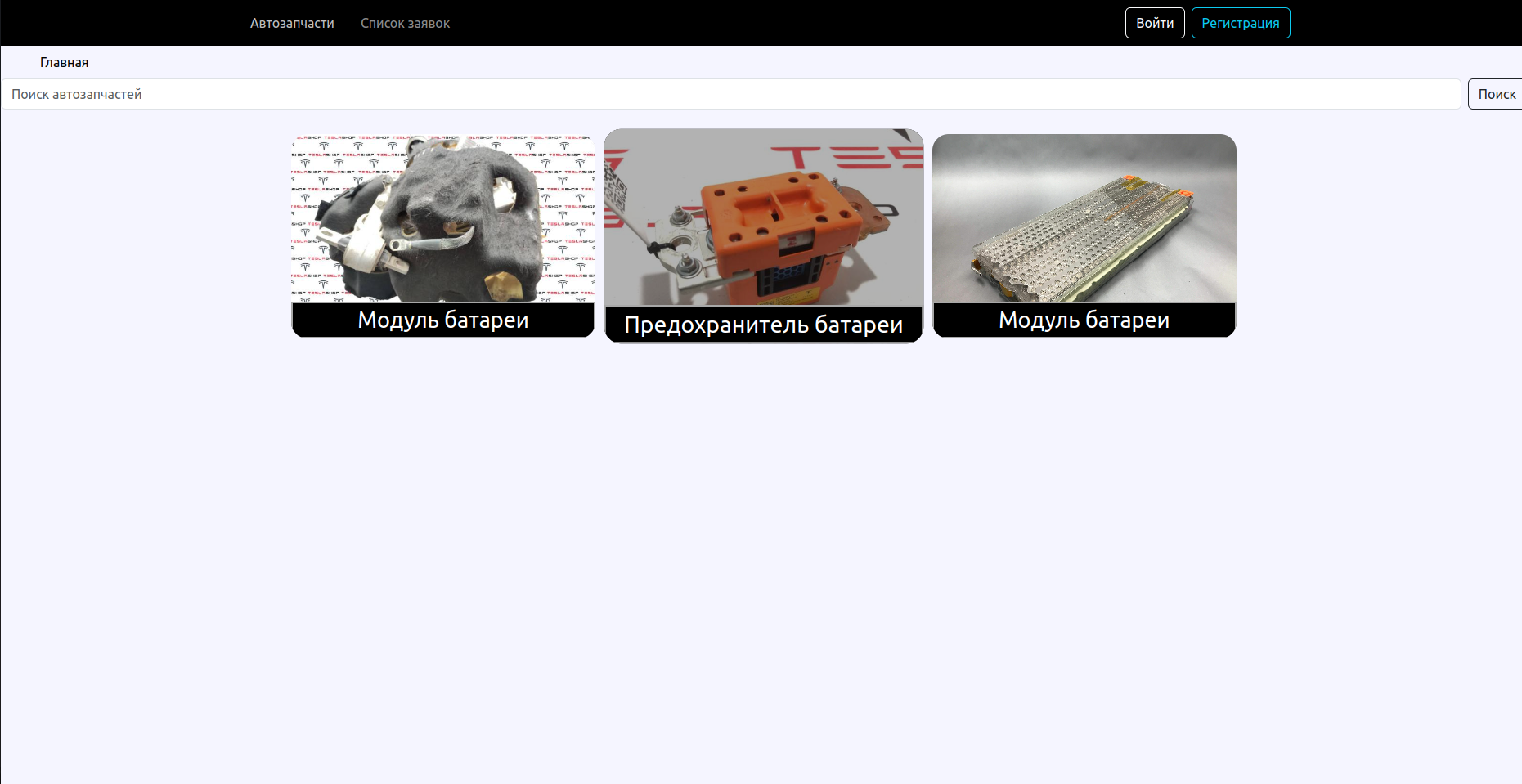
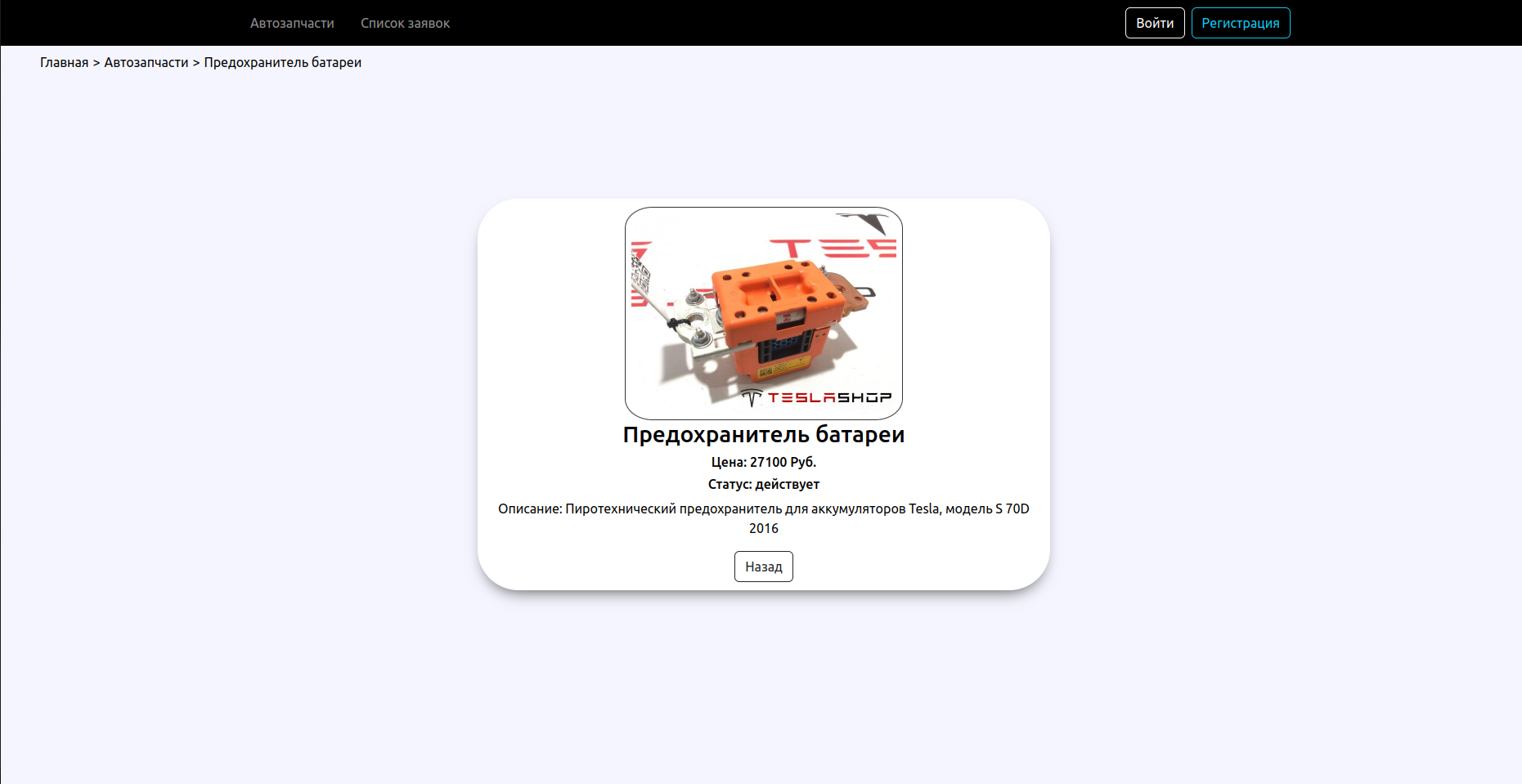
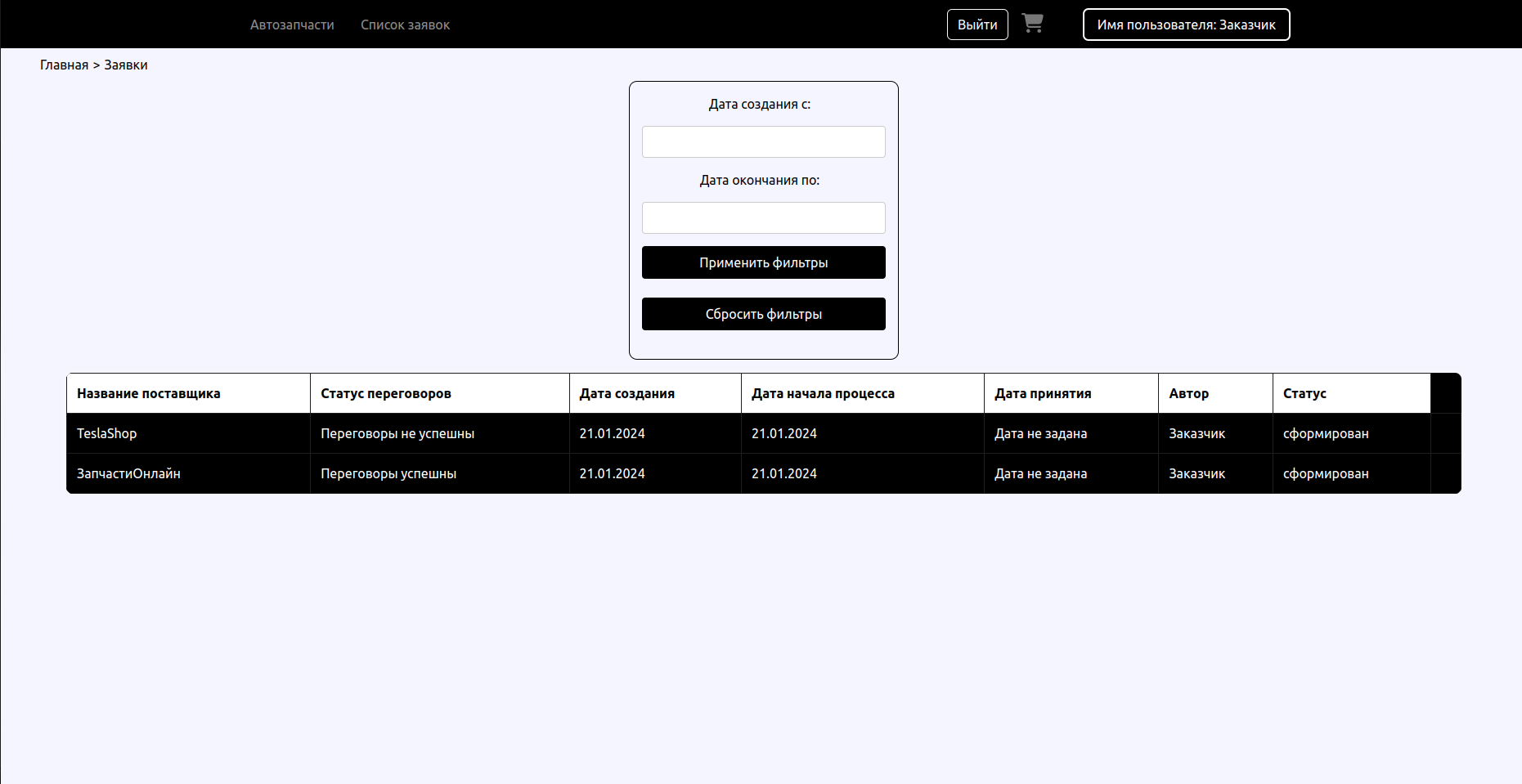


Рисунок 13 - Страница со списком автозапчастей

На странице с подробным описанием автозапчасти (рис. 14) отображается подробная информация о автозапчасти: Название, Цена, статус, описание автозапчасти.

Рисунок 14 - Страница с подробным описанием автозапчасти

На странице со списком уведомлений (рис. 15, 16) отображается список поставщиков. В зависимости от типа пользователя этот список будет функционально отличаться. Так, для заказчика отображается список созданных ими заявок: название поставщика, переговоры с поставщиком, дата создания, дата начала процесса, дата принятия, автор, статус. Также можно отфильтровать поставщиков по дате формирования поставщика, выбрав либо одну дату, либо промежуток.

Рисунок 15 - Страница со списком поставщиков (заказчик)

Для совета директоров функция этой страницы расширена. Для них отображается список всех заявок всех заказчиков с более подробной информацией: название поставщика, результат переговоров с поставщиком, дата создания, дата начала процесса, дата принятия, автор, совет директоров, статус, и две кнопки «Завершить» и «Отказать» для подтверждения или отмены поставщика. Также присутствует фильтр по создателю поставщика.

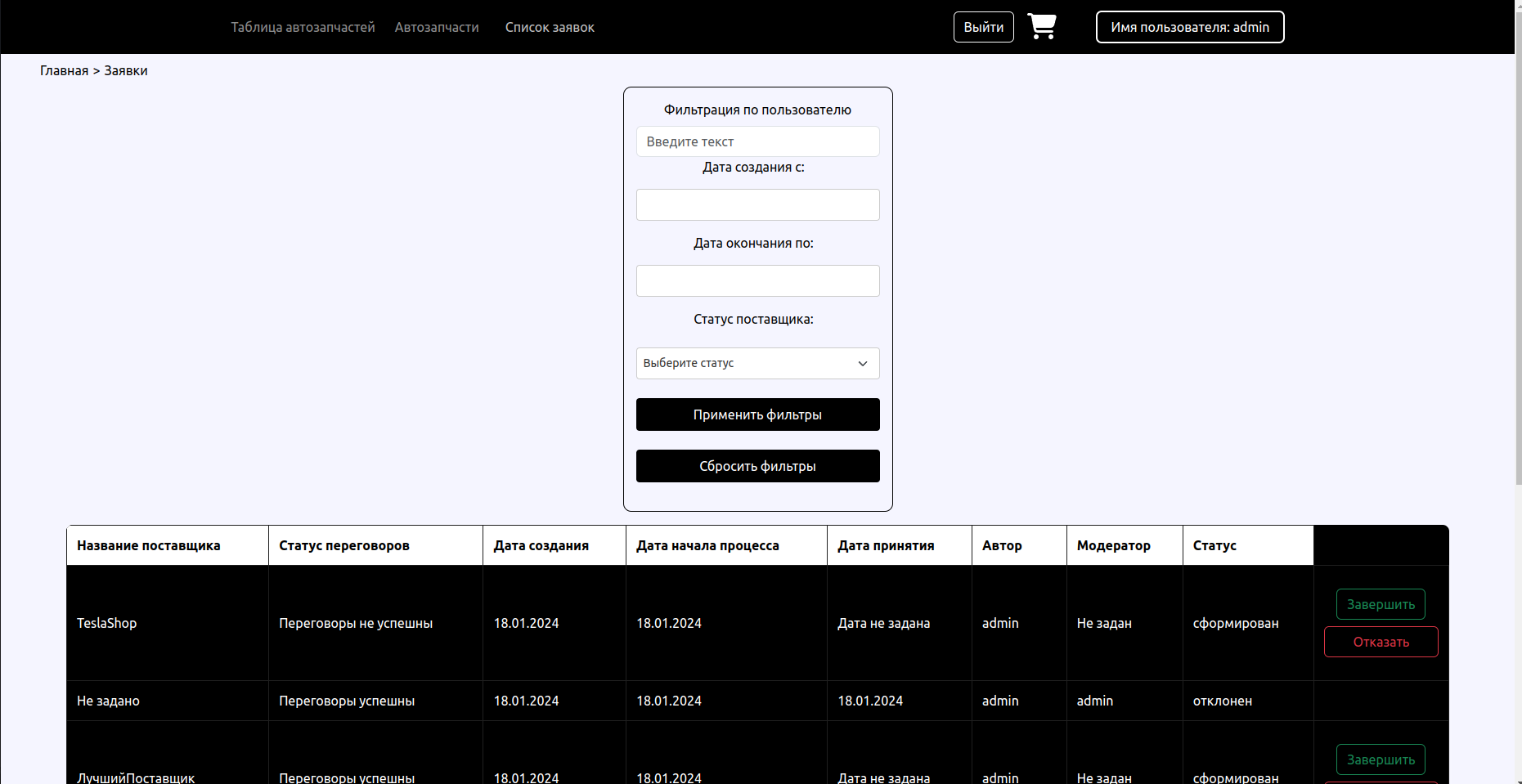


Рисунок 16 - Страница со списком поставщиков (совет директоров)

На странице с подробным описанием транзакции (рис. 17) отображается подробная информация об транзакции. Список выбранных автозапчастей в виде таблицы, а также все поля транзакции.

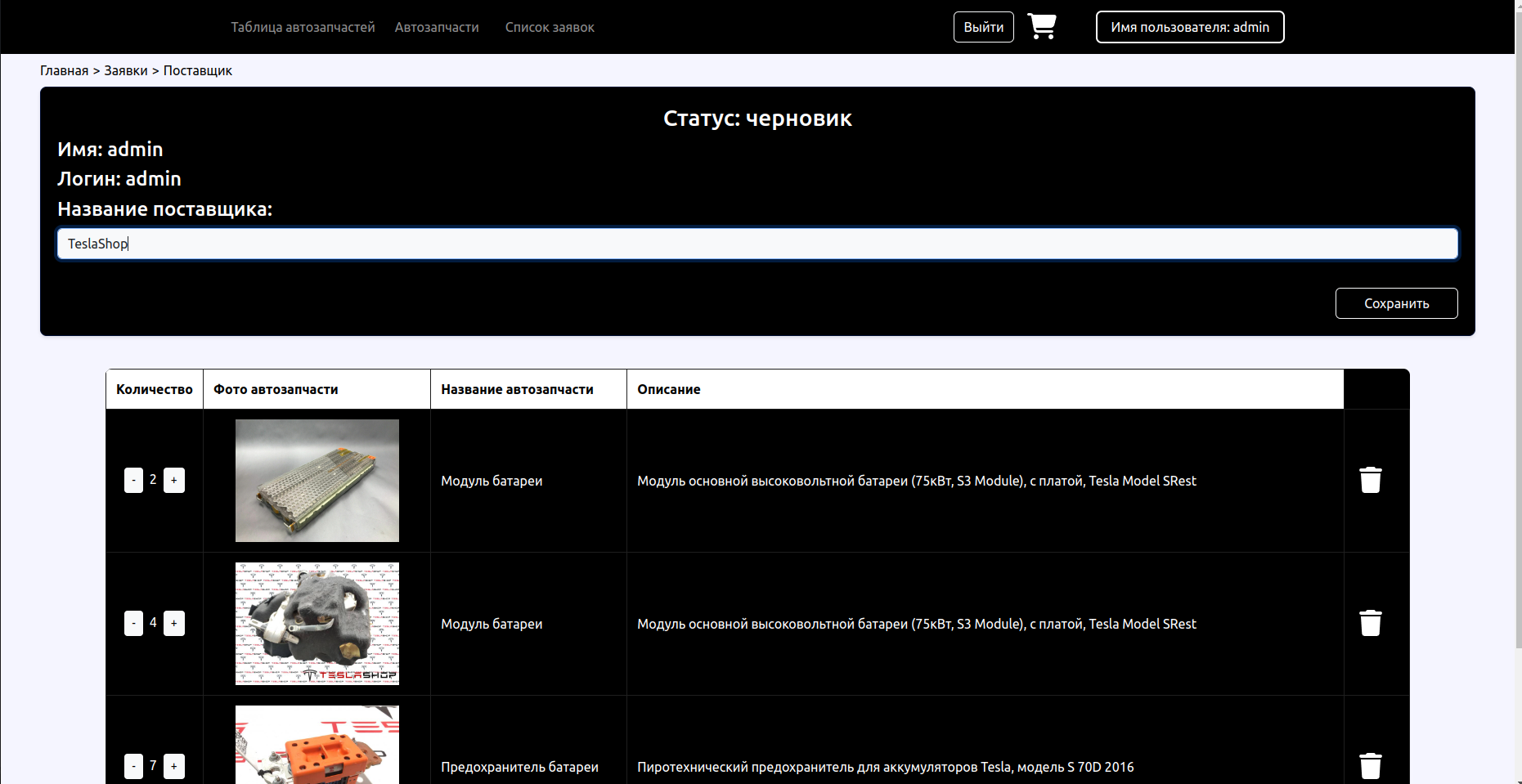


Рисунок 17 - Страница с подробным описанием поставщика

На странице с таблицей автозапчасти (рис. 18) совет директоров может в удобном формате просмотреть список всех автозапчастей, существующих в системе. Отображаются следующие поля: логотип автозапчасти, название автозапчасти, статус, описание, кнопки с удалением и возможностью редактировать. Также можно перейти на страницу создания новой автозапчасти (рис. 19).

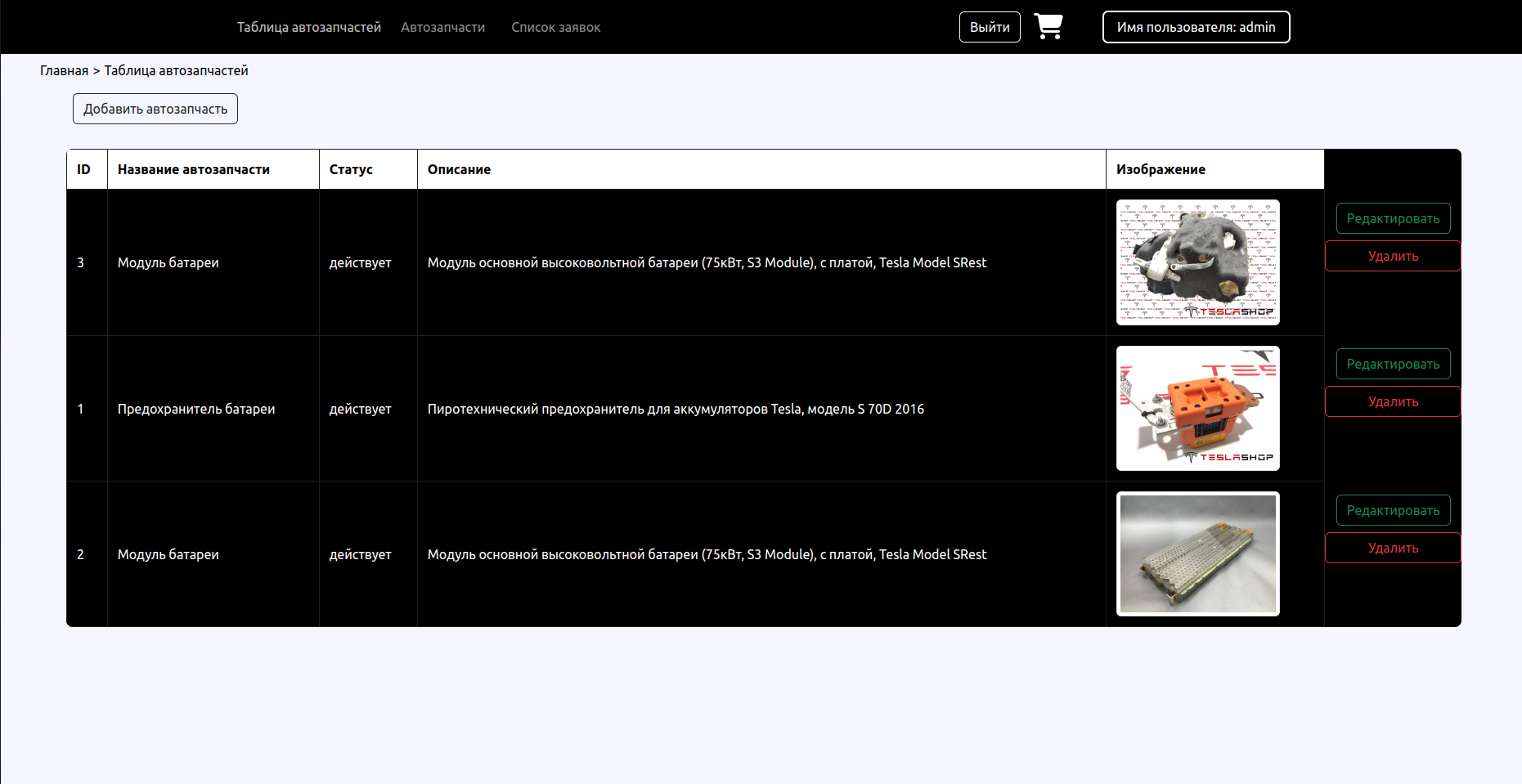


Рисунок 18 - Страница с таблицей автозапчасти

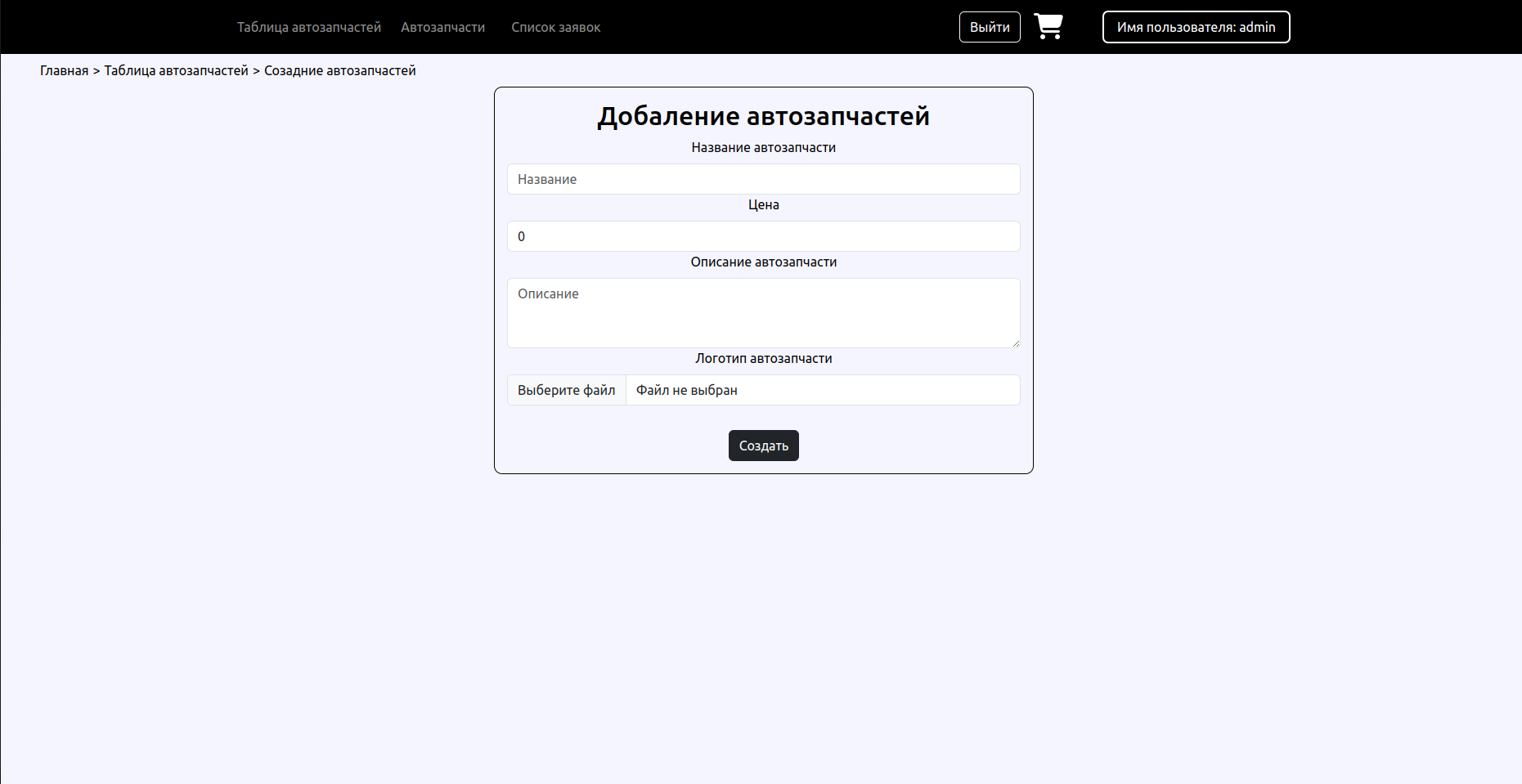


Рисунок 19 - Страница создания автозапчасти

# **ЗАКЛЮЧЕНИЕ**

В ходе работы были достигнуты следующие результаты:

1. Был разработан дизайн приложения с помощью набора стилей CSS и HTML тегов.
2. База данных была создана и расположена в docker контейнере.
3. Был создан веб-сервис на Golang, с использованием веб-фреймворка Gin.
4. Разработан интерфейс гостя с использованием технологии React Framework и подключен к веб-сервису.
5. Приложение интерфейса было развернуто на сервисе Github Pages по ссылке https://scremyda.github.io/DevelopmentNetworkApplicationFrontend.
6. В веб-сервис добавлена авторизация через JWT, а методы задокументированы через Swagger.
7. Реализован интерфейс заказчика. Доступ к нему имеют только авторизированные пользователи.
8. Выделенный сервис был разработан и развернут в отдельном виртуальном окружении Python.
9. Реализован интерфейс совета директоров для подтверждения новых автозапчастей и редактирования/добавления получателей.
10. Было реализовано мобильное приложение на React Native, повторяющее интерфейс веб-приложения на React.
11. Подготовлен набор документации, включающий РПЗ, ТЗ и набор UML диаграмм.
12. Исходный код проекта доступен в GitHub https://github.com/scremyda/DevelopmentNetworkApplication

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

* 1. Причины, по которым Китай — лидер по производству электромобилей [Электронный ресурс]. URL: https://quto.ru/journal/articles/prichiny-po-kotorym-kitai-lider-po-proizvodstvu-elektromobilei.htm (дата обращения 04.01.2024).
  2. Китайские электромобили вытесняют европейские и завоевывают первенство на рынке? [Электронный ресурс]. URL: https://vc.ru/volts/994730-kitayskie-elektromobili-vytesnyayut-evropeyskie-i-zavoevyvayut-pervenstvo-na-rynke (дата обращения 04.01.2024).
  3. Руководство по Go [Электронный ресурс] // GoLang. URL: https://metanit.com/go/tutorial/ (дата обращения: 11.09.2023)
  4. Руководство по Gin Web Framework [Электронный ресурс] // GoLang. URL: https://github.com/gin-gonic/gin (дата обращения: 12.09.2023).
  5. Руководство по Gorm [Электронный ресурс] // Gorm. URL: https://gorm.io/docs/index.html (дата обращения: 14.10.2023)
  6. Полное практическое руководство по Docker [Электронный ресурс] // Habr. URL: https://habr.com/ru/articles/310460/ (дата обращения: 10.10.2023).
  7. Руководство по React [Электронный ресурс] // Metanit. URL: https://metanit.com/web/react/ (дата обращения: 11.11.2023).
  8. Quick Start – React [Электронный ресурс] // React. URL: https://react.dev/learn (дата обращения: 11.11.2023).
  9. Руководство по React Bootstrap [Электронный ресурс] // React Bootstrap. URL: <https://react-bootstrap.netlify.app/docs/getting-started/introduction>
  10. Руководство по React Native [Электронный ресурс] // React Nativ. URL: <https://reactnative.dev/docs/getting-started>
  11. Руководство по PostgreSQL [Электронный ресурс] // Metanit. URL: https://metanit.com/sql/postgresql/ (дата обращения: 05.10.2023).

**Приложение. Техническое задание**

| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |
| --- | --- |

**Факультет «Информатика и системы управления»**

**Кафедра ИУ5 «Системы обработки информации и управления»**

Дисциплина «Разработка интернет-приложений»

Техническое задание

Тема: «Поставки комплектующий для производства электроавтомобилей»

Студент: Белозеров Д. С.

Группа ИУ5-54Б

Преподаватель: Канев А.И.

Москва, 2024

# **Цель**

Реализовать систему для заявок выбора поставщика, включающую в себя веб-сервис, веб-приложение, десктопное приложение и выделенный сервис для рассмотрения результата для тендера.

# **Назначение**

Система предназначена для заказчиков, совета директоров. В сервисе предусмотрен ограниченный доступ к поставщикам. Для получения доступа заказчик необходимо создать заявку на получение доступа к выбранному поставщику. Система предоставляет автоматизированный способ создания, учета и ведения заявок. Также она позволяет совету директоров принимать или отклонять заявки. Совет директоров имеют возможность редактировать существующие и создавать новые автозапчасти.

# **Задачи:**

* 1. Разработать дизайн приложения.
  2. Создать базу данных в PostgreSQL.
  3. Создать веб-сервис на языке Golang 1.20.
  4. Реализовать интерфейс гостя на технологии React.
  5. Развернуть веб-приложение React на Github Pages.
  6. Добавить авторизацию и аутентификацию в веб-сервис.
  7. Реализовать интерфейс заказчика в React.
  8. Реализовать интерфейс совета директоров в React.
  9. Создать мобильное приложение на React Native.
  10. Создать асинхронный сервис для отслеживания отправки уведомления на Python.
  11. Подготовить набор документации, включающий РПЗ, ТЗ и набор диаграмм.

# **Методы веб-сервиса**

| № | Метод | URL | Описание | Входные данные | Выходные данные |
| --- | --- | --- | --- | --- | --- |
| Методы авторизации и аутентификации | | | | | |
| 4.1.1 | POST | api/users/sign\_up | Регистрация | login: string  password: string  user\_name: string |  |
| 4.1.2 | POST | api/users/login | Вход в аккаунт | login: string  password: string | {  access\_token: string,  expires\_type: string  } |
| 4.1.3 | GET | api/logout/ | Выход из аккаунта  Доступно только авторизированным пользователям | auth cookies | status\_code: int |
| Методы автозапчастей | | | | | |
| 4.2.1 | GET | api/autoparts/upload-image | Обновляет фотографию автозапчасти. Доступно только совету директоров | auth header  file: File  autopart\_id: Int | {  image\_url: string,  status: string  } |
| 4.2.2 | GET | api/ autoparts | Возвращает список автозапчастей, удовлетворяющих переданным критериям | auth header  {  autopart: string,  search: string  } | {  comapanies: {  " autopart\_id": int,  name: string,  status\_:string,  description: string  image\_url: string  }[]  } |
| 4.2.3 | PUT | api/autoparts /request | Добавляет в поставщика пользовате-ля автозапчасть{ autopart\_id}, возращает id поставщика, в которую была добавлена автозапчасть  Доступно только авторизированным пользова-телям | auth header  {  autopart\_id: int,  draft\_id: int  } | {  id: int,  status: string  } |
| 4.2.4 | GET | api/autoparts /{id} | Возвращает автозапчасть | id: int | {  "id": int,  name: string,  status: string,  description: string  image\_url: string  } |
| 4.2.5 | POST | api/autoparts | Создает автозапчасти  Доступно только совету директоров | auth header  {  name: string,  description: string,  inn  image\_url: file?  } | {  autopart\_id: string,  status: string  } |
| 4.2.6 | PUT | api/ autoparts | Обновляет информацию о автозапчасти  Доступно только совету директоров | auth header  {  name: string?  description: string?  id: int?  status: string?  price: float  } | {  name: string,  description: string,  id: int,  status: string,  price: float  } |
| 4.2.7 | DELETE | api/ autoparts | Помечаетавтозапчасть как удаленную  Доступно только совету директоров | auth header  id: int | {  deleted\_id: int  } |
| Методы поставщика | | | | | |
| 4.3.1 | GET | api/assemblies | Возвращает список заявок пользователя  Доступно только авторизированным пользователям | auth header | tenders: {  creation\_date: string,  completion\_date: string,  formation\_date: string,  description: string,  autoparts\_ assemblies: {  {  autopart: {  name: string,  description: string,  id: 0,  image\_url: string,  status: string  },  autopart\_id: int,  assembly\_id: int,  id: int,  count: int  }[]  }[] |
| 4.3.2 | GET | api/assemblies /{id} | Возвращает по-дробную ин-формацию о за-явке  Доступно только авторизированным пользова-телям | auth header  id: int | {  tenders: {  creation\_date: string,  completion\_date: string,  formation\_date: string,  autopart\_ assemblies: Autopart [],  }[],  status: string  } |
| 4.3.3 | GET | api/assemblies /current | Возвращает те-кущую черновую заявку пользова-теля  Доступно только авторизированным пользова-телям | auth header | tenders: {  creation\_date: string,  completion\_date: string,  formation\_date: string,  autopart\_assemblies: Autopart []  } |
| 4.3.4 | DELETE | api/assembly-request-company | Удаляет из поставщика пользователя автозапчасть {autopart\_id}  Доступно только авторизированным пользова-телям | auth header  {  id: int  } | {  deleted\_autopart \_ assembly: int  status": string"  } |
| 4.3.5 | PUT | api/assemblies /form | Устанавливает заявке статус «сформирована»  Только если её текущий статус «черновик»  Доступно только авторизированным пользователям | auth header  status: string? | Status\_code:int |
| 4.3.6 | DELETE | api/ assemblies | Устанавливает заявке статус «удалена»  Только если её текущий статус «черновик»  Доступно только авторизированным пользова-телям | auth header  start\_date: datetime?  end\_date: datetime?  Status: string | {  Assemblies:Assemblies[],  Status:string  } |
| 4.3.7 | GET | api/assemblies?status&start\_date&end\_date | Возвращает отфильтрованный по датам созда-ния и статусу поставщика.  Доступно только совету директоров | auth cookies  start\_date: datetime?  end\_date: datetime?  Status: string | {  Assemblies:Assemblies [],  Status:string  } |
| 4.3.8 | PUT | api/assemblies /updateStatus | Устанавливает заявке статус «отклонена» или «заверше-на»  Только если её текущий статус «сформирована»  Доступно только совету директоров | {  assembly\_id: int,  Status: string  } | Status code |
| .Методы внешнего сервиса board\_of\_Directors\_check | | | | | |
| 4.4.1 | POST | api/assemblies /user-form-start | Совет директоров принимает решение о результатах переговоров с поставщиком. | assemblн\_id: int |  |

1. **Функциональные требования:**
   1. Главное меню. Здесь находятся кнопки для регистрации и авторизации или информация о текущем пользователе, и меню со списком доступных пунктов.
      1. Доступна всем пользователям.
      2. Действия
         1. Войти – переход на страницу 5.2. Только для гостей.
         2. Зарегистрироваться – переход на страницу 5.3. Только для гостей.
         3. Открыть список автозапчастей – переход на страницу 5.4. Для всех пользователей.
         4. Открыть список выбранных автозапчастей – переход на страницу 5.6. Только для авторизированных пользователей.
         5. Список заявок – переход на страницу 5.6. Только для пользователей.
   2. Страница с формой авторизации. На этой странице находится форма авторизации.
      1. Доступна гостям
      2. Действия
         1. Войти – производит запрос (метод 4.1.2).
         2. Зарегистрироваться – перенаправляет на страницу 5.3.
   3. Страница с формой регистрации. На этой странице находится форма авторизации.
      1. Доступна гостям
      2. Действия
         1. Зарегистрироваться – производит запрос (метод 4.1.1.), в котором передаются введенные данные формы.
         2. Войти - перенаправляет пользователя на страницу 5.2.
   4. Страница со списком автозапчастей. На этой странице располагается список всех получателей и панель фильтрации.
      1. Доступна всем пользователям.
      2. Выводится информация о автозапчастях в виде карточек (метод 4.2.2.).
         1. Фото автозапчасти.
         2. Название автозапчасти.
      3. Действия
         1. Отфильтровать автозапчасть по названию (метод 4.2.2.).
         2. Добавить автозапчасть к поставщику(метод 4.2.3.). Только для авторизированных пользователей.
         3. Нажатие на карточку – перенаправляет на страницу 5.5. Только для авторизированных пользователей.
   5. Страница с подробным описанием автозапчасти
      1. Доступна всем пользователям.
      2. Выводится информация о автозапчасти.
         1. Фото автозапчасти.
         2. Название автозапчасти.
         3. Статус автозапчасти.
         4. Описание автозапчасти.
         5. Цена автозапчасти.
      3. Действия
         1. Удалить автозапчасть (метод 4.2.7.). Только для совета директоров.
         2. Назад – перенаправляет на предыдущую страницу.
   6. Страница со списком поставщиков
      1. Доступна только авторизированным пользователям.
      2. Выводится информация об поставщиках в табличном формате (метод 4.3.1).
         1. Название поставщика.
         2. Дата создания.
         3. Дата формирования.
         4. Дата окончания.
         5. Создатель
         6. Кем обработана. Только для совета директоров.
      3. Действия
         1. Открыть подробную информацию об поставщике – перенаправляет на страницу 5.7.
         2. Установить уведомлению статус «отклонено» или «завершено» (метод 4.3.8.). Только для совета директоров.
         3. Отфильтровать список поставщиков по диапазону дат формирования и/или статусу (метод 4.2.1) и/или создателю (только для совета директоров).
   7. Страница с подробным описанием поставщика. Заказчик может изменить набор автозапчастей. Совет директоров может просматривать поставщиков всех пользователей.
      1. Доступна только авторизированным пользователям.
      2. Выводится информация о заявке. Список автозапчастей, выбранных в заявке, в виде таблицы (метод 4.3.2.).
         1. Фото автозапчасти.
         2. Имя создателя поставщика.
         3. Название поставщика.
         4. Статус поставщика
         5. Дата формирования
      3. Действия
         1. Удалить автозапчасть из поставщика (метод 4.3.4.). Если поставщик находится в статусе черновика. Только для создателя поставщика.
         2. Сформировать поставщика (метод 4.3.5.). Если поставщик находится в статусе черновика. Только для создателя поставщика.
         3. Удалить поставщика (метод 4.3.6.). Если поставщик находится в статусе черновика. Только для создателя уведомления.
   8. Страница создания автозапчасти.
      1. Доступна только совету директоров.
         1. Название автозапчасти.
         2. Описание автозапчасти.
         3. Фото автозапчасти.
         4. Цена.
      2. Действия
         1. Создать автозапчасти (метод 4.2.5)
   9. Страница с таблицей автозапчасти. Предоставляет совету директоров удобный способ отображения всех автозапчастей в табличной форме с возможностью редактирования Доступна только совету директоров.
      1. Доступна только совету директоров.
         1. Название автозапчасти.
         2. Статус автозапчасти.
         3. Описание автозапчасти.
         4. Фото автозапчасти.
      2. Действия
         1. Изменение полей автозапчасти. Можно изменять все вышеперечисленные поля автозапчасти.
         2. Сохранить изменения (методы 4.2.5. и 4.2.6.).
         3. Удалить автозапчасть (метод 4.2.7)
         4. Добавить автозапчасть. Открывает страницу 5.8.
2. **Требования к программному обеспечению:**
   1. Серверная часть
      1. ОС: Linux 6.5 / Windows 10
      2. Docker 24.0.6
      3. Python 3.11.5
      4. Golang 1.21.1
      5. redis 6.2-alpine
      6. nginx 1.19.2-alpine
      7. minio RELEASE.2022-10-15T19-57-03Z
      8. postgres 16.1
   2. Клиентская часть
      1. ОС: Windows/MacOS/Linux
      2. Веб-браузер: Safari 11.1+/Chrome 40+/Opera 27+/Firefox 44+/Yandex 20+
3. **Требования к аппаратному обеспечению:**
   1. Серверная часть
      1. Процессор минимум 2-ядерный с частотой от 2 ГГц.
      2. Оперативная память от 4 Гб.
      3. Место на жестком диске от 2 Гб.
   2. Клиентская часть
      1. Процессор с частотой от 1ГГц.
      2. Оперативная память от 512 Мб.