Техническое задание на разработку веб-приложения «Автосалон с дилерским обслуживанием»

1. Введение
   1. Цель

Целью разработки данной системы является автоматизация процессов выбора и покупки автомобилей, а также записи на дилерское обслуживание. Приложение должно обеспечить удобный интерфейс для выбора автомобиля с дополнительными опциями, записи на обслуживание и получения информации о состоянии транспортного средства.

* 1. Актуальность и назначение проекта

Современные автосалоны сталкиваются с необходимостью предоставления клиентам простого и эффективного инструмента для выбора автомобилей и записи на обслуживание. Автоматизированная система позволит:

* упростить процесс покупки автомобиля, включая выбор комплектации и опций;
* повысить удобство записи на техническое обслуживание с учетом загруженности сервиса;
* улучшить взаимодействие клиентов с автосалоном за счет информативной статистики обслуживаний.
  1. Основные пользователи системы

Покупатели — выбирают автомобили, добавляют опции, оформляют заказы и записываются на обслуживание.

Администраторы автосалона — управляют базой автомобилей, статистикой обслуживания и расписанием.

Сервисные специалисты — получают информацию о записях и статусах боксов для обслуживания.

1. Цели и задачи проекта
   1. Основные цели

* автоматизация процессов выбора автомобилей и записи на обслуживание;
* предоставление клиентам доступа к информации об их транспортных средствах и сервисной статистике.
  1. Конкретные задачи
* разработка функционала для выбора автомобиля с опциями;
* внедрение системы записи на дилерское обслуживание с учетом занятости боксов;
* реализация раздела статистики обслуживаний для каждого автомобиля;
* внедрение авторизации и регистрации пользователей.

1. Требования к системе
   1. Функциональные требования

Функциональность для покупателей:

* регистрация и авторизация;
* просмотр каталога автомобилей с фильтрацией по характеристикам (марка, модель, комплектация, цена);
* выбор автомобиля с добавлением опций (например, цвет, тип обивки, доп. оборудование);
* запись на обслуживание с интерактивным выбором свободного времени и бокса;
* просмотр истории обслуживаний (даты, выполненные работы, стоимость).

Функциональность для администратора:

* управление каталогом автомобилей (добавление, редактирование, удаление моделей и опций);
* управление расписанием и статусами боксов;
* мониторинг записей на обслуживание и статистики использования сервиса.

Функциональность для сервисных специалистов:

* просмотр расписания и записи на обслуживание;
* отметка статуса выполнения работ.
  1. Нефункциональные требования
* производительность: отклик интерфейса не более 2 секунд;
* масштабируемость: поддержка до 5000 пользователей одновременно;
* безопасность: защита данных пользователей и информации об автомобилях через SSL.

1. Требования к пользовательскому интерфейсу
   1. Основные экраны

* главная страница: каталог автомобилей с основными характеристиками;
* страница автомобиля: детальное описание модели, комплектации, опций и цены;
* личный кабинет: информация о текущих заказах;
* панель администратора: управление каталогом автомобилей и статистикой записей.
  1. Навигация

Главное меню с разделами: «Каталог», «Мои записи», «Обслуживание», «Личный кабинет», «Контакты».

Интерактивные элементы для выбора свободного времени в расписании ТО.

* 1. Юзабилити
* Простой и понятный интерфейс.

1. Требования к технической реализации
   1. Языки и технологии

* backend: Laravel, JavaScript.
* frontend: Laravel.
* база данных: SQL.
  1. Архитектура системы
* хранение данных в реляционной базе данных.
  1. Интеграции
* системы платежей: Stripe или PayPal для оплаты дополнительных опций;
* уведомления: интеграция с сервисами email-рассылок (SendGrid, Mailgun).

1. Требования к безопасности
   1. Аутентификация и авторизация

* аутентификация через JWT или OAuth2.
  1. Шифрование данных
* хранение паролей в зашифрованном виде (bcrypt).
  1. Политики доступа
* защита от SQL-инъекций и XSS-атак.

1. Ограничения и допущения
   1. Технические ограничения

* только автомобили, приобретенные в автосалоне, имеют доступ к записи на обслуживание;
* приложение рассчитано на использование через веб-браузеры.
  1. Финансовые ограничения
* бюджет проекта: до 300 ₽, включая тестирование и документацию.
  1. Сроки выполнения
* полная реализация проекта — 1,5 месяца с начала разработки

1. Требования к тестированию и приемке
   1. Типы тестирования

* функциональное тестирование: проверка всех реализованных функций;
* тестирование безопасности: проверка на уязвимости.
  1. Критерии приемки
* полное выполнение всех функциональных и нефункциональных требований;
* прохождение не менее 85% тестов по результатам приемочных испытаний.

1. Требования к документации
   1. Пользовательская документация

* руководство пользователя с инструкциями по навигации сайта, записи на обслуживание.
  1. Техническая документация
* описание архитектуры, структура базы данных.

1. План реализации
   1. Этапы разработки
2. анализ требований и проектирование — 2 недели;
3. разработка прототипа интерфейса — 2 неделя;
4. полноценная разработка системы — 4 месяца;
5. тестирование и внедрение — 2 недели.
   1. Сроки выполнения этапов
6. дата начала: 28 октября 2024 года;
7. дата завершения: 20 декабря 2024 года.

10.3 Ответственныелица

Менеджер проекта: Насридинов Н.З.

Ведущий разработчик: Насридинов Н.З.

1 UseCase диаграммы

Диаграмма прецедентов (диаграмма вариантов использования) -диаграмма, отражающая отношения между актёрами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

Диаграмма прецедентов представлена на рисунке 1.

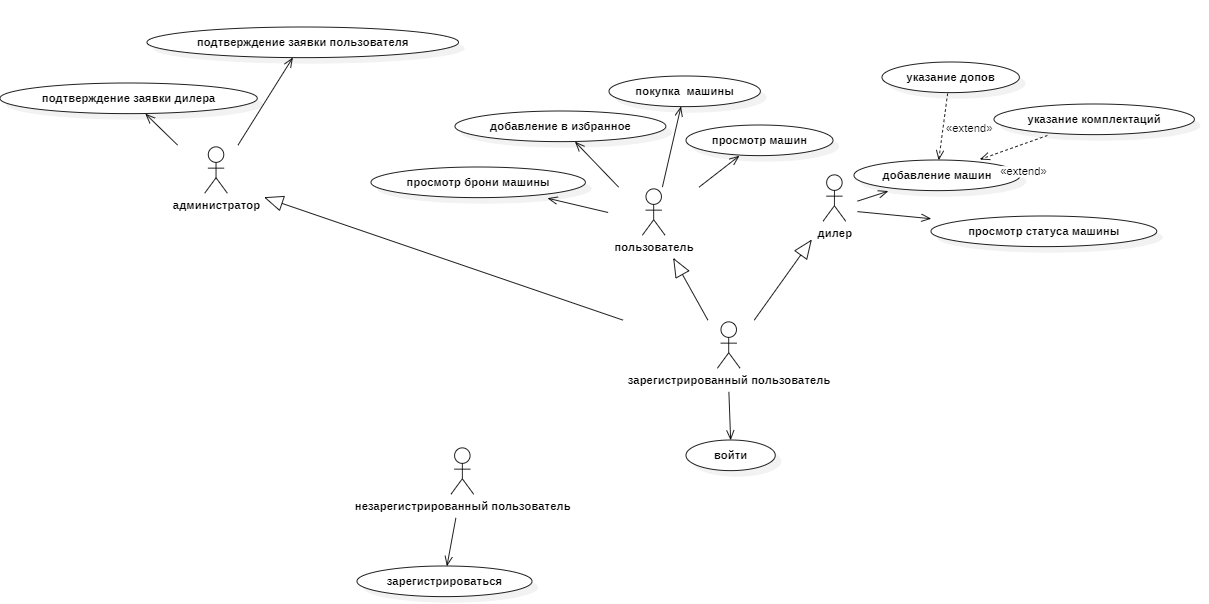


Рисунок 1 — Диаграмма прецедентов

2 Описание структуры базы данных

Схема отношений базы данных представлена на рисунке 2.

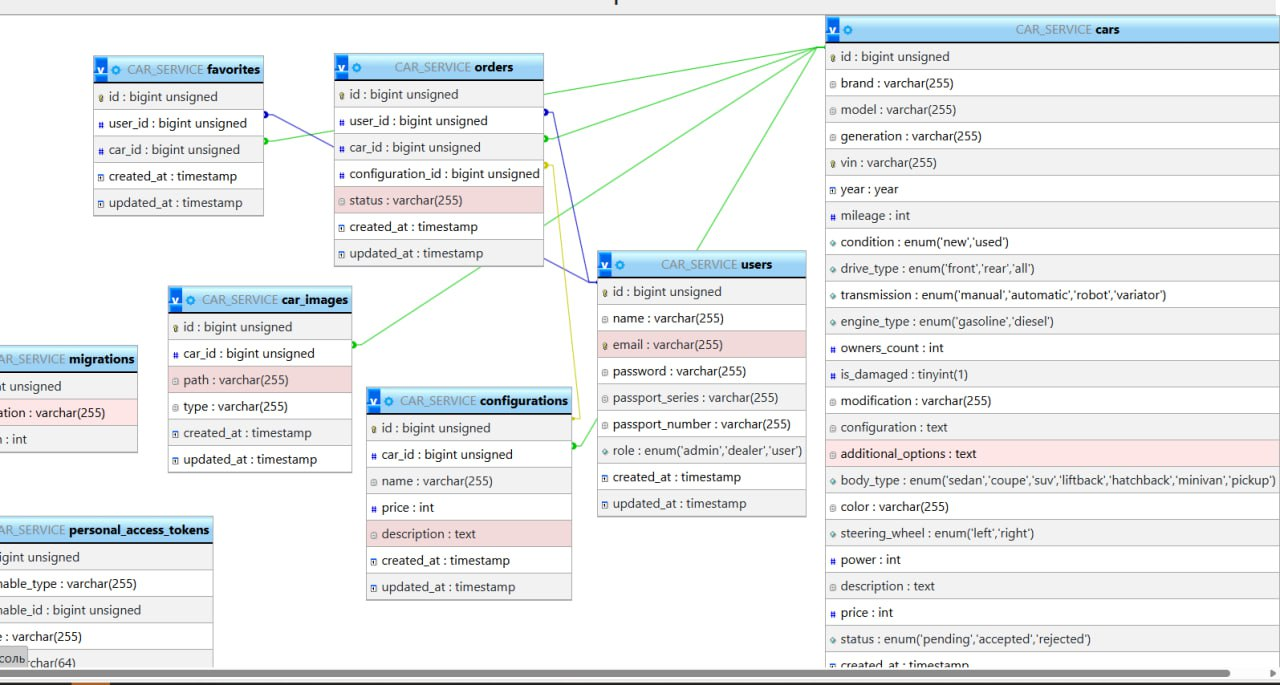


Рисунок 2 – Схема отношений базы данных

Структура таблицы users представлена в таблице 2.1.

Таблица 2.1. - Users(Пользователи)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Код пользователя | Id | INTEGER(4) | Первичный ключ, уникальное значение |
| Почта пользователя | email | VARCHAR(255) | Обязательное поле, уникальное значение |

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| ФИО | name | VARCHAR(255) | Обязательное поле |
| Серия паспорта | passport\_series | VARCHAR(255) | Обязательное поле |
| Номер паспорта | passport\_number | VARCHAR(255) | Обязательное поле |
| Пароль | password | VARCHAR(255) | Обязательное поле |
| Роль | role | ENUM(admin, dealer, user) | Обязательное поле |

Структура таблицы car\_images представлена в таблице 2.2.

Таблица 2.2 – Car\_images (фото машины)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Код | Id | INTEGER(4) | первичный ключ, уникальное значение |
| Код\_машины | Car\_id | INTEGER(4) | Внешний ключ (к Car) |
| Путь картинки | path | VARCHAR(255) | Обязательное поле |
| Тип картинки | type | VARCHAR(255) | Обязательное поле |

Структура таблицы cars представлена в таблице 2.3.

Таблица 2.3 – Cars(машина)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Код | Id | INTEGER(4) | первичный ключ, уникальное значение |
| Бренд | brand | VARCHAR(255) | Обязательное поле |
| Модель | model | VARCHAR(255) | Обязательное поле |
| Поколение | generation | VARCHAR(255) | Обязательное поле |
| Вин номер | vin | VARCHAR(255) | Обязательное поле |
| Год | year | YEAR(4) | Обязательное поле |
| Пробег | mileage | INTEGER(4) | Обязательное поле |
| Состояние | condition | ENUM(new, used) | Обязательное поле |
| Тип привода | drive\_type | ENUM(front, rear, all) | Обязательное поле |

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| Коробка передач | transmission | ENUM(manual, automatic, robot, variator) | Обязательное поле |
| Тип движка | engine\_type | ENUM(gasoline, diesel) | Обязательное поле |
| Количество владельцев | owners\_count | INTEGER(4) | Обязательное поле |
| Битыйне битый | is\_damaged | TINYINT | Обязательное поле |
| Модификация | modification | VARCHAR(255) | NULL |
| Конфигурация | configuration | TEXT | NULL |
| Дополнительные опции | additional\_options | TEXT | NULL |
| Тип кузова | body\_type | ENUM(sedan, coupe, suv, liftback, liftback, hatchback, minivan, pickup) | Обязательное поле |
| Цвет | color | VARCHAR(255) | Обязательное поле |
| Тип руля | steering\_wheel | ENUM(right, left) | Обязательное поле |
| Сила | power | INTEGER(4) | Обязательное поле |
| Описание | description | TEXT | NULL |
| Цена | price | INTEGER(4) | Обязательное поле |
| Статус | status | ENUM(pending, accepted, rejected) | Обязательное поле |

Структура таблицы favorites представлена в таблице 2.4.

Таблица 2.4 – Favorites (Избранное)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Код | Id | INTEGER(4) | первичный ключ, уникальное значение |
| Код пользователя | User\_id | INTEGER(4) | Внешний ключ (к Users) |

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| Код машины | Car\_id | INTEGER(4) | Внешний ключ (к Car) |

Структура таблицы configurations представлена в таблице 2.5.

Таблица 2.5 – Configurations (комплектация)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Код | Id | INTEGER(4) | первичный ключ, уникальное значение |
| Название | name | VARCHAR(255) | Обязательное поле |
| Код машины | Car\_id | BIGINT(20) | Внешний ключ (к Car) |
| Цена | price | INTEGER(4) | Обязательное поле |
| Описание | description | LONGTEXT | Обязательное поле |

Структура таблицы configurations представлена в таблице 2.6.

Таблица 2.6 – Orders (заказы)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Код | Id | INTEGER(4) | первичный ключ, уникальное значение |
| Код покупателя | User\_id | BIGINT(20) | Обязательное поле |
| Код машины | Car\_id | BIGINT(20) | Внешний ключ (к Car) |
| Код комплектации | Configuration\_id | INTEGER(4) | Внешний ключ (к Configuration) |
| Статус | status | VARCHAR(255) | Обязательное поле |