

Содержание

Рекомендации по построению архитектуры.....	2
Лабораторная работа №1	6
Лабораторная работа №2	26
Лабораторная работа №3	34
Лабораторная работа №4	46

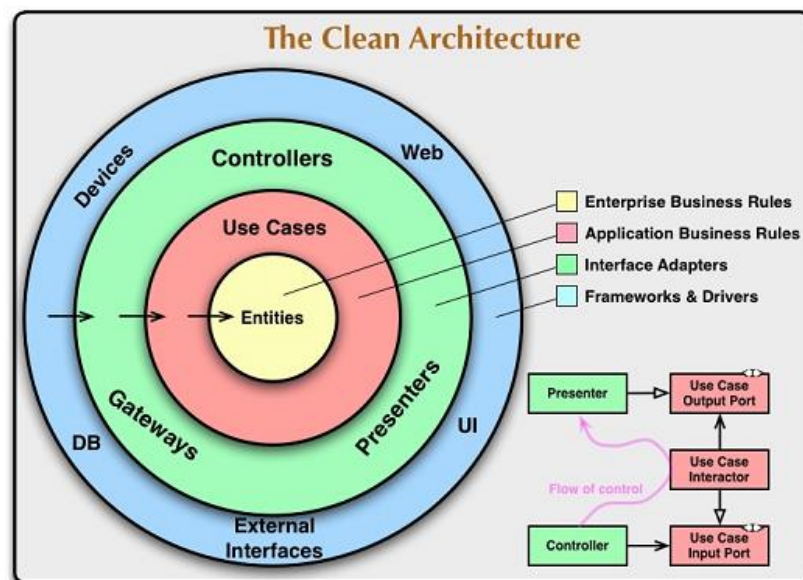
РЕКОМЕНДАЦИИ ПО ПОСТРОЕНИЮ АРХИТЕКТУРЫ

Существует достаточно много подходов для построения сложных систем с хорошей архитектурой. Несмотря на небольшие различия этих подходов, у них много общего. Разумеется, они все задают способы разбиения приложения на отдельные модули. При этом в каждой системе как минимум есть модули, содержащие бизнес-логику приложения, и модули для отображения данных. И каждый подход в итоге позволяет построить систему, которая удовлетворяет следующим принципам:

1. Архитектура должна быть независима от различных фреймворков. Разумеется, в современном мире мы не можем обходиться без каких-то библиотек, которые позволяют решать задачи намного быстрее и чаще эффективнее, чем это сделали бы мы в случае самостоятельной реализации. Но здесь важно понимать, что библиотека должна встраиваться в вашу архитектуру, а не архитектура должна подстраиваться под выбранную библиотеку. При использовании библиотеки, которая вынуждает вас переделывать архитектуру приложения, вы всегда будете сталкиваться с определенными ограничениями этой библиотеки и не сможете перестраивать архитектуру нужным вам образом. Нужно использовать библиотеки только в качестве вспомогательных инструментов.
2. Система должна быть протестирована. При этом вы должны иметь возможность тестировать как модули системы по отдельности, так и тестировать взаимодействие этих модулей между собой и интеграцию их в систему. Кроме того, необходимо тестировать систему без UI, реального сервера и работы с базой данных, то есть архитектура должна быть независима от окружения.
3. Из предыдущих пунктов плавно вытекают и следующие принципы, которые говорят о том, что ваше приложение должно быть независимо от всего: от интерфейса пользователя, от работы с базой данных, от работы сервера и от других элементов окружения. Независимость

архитектуры от окружения очень важна, так как это позволяет менять различные компоненты окружения без изменения самой архитектуры. Что подразумевается под изменением компонентов архитектуры? Это может быть изменение в выборе базы данных (или же вообще отказ от нее) или же изменение в UI-части приложения (к примеру, нужно изменить внешний вид экрана). Если для внесения таких изменений требует от вас изменения архитектуры, то, возможно, вам стоит пересмотреть архитектуру.

Для построения архитектуры системы предлагается рассмотреть следующую схему, где каждая концентрическая окружность является определенным компонентом системы:



Как видно из схемы, система состоит из бизнес-объектов, объектов для управления данными и бизнес-логикой, из слоя представления и слоя высокоуровневых фреймворков. Можно сказать, что в общем случае это все такое же разбиение на представление данных и логику работы с ними.

Такое разделение системы на слои является весьма логичным и понятным. И пока оно не привносит ничего нового. Главным является правило зависимостей – ни один внутренний слой не должен знать ничего о внешнем. Именно это и позволяет строить независимую архитектуру, принципы которой были описаны выше.

Рассмотрим элементы системы более детально, чтобы понять, в чем заключается роль каждого элемента и как он должен взаимодействовать с другими элементами системы.

Бизнес-объекты

С точки зрения работы с системой в итоге мы всегда работаем с определенными сущностями, которые определяются требованиями системы. По сути бизнес-объекты – это классы моделей с определенными методами или же набор каких-то структур данных. Эти классы отвечают логике вашего приложения, и они должны определять самые общие правила поведения.

Поскольку это самый внутренний слой, то он будет меняться только в крайнем случае, когда вы решите изменить саму суть системы. И, разумеется, он останется нетронутым, когда будет меняться, к примеру, способ работы с данными или интерфейс пользователя.

Сценарии взаимодействия

Этот слой содержит реализацию основных методов для работы системы и организует работу с данными и бизнес-объектами. Он использует бизнес-объекты и их логику для того, чтобы выполнить свои задачи. Этот слой можно рассматривать как некоторый посредник между бизнес-объектами и непосредственно слоем представления данных (такое определение немного условно, так как по правилу зависимостей этот посредник не должен знать ничего о слое представления).

Разумеется, в силу правила зависимостей этот слой также не будет меняться при изменении какого-то из элементов окружения. И изменения на этом слое в свою очередь не будут затрагивать бизнес-объекты.

Слой представления

На этом слое осуществляется преобразование данных из формата, который используют бизнес-объекты или сценарии взаимодействия, в формат, необходимый для работы системы. Под работой системы в данном случае подразумевается передача данных для отображения пользователю или другой службе.

На этом слое реализуются такие архитектурные паттерны как Model-View-Controller или Model-View-Presenter. К примеру, Controller может передавать данные в сценарии взаимодействия, получать результат и передавать его для отображения во View.

Фреймворки

Изначально мы говорили, что система не должна зависеть ни от каких фреймворков. Именно поэтому слой фреймворков на схеме является самым внешним. Поскольку именно на этом уровне пропадает вся абстракция, и мы используем конкретные средства для решения определенных задач. Под конкретными средствами здесь подразумевается база данных, фреймворк для UI и другие службы.

Поскольку мы соблюдаем правило зависимостей, внутренние слои ничего не знают о конкретных используемых фреймворках, что позволяет легко изменить любой из них без изменения внутренних слоев.

В описанной схеме указано 4 слоя, и этого достаточно для большинства систем. Но, разумеется, при необходимости эту систему можно масштабировать и увеличивать количество слоев. Самое главное при этом – сохранять правило зависимостей.

Использование такого подхода позволяет создать архитектуру, принципы которой были описаны в начале раздела. В силу правила зависимостей мы получаем соответствие принципам 1 и 3, а такое разбиение системы на модули позволяет тестировать ее на разных уровнях.

Однако нужно понимать, что это теоретическое изложение, которое является достаточно общим и должно удовлетворять любой системе. На практике всегда нужно адаптировать архитектуру под конкретную задачу или систему.

ЛАБОРАТОРНАЯ РАБОТА №1

Необходимо разработать мобильное приложение для планшета, удовлетворяющее следующим требованиям:

Минимальная версия ОС, поддерживаемая приложением, должна быть: Android - 11.0, iOS - 14.0.

В качестве бэкенда будет использован Supabase (<https://supabase.com/>). Для добавления в организацию курса необходимо написать свою почту преподавателю.

Данные по пользователям, автомобилям и бронированиям находятся в таблицах Supabase.

В работе необходимо использовать систему контроля версий Git. Репозиторий может быть открытым или закрытым.

Необходимо следовать предложенному дизайну. Реализация дополнительных экранов и логики может быть при желании осуществлена в соответствии с требованиями проекта и архитектурными решениями.

Необходимо осуществлять комментирование кода в созданных классах (Это поможет вам быстрее вспомнить, что делает класс на защите работы и не только). Обязательны следующие комментарии:

- Описание назначения класса

Необходимо корректно обрабатывать запросы к серверу. В случае получения ошибки от сервера необходимо отобразить соответствующий текст ошибки в диалоговом окне.

В случае отсутствия доступа в интернет на устройстве, необходимо отображать экран No Connection (Название экрана из Figma).

Необходимо отображать индикацию выполнения запроса с сервером. Приложение должно быть разработано с учетом устойчивости к сбоям и предусматривать механизмы обработки исключительных ситуаций, гарантирующие его непрерывную работу при любых обстоятельствах. В случае возникновения непредвиденных ошибок или сбоев, приложение должно сохранять свою функциональность и обеспечивать корректное

выполнение всех критически важных процессов, минимизируя влияние этих ситуаций на пользовательский опыт.

При разработке проекта приложения вам необходимо использовать архитектуру (см. раздел с описанием архитектуры), в которой будут разделены слои бизнес-логики, представлений и домена.

В слое представления необходимо явно выделить презентер или вьюмодель, все взаимодействие с бизнес-логикой должно осуществляться из презентера или вьюмодели.

Файлы проекта распределены по папкам в соответствии с архитектурой.

Вся верстка должна быть адаптивной (следует учитывать разные размеры экранов). Приложение должно корректно отображаться как на устройствах с маленькими экранами смартфонов, так и на устройствах с большими экранами, обеспечивая оптимальный пользовательский интерфейс и удобство взаимодействия независимо от размера экрана. Необходимо:

- Избегайте появления большого пустого пространства;
- Следите за отсутствием искажения элементов;
- Все элементы должны полностью находиться в границах и на месте, указанном в макете;
- Учитывайте расстояние между элементами;
- Используйте шрифты согласно макету.
- Дизайн предложен в Figma:

<https://www.figma.com/design/MpSh8Dp97oCbDJLFmW9Dko/DriveNext?node-id=3908-2710&t=4qTtKVar4IwGBCQN-1>

Экран «Экран загрузки»

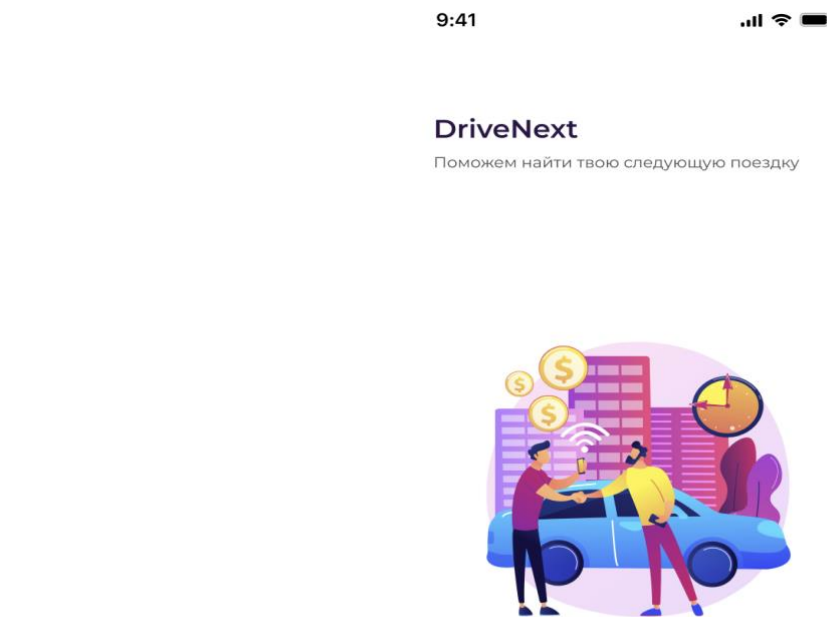


Рисунок 1 - Экран загрузки

Общее описание

Реализуйте экран для входа в приложении согласно макету. Данный экран отображается каждый раз при входе в приложение.

Функциональные требования

1. Отображение названия, логотипа и слогана приложения:

- Название отображается сверху по левому краю экрана.
- Слоган «Поможем найти твою следующую поездку» отображается под логотипом.
- Логотип приложения отображается в центре экрана.

2. Время отображения:

- Экран загрузки отображается на экране в течение 2-3 секунд или до завершения инициализации приложения.

Переходы

- После завершения загрузки автоматически, без дополнительного действия со стороны пользователя, происходит переход на экран входа в приложение, если пользователь открывает его первый раз, либо access token недействителен. Либо на главный экран приложения.

Экран «Нет подключения к интернету»



Рисунок 2 - Нет подключения к интернету

Общее описание

Реализуйте экран для отображения отсутствия интернета в приложении согласно макету.

Функциональные требования

1. Сообщение об ошибке:

- Текст «Нет подключения к интернету» отображается в центре экрана.
- Под основным сообщением отображается текст «Проверьте подключение и повторите попытку», информирующий пользователя о необходимости проверки подключения.
- Над текстом отображается иконка, символизирующая отсутствие соединения.

2. Кнопки:

- «Повторить попытку»

Кнопка становится активной сразу после отображения экрана.

- По нажатию на кнопку осуществляется повторная попытка подключения к интернету.
- Успех: приложение продолжает работу с переходом на главный экран или текущий процесс, требующий подключения.
- Ошибка: экран остается активным и снова отображает сообщение об ошибке.

Переходы

- В случае успешного подключения приложение переходит на главный экран или возвращается к процессу, который был прерван из-за отсутствия интернета.

Экран «Приветствия»

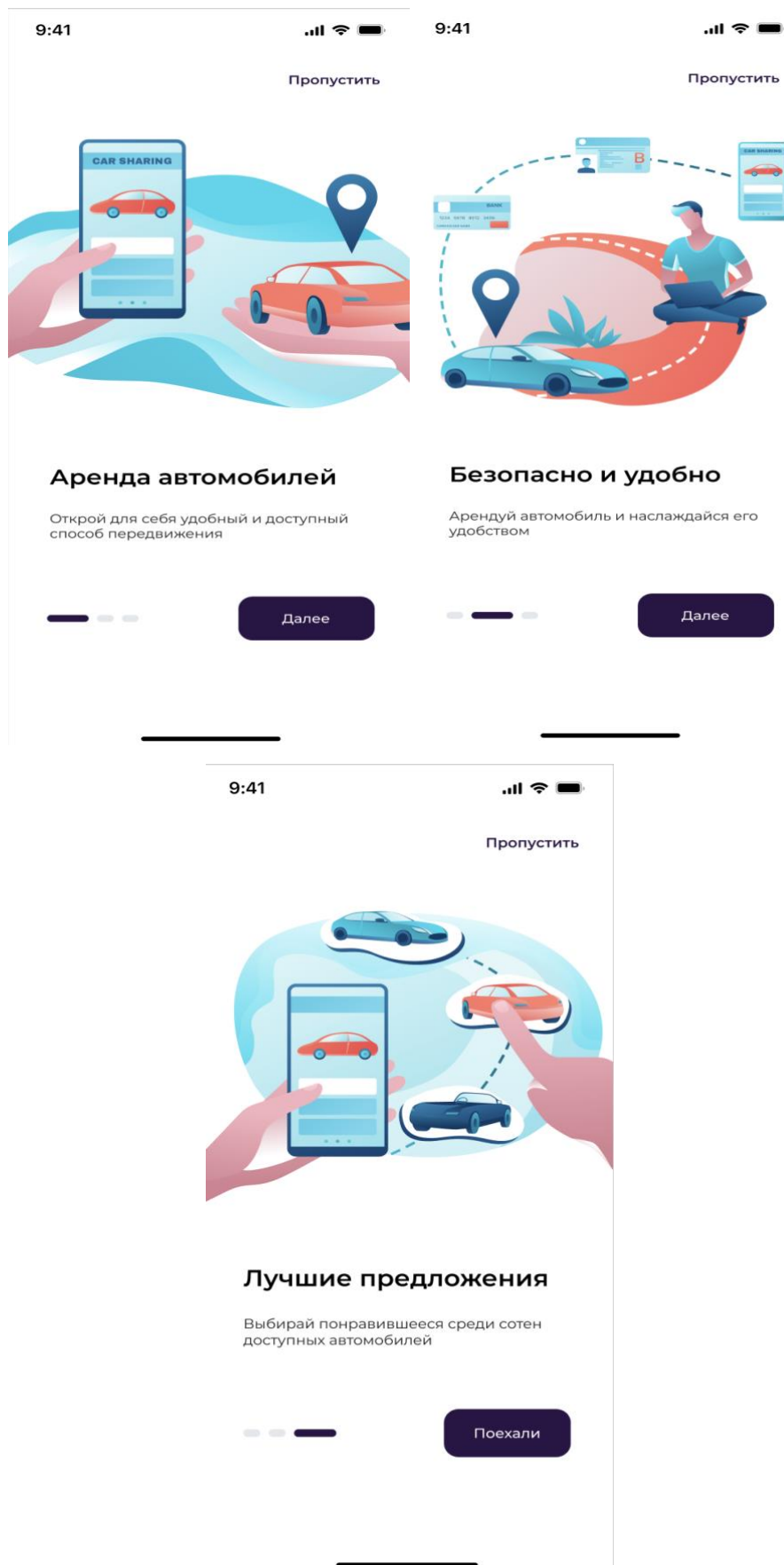


Рисунок 3 - Приветствие

Общее описание

Реализуйте экран приветствия для отображения преимуществ приложения согласно макету. Данный экран отображается при первом открытии приложения, если у пользователя нет актуального access токена.

Функциональные требования

1. Отображение основных преимуществ:

- Экран приветствия состоит из нескольких последовательных шагов (слайдов), которые пользователь может пролистать, чтобы ознакомиться с основными функциями и преимуществами приложения.
- Каждый слайд содержит заголовок и краткое описание функциональности, представленной на изображении.

2. Кнопки:

- **«Пропустить»**

Кнопка становится активной сразу после отображения экрана.

- При нажатии на кнопку пользователь сразу переходит к экрану входа/регистрации.

- **«Далее»**

Кнопка становится активной сразу после отображения экрана.

- При нажатии на кнопку пользователь видит следующий слайд.

- **«Поехали»**

Кнопка становится активной на последнем слайде.

- При нажатии на кнопку пользователь переходит к экрану входа/регистрации.

Переходы

- После прохождения приветственных экранов пользователь переходит на экран входа/регистрации.

Хранение данных

- После завершения процесса приветствия (Onboarding) информация о том, что пользователь прошел этот процесс, сохраняется, чтобы не показывать его повторно при последующих запусках приложения.

Экран «Входа/регистрации»

9:41



DriveNext

Поможем найти твою следующую поездку



Войти

Зарегистрироваться

Рисунок 4 - Вход/регистрация

Общее описание

Реализуйте экран выбора действия (входа/регистрации) для отображения основных опций приложения согласно макету. Экран предоставляет пользователю возможность либо войти в существующую учетную запись, либо зарегистрироваться, чтобы начать использовать приложение.

Функциональные требования

1. Отображение названия, логотипа и слогана приложения:

- Название отображается сверху по левому краю экрана.
- Слоган «Поможем найти твою следующую поездку» отображается под логотипом.
- Логотип приложения отображается в центре экрана.

2. Кнопки:

- «Войти»

Кнопка становится активной сразу после отображения экрана.

- ведет пользователя на экран авторизации, где он может ввести свои учетные данные для входа в приложение.

- **«Зарегистрироваться»**

Кнопка становится активной сразу после отображения экрана.

- ведет пользователя на экран регистрации, где он может создать новый аккаунт.

Переходы

- Пользователь переходит на следующие экраны в зависимости от того, на какую кнопку было произведено нажатие: «Войти» пользователь переходит на экран авторизации, «Зарегистрироваться» пользователь переходит на экран регистрации.

Экран «Вход в аккаунт»

9:41



Войдите в аккаунт

Пожалуйста, введите свои данные

Электронная почта

Пароль



[Забыли пароль?](#)

Войти



Войти через Google

[Зарегистрироваться](#)

Рисунок 5 – Вход в аккаунт

Общее описание

Реализуйте экран входа в приложения согласно макету. Точкой входа в данный экран является нажатие на кнопку «Войти» на экране входа/регистрации.

Функциональные требования

1. Поля ввода:

- Электронная почта: обязательное поле, проверка корректности (паттерн «name@domainname.com»).
- Пароль: обязательное поле, возможность показать/скрыть пароль при нажатии на иконку глаза.

2. Кнопки:

- «Войти»

Кнопка становится активной только после того, как будут заполнены все поля.

- По нажатию отправить запрос на авторизацию.
- Успех: сохранить токен, открыть «главный экран».
- Ошибка: отобразить сообщение об ошибке.
- **«Войти через Google»**
 - По нажатию инициировать процесс авторизации через Google OAuth.
 - Успех: сохранить токен, открыть пустой «главный экран».
 - Ошибка: отобразить сообщение об ошибке.
- **«Зарегистрироваться»**
 - По нажатию перейти на экран регистрации.

Переходы

- При успешной авторизации открыть «Главный экран».
- При нажатии на «Зарегистрироваться» перейти на экран регистрации.

Хранение данных

- Сохранить данные входа и активную сессию после перезапуска приложения.

Экран «Регистрация» (шаг 1)

9:41

< Создать аккаунт

Электронная почта

Введите электронную почту

Придумайте пароль

Введите пароль

Повторите пароль

Введите пароль

☐ Я согласен с условиями обслуживания и политикой конфиденциальности

Далее

Рисунок 6 – Регистрация (шаг 1)

Общее описание

Реализуйте экран для создания нового аккаунта пользователя в приложении согласно макету. Точкой входа в данный экран является нажатие на кнопку «Зарегистрироваться» на экране входа/регистрации или на экране входа.

Функциональные требования

1. Поля ввода:

- **Электронная почта:**
 - Обязательное поле, проверка корректности (паттерн «name@domainname.ru»).
- **Придумайте пароль:**
 - Обязательное поле, возможность показать/скрыть пароль при нажатии на иконку глаза.

- **Повторите пароль:**
 - Обязательное поле, проверка совпадения с полем «Придумайте пароль», возможность показать/скрыть пароль при нажатии на иконку глаза.

2. Чекбокс:

- **Согласие с условиями обслуживания и политикой конфиденциальности:**
 - Обязательный для отметки перед продолжением.

Переходы

1. Кнопка Далее»:

- Перейти на следующий экран регистрации.

2. Кнопка «Назад»:

- По нажатию вернуть пользователя на предыдущий экран.

Сообщения об ошибках

Любым образом отобразите сообщения об ошибках

- Некорректный email: «Введите корректный адрес электронной почты.»
- Несовпадающие пароли: «Пароли не совпадают.»
- Неотмеченный чекбокс: «Необходимо согласиться с условиями обслуживания и политикой конфиденциальности.»

Экран «Регистрация» (шаг 2)

9:41

< Создать аккаунт

Фамилия

Введите фамилию

Имя

Введите имя

Отчество

Введите отчество

Дата рождения

DD/MM/YYYY

Пол

☐ Мужской ☒ Женский

Далее

Рисунок 7 – Регистрация (шаг 2)

Общее описание

Реализуйте экран для ввода дополнительных данных при создании нового аккаунта пользователя в приложении согласно макету.

Функциональные требования

1. Поля ввода:

- **Фамилия:** обязательное поле.
- **Имя:** обязательное поле.
- **Отчество:** необязательное поле.
- **Дата рождения:** обязательное поле, проверка корректности формата (MM/DD/YYYY).
- **Пол:** обязательное поле, выбор между «Мужской» и «Женский».

2. Кнопки:

- **«Далее»:**

Кнопка становится активной только после того, как будут заполнены обязательные поля.

- Успех: завершить регистрацию и перейти на главный экран.
- Ошибка: отобразить сообщение об ошибке.

- **«Назад»:**

- По нажатию вернуться на предыдущий экран регистрации (шаг 1).

Переходы

- При успешной проверке данных перейти на следующий экран регистрации.
- При нажатии «Назад» перейти на первый экран регистрации.

Сообщения об ошибках

- Незаполненные обязательные поля: «Пожалуйста, заполните все обязательные поля.»
- Некорректная дата рождения: «Введите корректную дату рождения.»

Экран «Регистрация» (шаг 3)

9:41

< Создать аккаунт

Добавление фотографии поможет владельцам и арендаторам узнать друг друга, когда они будут забирать машину

Номер водительского удостоверения

0000000000

Дата выдачи

DD/MM/YYYY

Загрузите фото водительского удостоверения

Загрузить фото

Загрузите фото паспорта

Загрузить фото

Далее

Рисунок 8 – Регистрация (шаг 3)

Общее описание

Реализуйте экран для загрузки документов и фото пользователя при создании нового аккаунта в приложении согласно макету.

Функциональные требования

1. Добавление фотографии:

- Иконка для добавления фото профиля: обязательно для заполнения.
- По нажатию на иконку открыть галерею или камеру для выбора фото.

2. Поля ввода:

- **Номер водительского удостоверения:** обязательное поле.

- **Дата выдачи:** обязательное поле, проверка корректности формата (DD/MM/YYYY).

3. Загрузка документов:

- **Загрузить фото водительского удостоверения:**
 - Кнопка «Загрузить фото» для открытия галереи или камеры.
- **Загрузить фото паспорта:**
 - Кнопка «Загрузить фото» для открытия галереи или камеры.

4. Кнопки:

- **«Далее»:**

Кнопка становится активной только после того, как будут заполнены обязательные поля.

- Успех: завершить регистрацию и перейти на экран «Успешная регистрация».
 - Ошибка: отобразить сообщение об ошибке.
- **«Назад»:**
 - По нажатию вернуться на предыдущий экран регистрации (шаг 2).

Сообщения об ошибках

- Незаполненные обязательные поля: «Пожалуйста, заполните все обязательные поля.»
- Некорректная дата выдачи: «Введите корректную дату выдачи.»
- Отсутствие загруженных фото: «Пожалуйста, загрузите все необходимые фото.»

Экран «Успешная регистрация»

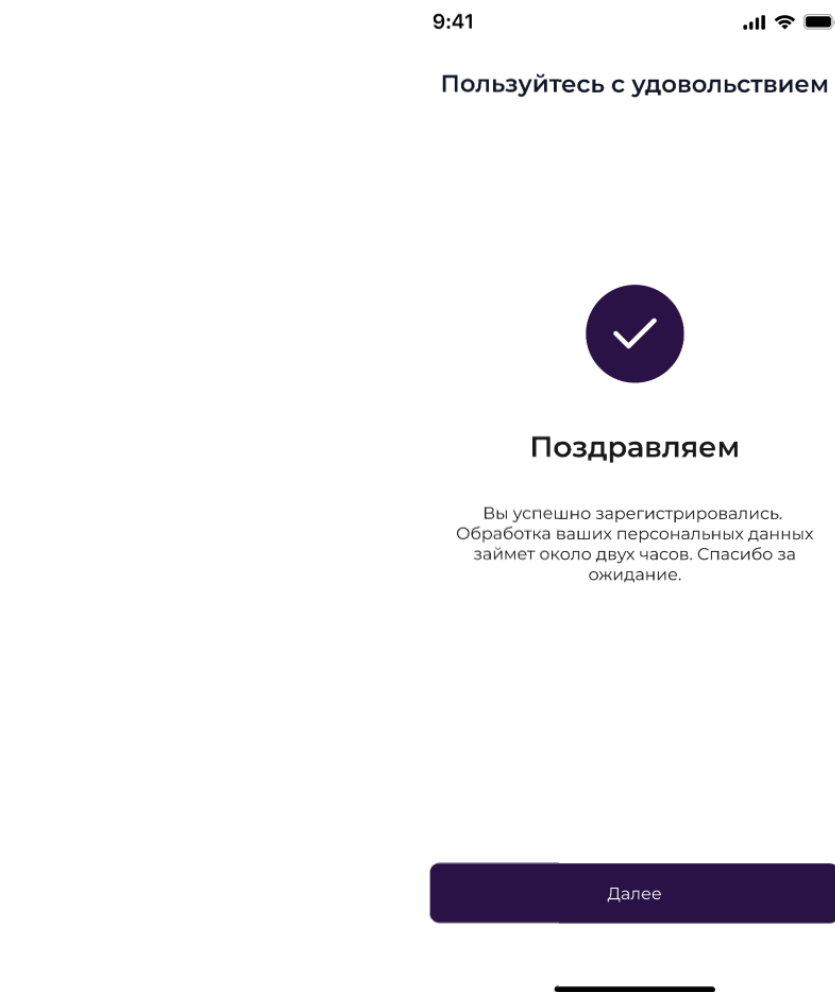


Рисунок 9 – Успешная регистрация

Общее описание

Реализуйте экран поздравления после успешной регистрации пользователя в приложении согласно макету.

Функциональные требования

1. Кнопка:

- «Далее»: перейти на главный экран приложения.

ЛАБОРАТОРНАЯ РАБОТА №2

Экран «Настройки»

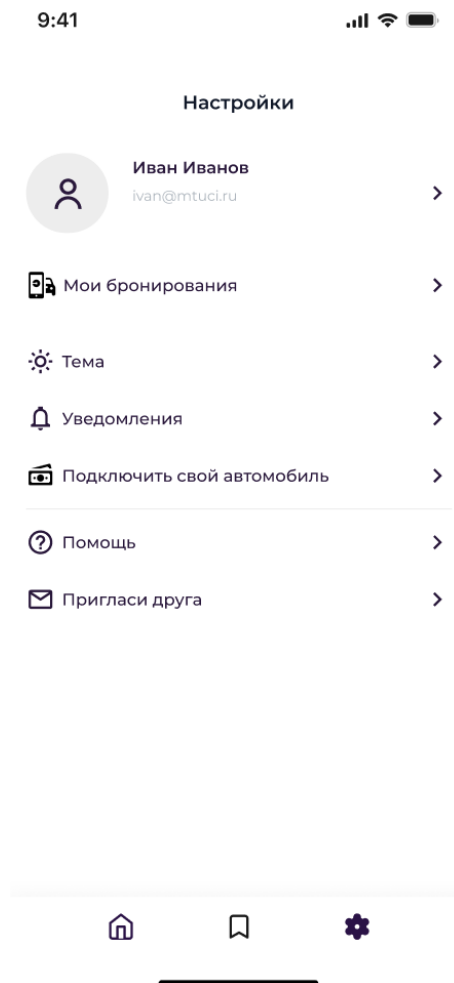


Рисунок 10 – Настройки

Общее описание

Реализуйте экран настроек пользователя в приложении согласно макету. Точкой входа в данный экран является нажатие на иконку шестеренки на нижней панели навигации.

Функциональные требования

1. **Профиль пользователя:**
 - Отображение аватара пользователя.
 - Отображение имени пользователя.
 - Отображение электронной почты пользователя.
 - По нажатию переход на экран «Профиль».
2. **Меню настроек:**

- **Мои бронирования:** по нажатию переход на экран «Мои бронирования»
- **Тема**
- **Уведомления**
- **Подключить свой автомобиль:** по нажатию переход на экран добавления автомобиля.
- **Помощь**
- **Пригласи друга**

Хранение данных

- Сохранение изменений профиля, тем и настроек уведомлений после перезапуска приложения.

Экран «Профиль»

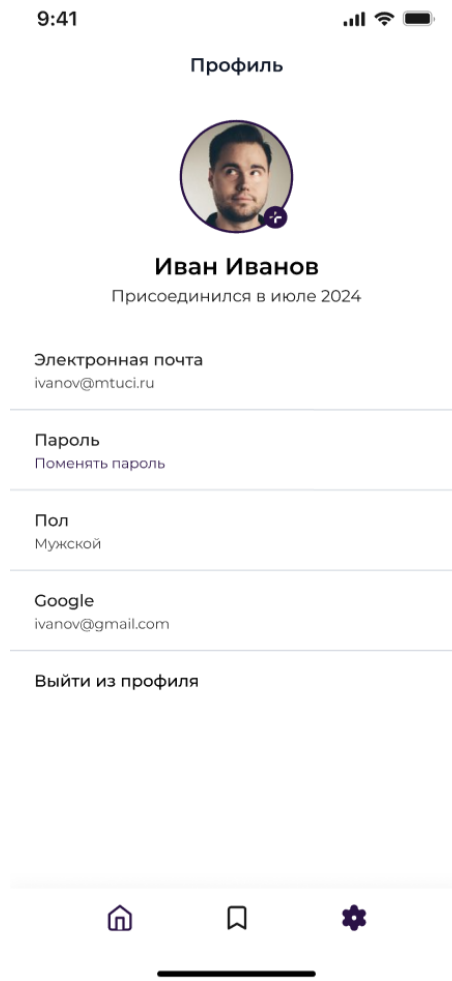


Рисунок 11 – Профиль

Общее описание

Реализуйте экран профиля пользователя в приложении согласно макету. Точкой входа является нажатие на профиль пользователя на экране «Настройки»

Функциональные требования

1. Аватар пользователя:

- Отображение текущего аватара.
- По нажатию открытие возможности загрузить новый аватар.

2. Кнопка «Выйти из профиля»:

- По нажатию выполнение выхода пользователя из аккаунта и возвращение на экран входа в аккаунт.

Переходы

- При нажатии на «Выйти из профиля» выполнение действия выхода и переход на экран входа.

Экран «Главная»

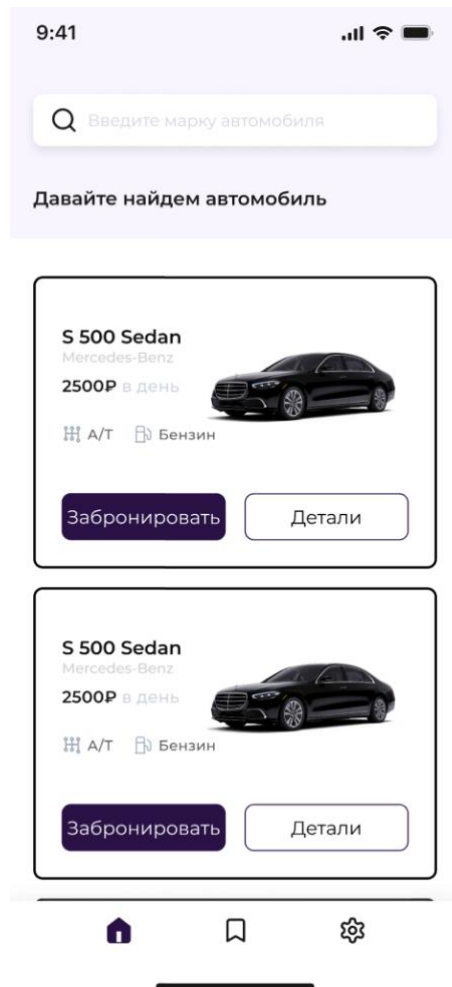


Рисунок 12 – Главная

Общее описание

Реализуйте экран для отображения списка доступных автомобилей для аренды в приложении согласно макету. Точкой входа в данный экран является нажатие на иконку дома на нижней панели навигации.

Функциональные требования

1. Отображение списка автомобилей:

- Список всех доступных автомобилей должен быть загружен с сервера при открытии экрана.
- Каждый автомобиль должен отображаться в виде карточки с информацией о модели, марке, цене и основных характеристиках.

2. Поиск автомобилей:

- Введите марку автомобиля в поле поиска.

- После ввода марки должен осуществляться поиск автомобилей данной марки и переход на экран «Результаты поиска».

3. Действия с автомобилем:

- **Забронировать автомобиль:**
 - По нажатию на кнопку «Забронировать» переход на экран «Оформление аренды»
- **Просмотр деталей:**
 - По нажатию на кнопку «Детали» переход на экран «Детали»

4. Загрузка данных:

- При загрузке данных должен отображаться индикатор загрузки.
- В случае ошибки загрузки данных показывать сообщение об ошибке и кнопку для повторной попытки загрузки.

Сообщения об ошибках

- Ошибка поиска: «Не удалось выполнить поиск. Попробуйте снова.»
- Ошибка загрузки данных: «Не удалось загрузить данные. Попробуйте снова.»

Экран «Результаты поиска»

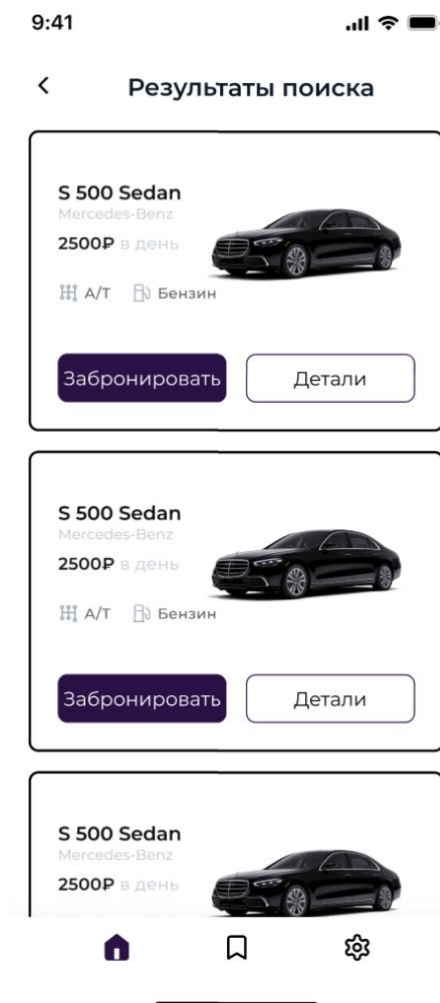


Рисунок 13 – Результаты поиска

Общее описание

Реализуйте экран для отображения результатов поиска автомобилей в приложении согласно макету. Точкой входа в данный экран является ввод запроса в поле поиска на экране «Главная».

Функциональные требования

1. Отображение результатов поиска:

- Результаты поиска автомобилей должны быть загружены с сервера на основе введенного запроса.
- Каждый автомобиль должен отображаться в виде карточки с информацией о модели, марке, цене и основных характеристиках.

2. Действия с автомобилем:

- **Забронировать автомобиль:**

- По нажатию на кнопку «Забронировать» переход на экран «Оформление аренды».
- **Просмотр деталей:**
 - По нажатию на кнопку «Детали» переход на экран «Детали».

3. Загрузка данных:

- При загрузке данных должен отображаться индикатор загрузки.
- В случае ошибки загрузки данных показывать сообщение об ошибке и кнопку для повторной попытки загрузки.

4. Обновление данных:

- Обновление списка автомобилей при изменении поискового запроса.
- Обновление результатов поиска при перезапуске приложения.

Сообщения об ошибках

Любым способом отобразите сообщения об ошибках в случае их возникновения:

- Ошибка поиска: «Не удалось выполнить поиск. Попробуйте снова.»
- Ошибка загрузки данных: «Не удалось загрузить данные. Попробуйте снова.»

ЛАБОРАТОРНАЯ РАБОТА №3

Экран «Детали»



Рисунок 14 – Детали

Общее описание

Реализуйте экран для отображения детальной информации об автомобиле в приложении согласно макету. Точкой входа в данный экран является нажатие на кнопку «Детали» на экране результатов поиска или каталога автомобилей.

Функциональные требования

1. Действия с автомобилем:

- **Забронировать автомобиль:**
 - По нажатию на кнопку «Забронировать» переход на экран «Оформление аренды».

- **Добавить в избранное:**
 - По нажатию на иконку «сердце» добавление автомобиля в список избранного.

2. Загрузка данных:

- При загрузке данных должен отображаться индикатор загрузки.
- В случае ошибки загрузки данных показывать сообщение об ошибке.

3. Навигация:

- **Кнопка «Назад»:** по нажатию возврат на предыдущий экран.

Сообщения об ошибках

Любым способом отобразите сообщения об ошибках в случае их возникновения:

- Ошибка загрузки данных: «Не удалось загрузить данные. Попробуйте снова.»
- Ошибка добавления в избранное: «Не удалось добавить в избранное. Попробуйте снова.»

Экран «Оформление аренды»

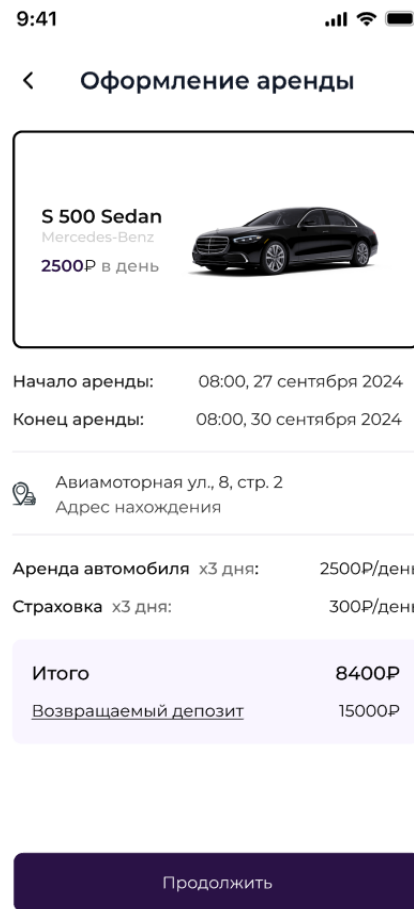


Рисунок 15 – Оформление аренды

Общее описание

Реализуйте экран для оформления бронирования автомобиля в приложении согласно макету. Точкой входа в данный экран является нажатие на кнопку «Забронировать» на экране результатов поиска, каталога автомобилей или экране деталей.

Функциональные требования

1. Отображение информации о бронировании:

- Название автомобиля, модель, марка и стоимость аренды в день.
- Даты начала и конца аренды с возможностью их выбора.
- Адрес местоположения автомобиля.

2. Стоимость:

- Отображение стоимости аренды автомобиля за весь период.

- Отображение стоимости страховки за весь период.
- Отображение итоговой суммы (с учетом аренды и страховки).
- Отображение возвращаемого депозита.

3. Кнопка «Продолжить»:

- При нажатии на кнопку происходит переход на следующий шаг оформления аренды.

Переходы

- **Кнопка «Продолжить»:** Переход на следующий экран, уведомляющий пользователя об успешном бронировании.

Экран «Удачной поездки»

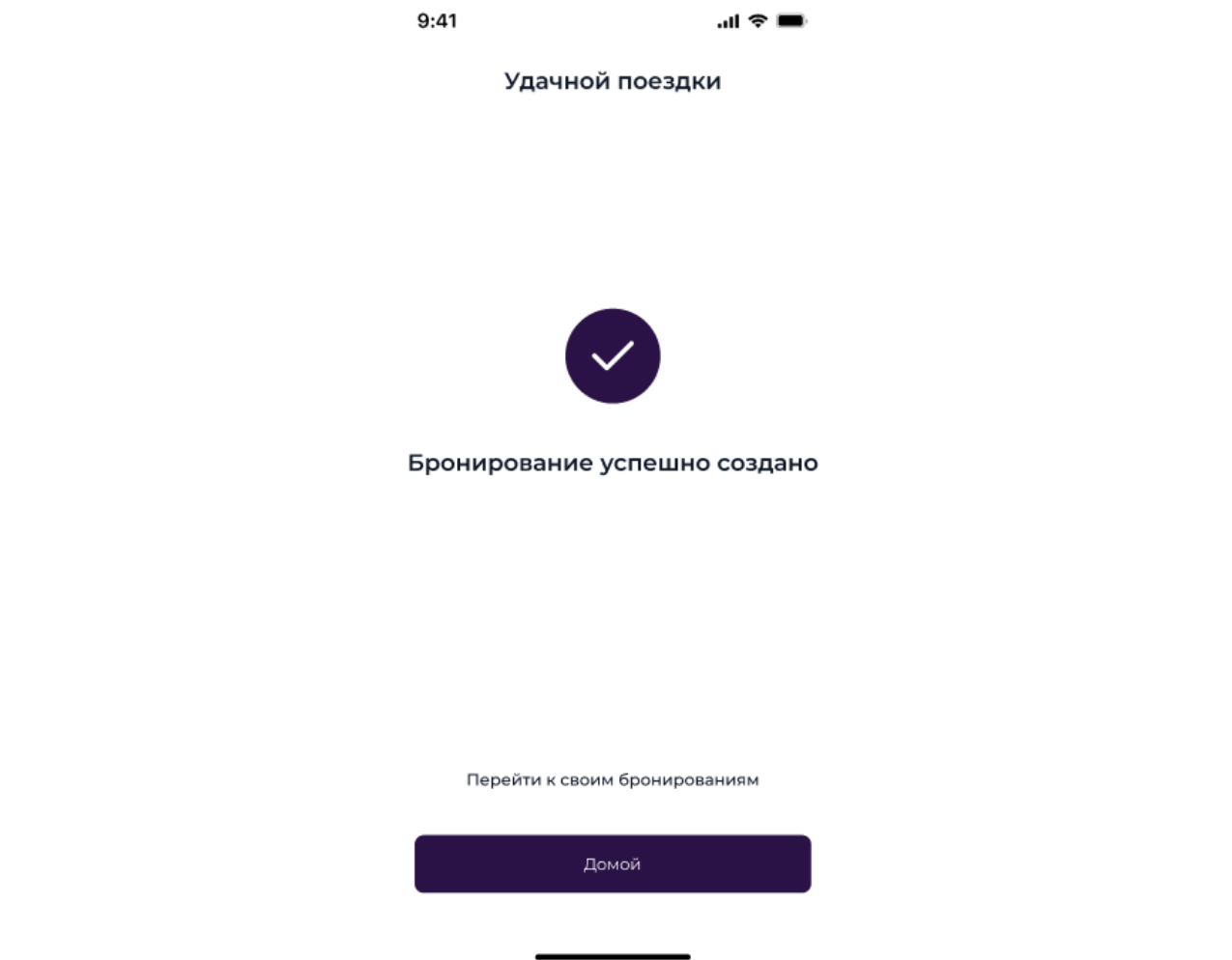


Рисунок 16 – Удачной поездки

Общее описание

Реализуйте экран для отображения успешного бронирования автомобиля в приложении согласно макету. Точкой входа в данный экран является нажатие на кнопку «Далее» на экране оформления бронирования.

Функциональные требования

1. Сообщение об успешном бронировании:

- Отображение сообщения о том, что бронирование было успешно создано.

2. Кнопки навигации:

- «Перейти к своим бронированиям»: По нажатию на эту кнопку пользователь переходит на экран с перечнем его активных бронирований.

- **«Домой»:** По нажатию на эту кнопку пользователь возвращается на главный экран приложения.

Экран «Мои бронирования»

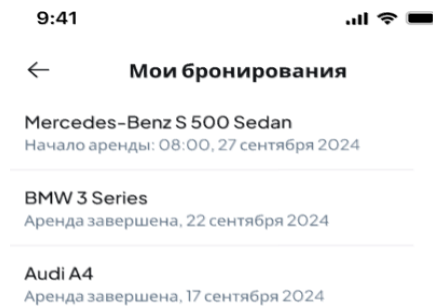


Рисунок 17 – Мои бронирования

Общее описание

Реализуйте экран для отображения списка активных и завершенных бронирований автомобилей в приложении согласно макету. Точкой входа в данный экран является нажатие на кнопку «Перейти к своим бронированиям» с экрана успешного бронирования или выбор пункта «Мои бронирования» на экране настроек.

Функциональные требования

1. Отображение списка бронирований:

- Список включает активные и завершенные бронирования.
- Для каждого активного бронирования отображается марка, модель автомобиля, а также дата и время начала аренды.
- Для завершенных бронирований отображается марка, модель автомобиля, а также дата завершения аренды.

Переходы

- **Нажатие на активное бронирование:** Переход на экран с детальной информацией о бронировании.
- **Кнопка «Назад»:** Возвращение на предыдущий экран.

Экран «Детальная информация о бронировании»

9:41

< Бронирование # 874659

S 500 Sedan
Mercedes-Benz
2500Р в день

Адрес нахождения Авиамоторная ул., 8, стр. 2

Начало аренды 08:00, 27 сентября 2024

Конец аренды 08:00, 30 сентября 2024

ФИО водителя Иван Иванов

Номер ВУ 45164634

Статус аренды Одобрено

Аренда x3 дня: 2500Р/день

Страховка x3 дня: 300Р/день

Общая сумма 8400Р

Отменить бронирование

Рисунок 18 – Детальная информация о бронировании

Общее описание

Реализуйте экран для отображения детальной информации о конкретном бронировании автомобиля в приложении согласно макету. Точкой входа в данный экран является нажатие на активное бронирование на экране списка бронирований.

Функциональные требования

1. Отображение информации о бронировании:

- Номер бронирования.
- Название автомобиля, марка, модель и стоимость аренды в день.
- Адрес местоположения автомобиля.
- Даты начала и окончания аренды.
- ФИО водителя.

- Номер водительского удостоверения.
- Статус аренды (например, «Одобрено»).

2. Отображение стоимости:

- Отображение стоимости аренды автомобиля за весь период.
- Отображение стоимости страховки за весь период.
- Отображение общей суммы бронирования.

Переходы

- **Кнопка «Отменить бронирование» в случае активного бронирования:** Выполнение действия отмены бронирования и переход на экран списка бронирований.
- **Кнопка «Назад»:** Возвращение на предыдущий экран без изменений.

Экран «Избранное»

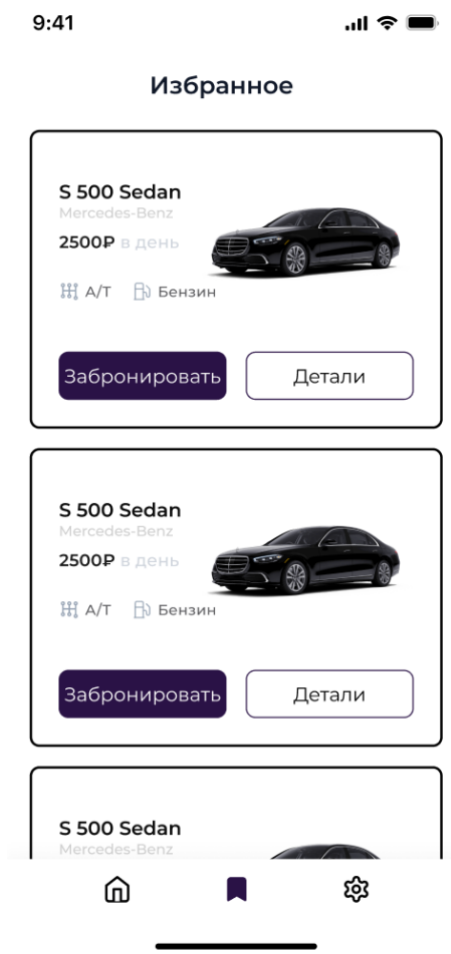


Рисунок 19 – Избранное

Общее описание

Реализуйте экран для отображения избранных автомобилей в приложении согласно макету. Точкой входа в данный экран является нажатие на иконку «Избранное» в нижней навигации.

Функциональные требования

1. Отображение списка избранных автомобилей:

- Каждый автомобиль отображается в виде карточки с информацией о модели, марке, цене и основных характеристиках.
- В каждой карточке доступны две кнопки: «Забронировать» и «Детали».

2. Действия с автомобилем:

- **Забронировать автомобиль:**

- По нажатию на кнопку «Забронировать» происходит переход на экран оформления аренды.
- **Просмотр деталей:**
 - По нажатию на кнопку «Детали» происходит переход на экран с детальной информацией об автомобиле.

Переходы

- **Кнопка «Забронировать»:** Переход на экран оформления аренды.
- **Кнопка «Детали»:** Переход на экран с детальной информацией об автомобиле.

ЛАБОРАТОРНАЯ РАБОТА №4

Экран «Стать арендодателем»

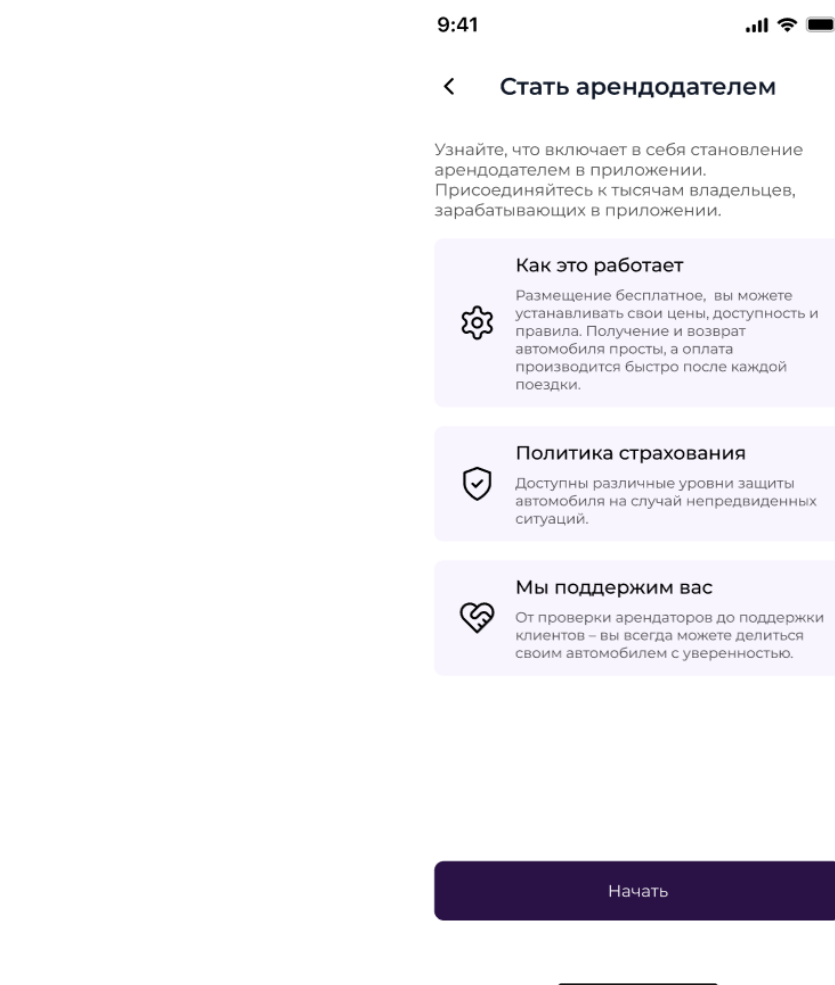


Рисунок 20 – Стать арендодателем

Общее описание

Реализуйте экран согласно макету. Точкой входа в данный экран является нажатие на поле «Подключить свой автомобиль» на экране настроек.

Функциональные требования

1. **Отображение информационных блоков.**
2. **Кнопка «Начать»:**
 - При нажатии на кнопку пользователь переходит на следующий экран, где начинается процесс добавления автомобиля для аренды.

Переходы

- **Кнопка «Начать»:** Переход на экран добавления автомобиля в приложение.
- **Кнопка «Назад»:** Возвращение на экран настроек.

Экран «Добавление автомобиля (шаг 1)»

9:41

< Добавление автомобиля

Где расположен ваш автомобиль?

Авиамоторная ул., 8, стр. 2

Далее

Рисунок 21 – Добавление автомобиля (шаг 1)

Общее описание

Реализуйте экран добавления автомобиля в приложении согласно макету.

Точкой входа в данный экран является нажатие на кнопку «Начать» на экране «Стать арендодателем».

Функциональные требования

1. Поле ввода адреса:

- Пользователь должен ввести адрес, где расположен автомобиль.

2. Кнопка «Далее»:

- **Неактивное состояние:** Кнопка «Далее» должна быть неактивной, пока поле ввода адреса не заполнено.

- **Активное состояние:** После ввода адреса кнопка становится активной, и по нажатию на неё осуществляется переход на следующий шаг добавления автомобиля.

Переходы

- **Кнопка «Далее» (активная):** Переход на следующий экран добавления автомобиля после ввода адреса.
- **Кнопка «Назад»:** Возвращение на предыдущий экран («Стать арендодателем»).

Экран «Добавление автомобиля (шаг 2)»

9:41

< Добавление автомобиля

Год выпуска

Год

Марка автомобиля

Марка

Модель

Модель

Трансмиссия

Пробег

Добавьте описание

Пожалуйста, добавьте дополнительную информацию о вашем автомобиле, которая может быть интересна арендатору. Например, о внутреннем оформлении, мощности и других особенностях вашего автомобиля.

Отправить

Рисунок 22 – Добавление автомобиля (шаг 2)

Общее описание

Реализуйте экран добавления информации об автомобиле в приложении согласно макету. Точкой входа в данный экран является нажатие на кнопку «Далее» на предыдущем экране добавления автомобиля.

Функциональные требования

1. Поля ввода:

- Год выпуска автомобиля.
- Марка автомобиля.
- Модель автомобиля.
- Тип трансмиссии (с выпадающим списком).
- Пробег автомобиля.

- Описание автомобиля (например, информация о внутреннем оформлении, мощности, особенностях).

2. Кнопка «Отправить»:

- **Неактивное состояние:** Кнопка «Отправить» должна быть неактивной, пока все поля не заполнены.
- **Активное состояние:** После заполнения всех полей кнопка становится активной, и по нажатию на неё происходит отправка данных и переход на следующий шаг.

Переходы

- **Кнопка «Отправить» (активная):** После заполнения всех полей и нажатия кнопки данные отправляются, и пользователь переходит на следующий экран процесса добавления автомобиля.
- **Кнопка «Назад»:** Возвращение на предыдущий экран добавления автомобиля (ввода адреса).

Экран «Добавление фотографий автомобиля»



Рисунок 23 – Добавление фотографий автомобиля

Общее описание

Реализуйте экран добавления фотографий автомобиля в приложении согласно макету. Точкой входа в данный экран является нажатие на кнопку «Отправить» на предыдущем экране добавления информации об автомобиле.

Функциональные требования

1. Отображение загруженных фотографий:

- Пользователь может добавить до 5 фотографий автомобиля, которые будут отображаться на этом экране.

2. Кнопка «Далее»:

- Кнопка становится активной только после того, как будут добавлены фотографии.

Переходы

- **Кнопка «Далее»:** Переход на следующий экран процесса добавления автомобиля после добавления фотографий.
- **Кнопка «Назад»:** Возвращение на предыдущий экран добавления информации об автомобиле.

Экран «Успешное добавление автомобиля»

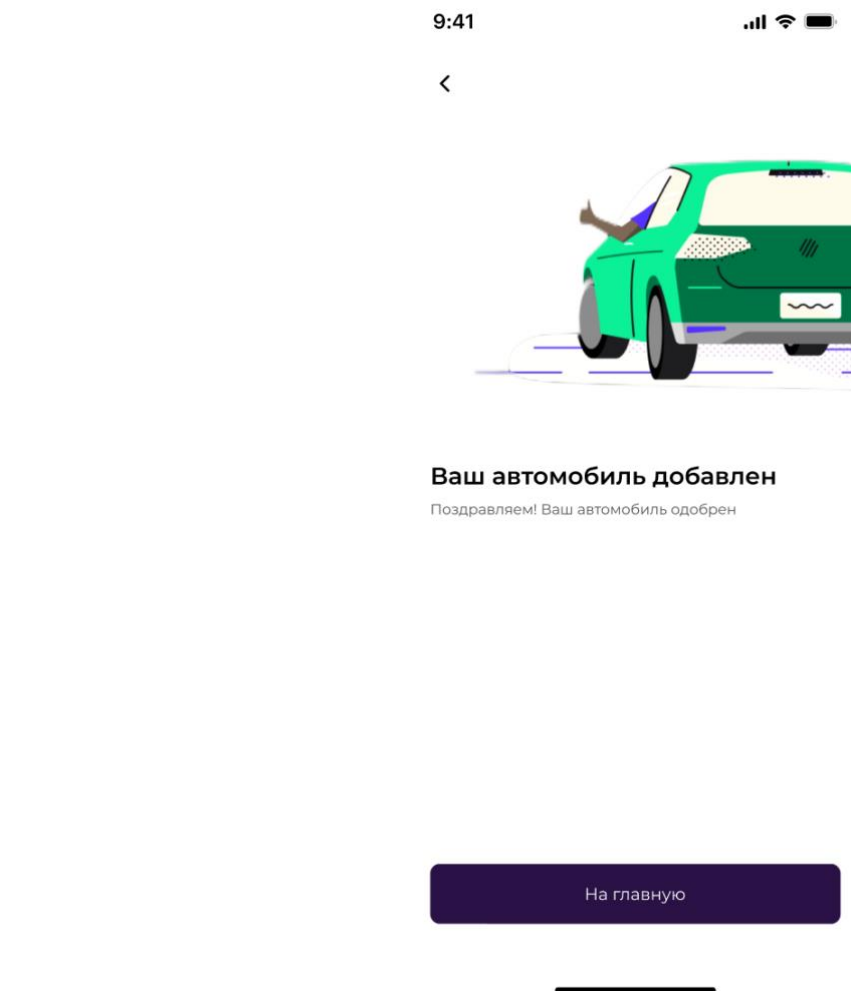


Рисунок 24 – Успешное добавление автомобиля

Общее описание

Реализуйте экран подтверждения добавления автомобиля в приложении согласно макету. Точкой входа в данный экран является успешное завершение процесса добавления автомобиля.

Функциональные требования

1. Сообщение об успешном добавлении:

- Отображение сообщения, подтверждающего, что автомобиль был успешно добавлен и одобрен.

2. Кнопка «На главную»:

- По нажатию на кнопку пользователь возвращается на главный экран приложения.

Переходы

- **Кнопка «На главную»:** Переход на главный экран приложения.
- **Кнопка «Назад»:** Возвращение на предыдущий экран (если такое поведение предусмотрено).