**Министерство науки и высшего образования Российской Федерации**

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»**

(ВлГУ)

Кафедра ВТиСУ

КУРСОВАЯ РАБОТА

по дисциплине «Разработка мобильных приложений»

на тему: «Разработка мобильного приложения на React Native. Часть 5. Разработка мобильного приложения для туристического агентства»

Студент: Андронова Е.С.

Институт: ИИТЭ

Направление: 09.03.03

Группа: ПИ-121

№ зачетной книжки: 121001153

Руководитель: Шутов А. В.

Владимир 2024 г.

**Оглавление**

[Введение 3](#_gjdgxs)

[Глава 1. Анализ предметной области и современные технологии мобильной коммерции 5](#_1fob9te)

[1.1. Современные тенденции мобильной коммерции 5](#_3znysh7)

[1.2. Обзор существующих решений 7](#_2et92p0)

[1.3. Выводы по анализу 9](#_tyjcwt)

[Глава 2. Разработка мобильного приложения для туристического агентства 11](#_3dy6vkm)

[2.1. Функциональные и нефункциональные требования 11](#_1t3h5sf)

[2.2. Архитектура приложения 12](#_4d34og8)

[2.3 Реализация функционала 18](#_2s8eyo1)

[2.3.1. Фильтрация и поиск туров 18](#_17dp8vu)

[2.3.2. Просмотр каталога туров 25](#_3rdcrjn)

[2.3.3. Бронирование туров 30](#_26in1rg)

[Заключение 41](#_35nkun2)

[Список литературы 42](#_1ksv4uv)

# **Введение**

Современный мир стремительно движется в сторону цифровизации, и мобильные технологии занимают ключевую роль в этом процессе. Мобильные приложения стали незаменимым инструментом как для пользователей, так и для бизнеса, предлагая доступ к услугам и продуктам в любое время и из любого места. В туристической отрасли мобильные приложения занимают особое место, формируя новое направление – цифровой туризм. Это направление активно развивается благодаря доступности смартфонов, быстрому интернету и удобству использования мобильных платформ.

На фоне глобального интереса к путешествиям и отдыху, разработка мобильного приложения для туристического агентства становится актуальной и востребованной. Возможность подбора, фильтрации и бронирования туров через смартфон упрощает процесс планирования поездки, повышает лояльность клиентов и улучшает взаимодействие с пользователем. Удобство поиска, персонализация предложений и прозрачность оформления бронирований являются ключевыми факторами успешной работы любого туристического сервиса.

Цель работы

Целью данной курсовой работы является разработка удобного и многофункционального мобильного приложения для туристического агентства на платформе React Native, которое включает следующие возможности:

• Просмотр каталога туров: пользователи могут ознакомиться с доступными турами и горячими предложениями, сопровождаемыми подробным описанием и изображениями.

• Бронирование туров: пользователи могут выбрать тур, указать предпочтительный отель, место отдыха и даты, а также оформить бронирование.

• Фильтрация и поиск туров: предоставление пользователям возможности подбора туров по различным параметрам, таким как страна, город, цена, даты, тип отдыха и рейтинг отелей.

Задачи работы

Для достижения поставленной цели необходимо выполнить следующие задачи:

1. Провести анализ предметной области, изучить существующие решения и определить потребности целевой аудитории.

2. Разработать удобный и интуитивно понятный интерфейс для пользователей, который позволит легко искать и бронировать туры.

3. Реализовать функционал для фильтрации и поиска туров по ключевым параметрам.

4. Разработать модуль бронирования, включающий выбор отеля, указание дат и оформление заявки на тур.

5. Провести тестирование приложения для проверки его функциональности, удобства использования и стабильности работы.

Практическая значимость

Практическая значимость проекта заключается в создании программного продукта, который обеспечивает:

• Удобство поиска и бронирования туров для конечного пользователя.

• Повышение вовлеченности клиентов благодаря персонализации и адаптивному интерфейсу.

• Эффективное продвижение туристических услуг и повышение конкурентоспособности агентства.

# **Глава 1. Анализ предметной области и современные технологии мобильной коммерции**

## **1.1. Современные тенденции мобильной коммерции**

Мобильная коммерция (m-commerce) представляет собой современное направление электронной коммерции, ориентированное на использование мобильных устройств для покупок. В последние годы m-commerce стал неотъемлемой частью повседневной жизни многих пользователей благодаря быстрому распространению смартфонов, доступности высокоскоростного интернета и удобству, которое предлагают мобильные приложения. По статистике, более 60% всех онлайн-покупок сейчас осуществляется через мобильные устройства, что подчеркивает значимость этой области для бизнеса и пользователей.

Ключевые Тенденции в M-Commerce

1. Рост популярности мобильных платежей

Одной из ключевых тенденций мобильной коммерции является растущая популярность мобильных платежных систем. Такие технологии, как Apple Pay, Google Pay и оплата с использованием QR-кодов, делают процесс оплаты более простым и безопасным. Пользователи могут быстро совершать платежи с помощью одного касания или сканирования, что ускоряет покупки и обеспечивает высокий уровень защиты данных. Это повышает доверие потребителей к мобильной коммерции, так как они ощущают безопасность и удобство использования таких технологий.

1. Персонализация пользовательского опыта

Современные мобильные приложения активно внедряют персонализацию, используя данные о поведении пользователей для формирования рекомендаций и предложений, максимально соответствующих их индивидуальным предпочтениям. Такой подход включает анализ истории покупок, интересов в определенных категориях товаров и даже геолокации. Персонализированные предложения позволяют не только удерживать внимание клиентов, но и увеличивать конверсию продаж, поскольку пользователи получают актуальные для них и потенциально интересные товары или услуги.

1. Интеграция с социальными сетями

В наше время всё больше пользователей отдают предпочтение покупкам через социальные сети, такие как Instagram и TikTok. Эти платформы позволяют приобретать товары прямо во время просмотра контента, что делает процесс быстрым и удобным. Интеграция инструментов для покупок в социальных сетях помогает брендам расширять охват аудитории и выстраивать более тесное взаимодействие с клиентами. Такие функции, как «шоппинг-экран» в Instagram, делают покупки более естественными и увлекательными для пользователей.

1. Мобильные уведомления

Push-уведомления представляют собой эффективный инструмент для оперативного информирования пользователей о новых продуктах, акциях, скидках и актуальных новостях. Благодаря персонализации и сегментации такие уведомления становятся более результативными. Получая релевантную информацию, пользователи чаще принимают решение о покупке, что способствует повышению конверсии и укреплению их лояльности к бренду.

1. Применение M-Commerce в бизнесе

Мобильные приложения, особенно в сфере техники, включая персональные компьютеры, предоставляют бизнесу широкий спектр возможностей. Они способствуют повышению лояльности клиентов за счет уникальных продуктов и услуг, а также позволяют оптимизировать внутренние процессы. Компании могут проводить эффективные маркетинговые кампании, анализируя поведение пользователей в реальном времени и адаптируя свои предложения к актуальным тенденциям. Например, через мобильные приложения можно реализовывать специальные акции, предлагать эксклюзивные товары и организовывать мероприятия для постоянных клиентов.

## **1.2. Обзор существующих решений**

Обзор мобильных приложений для туристических агентств в России включает наиболее популярные платформы, которые предоставляют пользователям широкий спектр возможностей для поиска и бронирования туров. Рассмотрим несколько ключевых приложений на российском рынке.

Tutu.ru

Описание:

Tutu.ru — одна из ведущих российских платформ для поиска и бронирования туристических услуг.

Ключевые функции:

• Широкий выбор туров: Пользователи могут найти пакетные туры, отели и билеты от различных поставщиков.

• Фильтрация по параметрам: Удобный поиск по странам, датам, бюджету и типу отдыха.

• Отслеживание бронирования: Возможность следить за статусом заявки на тур в реальном времени.

• Скидки и акции: Раздел с актуальными предложениями для экономных путешественников.

Level.Travel

Описание:

Level.Travel — агрегатор туров, предоставляющий пользователям возможность сравнения предложений от ведущих туроператоров.

Ключевые функции:

• Доступ к каталогу туров: Удобный интерфейс для просмотра и сравнения туров, включая предложения по горячим путевкам.

• Персонализированные рекомендации: Подбор туров на основе предпочтений пользователя.

• Проверка наличия: Информация о доступных турах с указанием дат и отелей.

• Гибкая оплата: Различные способы оплаты, включая рассрочку и оплату частями.

Onlinetours

Описание:

Onlinetours специализируется на поиске туров и пакетных предложений для отдыха в России и за рубежом.

Ключевые функции:

• Сравнение туров: Возможность сравнивать предложения от разных туроператоров.

• Отзывы и рейтинги: Система пользовательских оценок отелей и туров для удобства выбора.

• Специальные предложения: Горящие туры и акции для мобильных пользователей.

• Поддержка клиентов: 24/7 онлайн-консультанты для помощи с выбором и бронированием.

Aviasales

Описание:

Aviasales — популярное приложение для поиска авиабилетов, которое также предоставляет услуги по бронированию туров.

Ключевые функции:

• Комплексное бронирование: Возможность поиска туров с включенными авиабилетами и проживанием.

• Интерактивный интерфейс: Удобный просмотр информации о турах и отелях.

• История бронирований: Просмотр ранее забронированных туров для повторных поездок.

• Горящие туры: Отдельный раздел с акциями и скидками.

Travelata

Описание:

Travelata — приложение для поиска и бронирования туров с простым и функциональным интерфейсом.

Ключевые функции:

• Гибкая фильтрация: Подбор туров по странам, стоимости, типу отдыха и рейтингу отелей.

• Отображение деталей: Подробные описания и фотографии отелей, маршрутов и экскурсий.

• Онлайн-оплата: Удобная система оплаты, включая карты и электронные кошельки.

• Лояльность: Программа бонусов для постоянных клиентов.

Каждое из перечисленных приложений обладает уникальными функциями, которые делают процесс поиска и бронирования туров удобным и привлекательным для пользователей.

## **1.3. Выводы по анализу**

Проведенный анализ показал, что для создания конкурентоспособного мобильного приложения для туристического агентства необходимо учитывать следующие аспекты:

• Удобство интерфейса: Приложение должно быть интуитивным, с простым и понятным дизайном, чтобы пользователи могли легко находить и бронировать туры.

• Быстродействие: Высокая скорость загрузки страниц и обработки запросов критически важна для удержания пользователей и повышения их удовлетворенности.

• Функциональность: Важно обеспечить удобный поиск и фильтры туров, включая параметры по странам, датам, бюджету и типу отдыха, а также реализовать систему авторизации и бронирования.

• Уникальные возможности: Приложение должно предлагать функции, которых нет в конкурирующих продуктах, такие как персонализированные рекомендации, гибкие условия оплаты и эксклюзивные предложения.

Особое внимание следует уделить созданию административной панели, которая позволит эффективно управлять ассортиментом туров, акциями и бронированиями, обеспечивая удобство как для пользователей, так и для туристического агентства.

# **Глава 2. Разработка мобильного приложения для туристического агентства**

## **2.1. Функциональные и нефункциональные требования**

Для успешной разработки мобильного приложения для туристического агентства важно определить функциональные и нефункциональные требования. Эти требования задают основные ориентиры для разработки и помогают создать продукт, соответствующий ожиданиям пользователей.

**Функциональные требования:**

Авторизация и регистрация пользователя:

• Возможность входа в приложение с использованием логина и пароля.

• Регистрация новых пользователей с обязательной валидацией данных.

Просмотр каталога туров:

• Отображение списка доступных туров с фотографиями, названием, ценой, описанием и рейтингом.

• Возможность просматривать подробную информацию о каждом туре.

Бронирование:

• Выбор тура, предпочтительного отеля, места отдыха и дат.

• Оформление бронирования с возможностью подтверждения и отмены заявки.

Фильтрация и поиск туров:

• Удобная фильтрация по параметрам, таким как страна, город, тип отдыха, цена и рейтинг отеля.

• Интерактивный поиск по ключевым словам.

Административная панель:

• Управление каталогом туров: добавление, удаление и редактирование информации.

• Просмотр и обработка заявок на бронирование.

**Нефункциональные требования:**

1. Интуитивно понятный интерфейс: Приложение должно быть удобным и простым в использовании, даже для пользователей с минимальным опытом работы с мобильными приложениями.
2. Кроссплатформенность: Приложение должно быть разработано для работы как на Android, так и на iOS, чтобы охватить максимальную аудиторию.
3. Высокая производительность: Быстрая загрузка страниц, отзывчивость интерфейса и стабильная работа при высокой нагрузке.
4. Безопасность данных: Гарантия сохранности персональной информации и защита финансовых операций.

## **2.2. Архитектура приложения**

Проектирование архитектуры мобильного приложения является ключевым этапом, который влияет на стабильность, производительность и масштабируемость системы.

Для данного проекта приложение разрабатывается на React Native, что обеспечивает кроссплатформенность и упрощает поддержку приложения. Взаимодействие с базой данных будет организовано через REST API. Основные компоненты архитектуры включают:

Технологии:

• Реализована на React Native, что обеспечивает кроссплатформенность (поддержка iOS и Android).

• Для выполнения HTTP-запросов к серверу используется библиотека Axios.

Функционал:

• Отправка запросов на сервер для получения данных о турах, их фильтрации и управления бронированиями.

• Реализация пользовательского интерфейса, включая просмотр каталога туров, оформление бронирований и управление профилем пользователя.

Серверная часть (Node.js и CMS Strapi):

• Сервер построен на платформе Node.js, что обеспечивает высокую производительность и надёжность.

• CMS Strapi используется для создания и управления контентом, а также для автоматической генерации REST API.

• Strapi позволяет легко настраивать структуры данных и обеспечивает интуитивно понятный интерфейс для администраторов.

REST API:

• Генерируется автоматически CMS Strapi для всех сущностей, таких как туры, бронирования, категории и пользователи.

• API поддерживает основные методы HTTP (GET, POST, PUT, DELETE), обеспечивая полный доступ к функционалу приложения.

База данных(SQLite):

• В качестве базы данных используется SQLite, которая интегрирована с CMS Strapi.

• SQLite хранит все данные приложения, включая:

• Информацию о турах (название, описание, фотографии, цены, доступные даты).

• Пользовательские данные (учётные записи, сохранённые бронирования).

• Административные данные (категории туров, фильтры).

• Лёгкость SQLite делает её идеальным выбором для приложений с умеренным объемом данных.

4. Взаимодействие клиента с сервером:

• Клиентская часть обращается к REST API, сгенерированному Strapi, через библиотеку Axios.

Эта архитектура обеспечивает надежность, масштабируемость и удобство использования, что соответствует поставленным функциональным и нефункциональным требованиям.

1. Общая архитектура приложения

Архитектура мобильного приложения для туристического агентства разработана с использованием современных технологий, которые обеспечивают высокую производительность, удобство разработки и масштабируемость. Она включает серверную часть, построенную на Node.js с использованием CMS Strapi, и клиентскую часть, взаимодействующую с сервером через REST API.

**Клиентская сторона (мобильное приложение):**

Мобильное приложение на React Native предоставляет интуитивно понятный интерфейс и удобное взаимодействие с пользователем. Его архитектура включает несколько ключевых компонентов:

1. Пользовательский интерфейс (UI):

• Состоит из различных экранов и компонентов (например, экран каталога туров, бронирования, фильтров).

• Обеспечивает удобное взаимодействие пользователя с приложением.

2. Модель (Model):

• Обрабатывает данные приложения, включая объекты, представляющие туры, пользователей и бронирования.

• Отвечает за управление данными, полученными от сервера.

3. Контроллер (Controller):

• Компоненты React Native (например, функциональные компоненты или хуки) управляют логикой взаимодействия между UI и моделью.

• Реализуют обработку пользовательских действий, таких как выбор тура или отправка заявки на бронирование.

4. Сетевой слой:

• Для взаимодействия с сервером используется библиотека Axios.

• Сетевой слой обрабатывает асинхронные HTTP-запросы, обеспечивая получение данных с сервера без блокировки интерфейса.

• Поддерживает методы GET (для получения данных) и POST (для отправки данных).

**Серверная сторона (Node.js с CMS Strapi):**

1. CMS Strapi:

• Используется для управления контентом приложения, включая туры, категории, пользователей и бронирования.

• Автоматически генерирует REST API, который позволяет клиенту взаимодействовать с сервером.

2. REST API:

• Поддерживает CRUD-операции:

• GET: Получение списка туров, деталей тура или информации о пользователе.

• POST: Оформление бронирования или регистрация нового пользователя.

• PUT: Обновление информации о туре или бронировании.

• DELETE: Удаление бронирования.

• Возвращает данные в формате JSON, обеспечивая удобное взаимодействие с клиентом.

3. База данных (SQLite):

• Хранит данные о турах, пользователях и бронированиях.

• SQLite интегрирована с CMS Strapi для автоматического управления данными.

• Легкая и производительная, что делает её подходящей для небольших и средних приложений.

1. Взаимодействие компонентов

1. UI:

• Пользователь взаимодействует с интерфейсом приложения, выполняя действия, такие как выбор фильтров или оформление бронирования.

• Приложение формирует HTTP-запрос, который отправляется на сервер.

2. HTTP-запрос:

• Используя библиотеку Axios, приложение отправляет запросы на сервер.

• Запросы могут быть:

• GET: Получение данных (например, списка доступных туров).

• POST: Отправка данных (например, информация о бронировании).

1. Обработка сервером:

• Сервер на Node.js с помощью CMS Strapi обрабатывает запросы и взаимодействует с базой данных SQLite.

• Выполняются операции выборки, вставки или обновления данных.

1. База данных (SQLite):

База данных, используемая в приложении, реализована на SQLite, что обеспечивает компактное и производительное хранение данных. Однако SQLite не имеет графической оболочки для управления данными, и все операции, такие как создание таблиц, выполнение запросов и управление записями, выполняются через команды SQL. Это требует использования соответствующих инструментов для доступа и работы с базой данных, например, через программный интерфейс или консольные утилиты. Несмотря на отсутствие визуального интерфейса, SQLite остаётся эффективным решением для приложений, благодаря своей лёгкости и интеграции с серверной и клиентской частями.

• Сервер обращается к SQLite для выполнения операций.

• Например, сохранение информации о новом бронировании или обновление данных о доступных турах.

1. Ответ на клиент:

• Сервер формирует ответ в формате JSON и отправляет его обратно на клиент.

• Пример: список туров или подтверждение успешного бронирования.

1. Обновление интерфейса:

• Приложение разбирает полученный JSON-ответ и обновляет UI, чтобы отобразить актуальную информацию пользователю.

**Преимущества архитектуры:**

1. Модульность и масштабируемость:

• Использование CMS Strapi упрощает управление контентом и API.

2. Удобство разработки:

• React Native и SQLite обеспечивают лёгкость интеграции и высокую производительность.

3. Гибкость:

• REST API позволяет адаптировать функционал под новые требования.

4. Кроссплатформенность:

• React Native позволяет использовать приложение как на Android, так и на iOS.

База данных не имеет графиической оболочки

Эта архитектура обеспечивает надежность, высокую производительность и удобство как для пользователей, так и для администраторов туристического агентства.

## **2.3 Реализация функционала**

### **2.3.1. Фильтрация и поиск туров**

Для удобства работы с приложением процесс регистрации отсутствует. При запуске приложения пользователь сразу попадает на экран поиска и фильтрации туров. Тем не менее, для обеспечения безопасного доступа к персональным данным при оформлении бронирований предусмотрена система авторизации.

Основные этапы работы функционала авторизации:

1. Инициализация интерфейса:

• При открытии приложения загружается экран поиска туров с доступными фильтрами (страна, отель, дата, цена).

• В случае необходимости авторизации (например, при попытке забронировать тур) отображается экран входа, где пользователь вводит логин и пароль.

2. Валидация данных:

• При нажатии кнопки “Вход” выполняется проверка заполненности всех обязательных полей.

• Если одно или несколько полей не заполнены, отображается уведомление с просьбой ввести данные.

• В случае некорректных данных (например, неправильный пароль) приложение показывает сообщение об ошибке.

3. Обработка результатов:

• Если авторизация прошла успешно, пользователь получает доступ к полному функционалу приложения, включая бронирование туров и просмотр сохраненных данных.

• При ошибке авторизации пользователь видит уведомление с рекомендацией (например, проверить правильность логина и пароля).

4. Работа с сетевыми запросами:

• Для отправки запросов и обработки ответов используется библиотека Axios, обеспечивающая эффективную работу с REST API.

• Клиент анализирует код ответа и обновляет интерфейс в соответствии с результатом.

5. Исключение необходимости регистрации:

• Возможность регистрации новых пользователей может быть добавлена на отдельный экран, где пользователь указывает имя, email, телефон и пароль.

• Основной упор сделан на простоту взаимодействия, чтобы пользователь мог начать работу с приложением без регистрации.

Ниже представлен интерфейс фильтрации и поиска туров

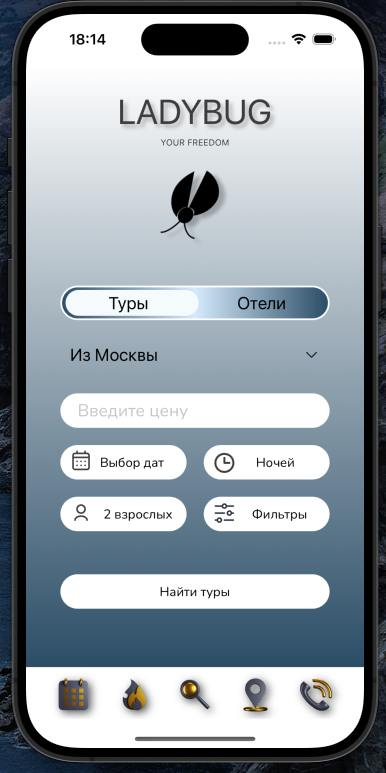
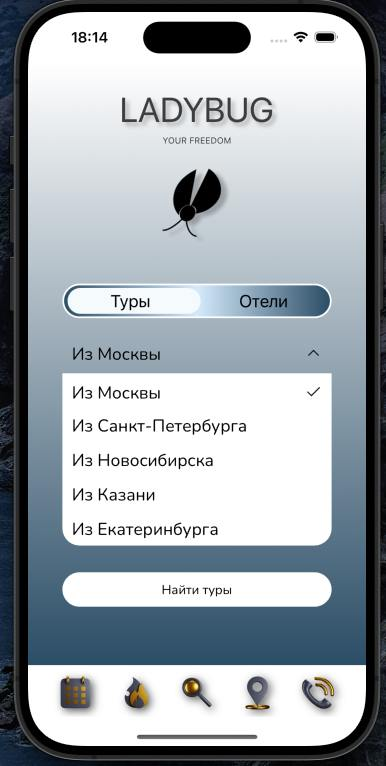
 

Рис.1, 2 – Интерфейс фильтрации, поиска туров и отелей

Интерфейс позволяет пользователям быстро находить подходящие туры, задавая параметры поиска.

Основные элементы:

1. Поле поиска: Поиск по стоимости тура.

• Направление: Отправление из какого города

• Выбор: Даты тура, продолжительность отдыха, количество человек.

3. Кнопка “Найти туры”: Поиск туров из уже добавленных.

4. Результаты поиска: Список туров с краткой информацией (название, фото, цена).

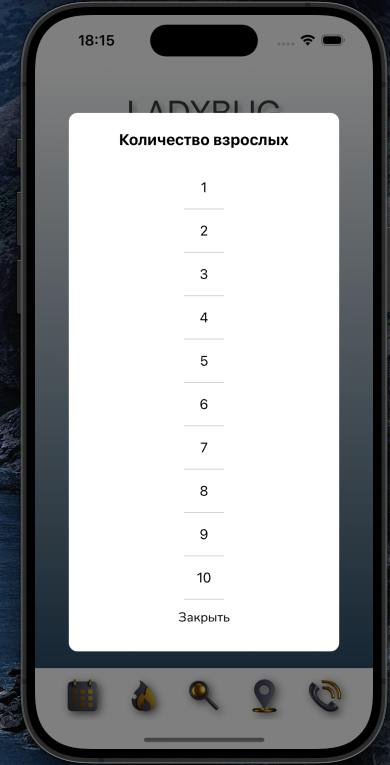
 

Рис.3, 4 – Выбор даты тура и количества человек

Функции:

• Быстрый поиск по ключевым словам.

• Настройка фильтров для точного подбора туров.

• Интерактивность: обновление результатов при изменении параметров.

• Сохранение предпочтений и сортировка результатов.

Этот интерфейс обеспечивает удобный и быстрый выбор туров, соответствующих запросам пользователей.

Ниже приведен пример кода:

import React, { useState } from 'react'

import { Alert, StyleSheet, Text, TouchableOpacity, View } from 'react-native'

import Header from './Header'

import { TourType } from '@/app/apiController/getTours';

import Rate from '../InfoBlocks/Rate';

import TextField from '../Inputs/TextField';

import { LinearGradient } from 'expo-linear-gradient';

import SimpleText from '../Inputs/SimpleText';

import { ReservationRouteProp } from '@/app/navigation/\_Tours/TourStackParams';

import { Bid, createBid } from '@/app/apiController/createBid';

const Reservation: React.FC<{ route: ReservationRouteProp }> = ({ route }) => {

const params = route.params as TourType;

const [name, setName] = useState('');

const [number, setNumber] = useState('');

const [email, setEmail] = useState('');

const [parentWidth, setParentWidth] = useState<number>(0);

const handleLayout = (event: any) => {

const { width } = event.nativeEvent.layout;

setParentWidth(width);

};

const handleCreateBid = () => {

const bid: Bid = {

name: name,

phone: number,

email: email,

TourName: params.title,

comment: '',

code: ''

}

createBid(bid).then(result => {

if (result != null) {

Alert.alert('Успех', 'Заявка создана');

} else {

Alert.alert('Ошибка', 'Заявка не отправлена');

}

});

}

return (

<LinearGradient

colors={['#FFFFFF', '#2B4D66']}

style={styles.container}

>

<Header title={'бронирование'} />

<View style={styles.content}>

<Text style={styles.titleText}>{params.title}</Text>

<View style={{paddingVertical: 15}}>

<Rate rate={params.rate} />

</View>

<View style={{paddingVertical: 30, marginTop: 20, gap: 25}}>

<Text style={{fontFamily: 'Nunito', textAlign: 'center', fontSize: 20, fontWeight: 400}}>Запрос информации по туру</Text>

<TextField value={name} setValue={setName} placeholder='Фамилия, имя' />

<TextField value={number} setValue={setNumber} placeholder='Мобильный телефон' />

<TextField value={email} setValue={setEmail} placeholder='Email' />

<View

onLayout={handleLayout}

style={{ flexDirection: 'row', gap: 20, justifyContent: 'center' }}

>

<SimpleText title='Комментарий' parentWidth={parentWidth} gap={20} icon='add-comment' />

<SimpleText title='Промокод' parentWidth={parentWidth} gap={20} icon='message-check' />

</View>

<TouchableOpacity style={styles.button} onPress={handleCreateBid}>

<Text style={styles.buttonText}>Оставить заявку</Text>

</TouchableOpacity>

</View>

</View>

</LinearGradient>

)

}

const styles = StyleSheet.create({

container: {

flex: 1,

backgroundColor: '#fff'

},

content: {

padding: 20

},

titleText: {

fontSize: 20,

fontWeight: 700,

fontFamily: 'Nunito',

},

button: {

marginTop: 20,

backgroundColor: '#2B4D66',

borderRadius: 30,

padding: 10,

},

buttonText: {

color: '#fff',

fontFamily: 'Nunito',

fontSize: 15,

textAlign: 'center'

},

});

export default Reservation

### **2.3.2. Просмотр каталога туров**

Каталог туров – это ключевая часть мобильного приложения туристического агентства, предоставляющая пользователям возможность изучать доступные туры и подбирать оптимальные варианты для отдыха. Каталог включает подробное описание каждого тура и удобные инструменты для его фильтрации и поиска.

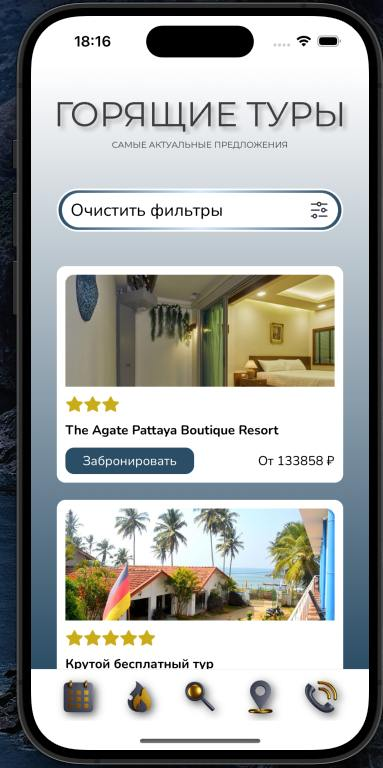
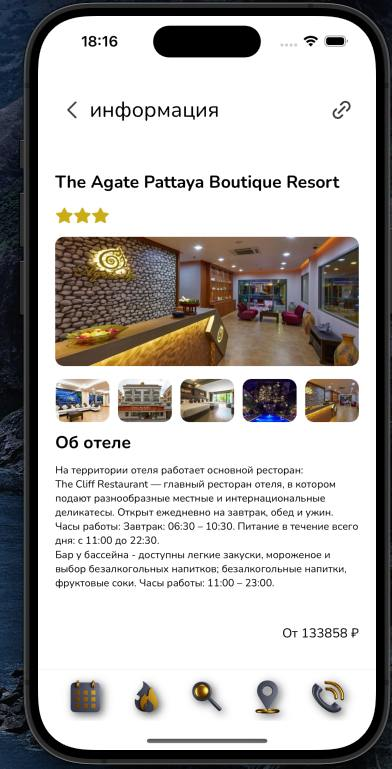
 

Рис.5, 6 – Интерфейс каталога туров и информации о них

Основные функции каталога туров:

1. Отображение списка туров:

• Пользователи видят список доступных туров с краткой информацией:

• Название тура (например, “Тур в Турцию – всё включено”).

• Основное изображение (фото отеля, достопримечательности или пляжа).

• Цена за тур (например, “от 50 000 ₽”).

• Дата начала тура.

2. Детальная информация о туре:

• При выборе тура открывается карточка с полным описанием:

• Страна и город назначения.

• Продолжительность тура (например, “7 ночей, 8 дней”).

• Условия проживания (отель, тип номера, питание).

• Доступные даты начала тура.

• Услуги, включенные в тур (перелёт, трансфер, экскурсии).

• Описание достопримечательностей, которые можно посетить.

• Рейтинг и отзывы предыдущих путешественников.

3. Фильтрация туров:

• Пользователь может задать параметры поиска:

• Страна или город.

• Диапазон цен.

• Тип отдыха (пляжный, экскурсионный, активный).

• Рейтинг отеля.

• Дата отправления.

4. Поиск по ключевым словам:

• Возможность быстро найти тур по названию или ключевым словам (например, “Турция 2024”, “горящие туры”).

5. Отображение горячих предложений:

• Отдельный раздел каталога, где представлены туры с акциями и скидками.

6. Добавление в избранное:

• Пользователь может сохранять интересные туры в список избранных для быстрого доступа в будущем.

Техническая реализация:

• Данные: Каталог загружается с сервера через REST API.

• Кэширование: Для ускорения работы приложения туры сохраняются локально (например, с использованием SQLite или Room).

• Обновление: Каталог автоматически обновляется при запуске приложения или при обновлении фильтров.

Ниже приведен пример кода получения туров:

import axios from "axios";

// Типы для изображений

interface PictureFormat {

url: string;

}

interface Picture {

formats?: {

thumbnail?: PictureFormat;

large?: PictureFormat;

};

url?: string;

}

// Типы для тура

export interface TourType {

title: string;

description: string;

cost: number;

rate: number;

startDate: Date;

endDate: Date;

images: { thumbnail: string; large: string }[];

}

// Функция получения туров с API

export const getTours = async (): Promise<TourType[] | null> => {

try {

const response = await axios.get<{ data: Array<any> }>(

"http://localhost:1337/api/tours?populate=\*"

);

const data = response.data.data;

if (!data || data.length === 0) {

console.warn("Ответ пустой");

return [];

}

// Форматируем туры

return data.map(post => {

const { title, description, cost, rate, pictures, startDate, endDate } = post;

const images = (pictures || []).map((pic: Picture) => ({

thumbnail: http://localhost:1337${pic.formats?.thumbnail?.url || ""},

large: http://localhost:1337${pic.formats?.large?.url || pic.url || ""},

}));

return { title, description, cost, rate, images, startDate, endDate };

});

} catch (error: any) {

console.error("Ошибка при загрузке туров:", error.message);

return null;

}

};

### **2.3.3. Бронирование туров**

Функционал бронирования туров – одна из ключевых возможностей мобильного приложения туристического агентства, предоставляющая пользователям удобный способ оформления заявки на выбранный тур. Этот процесс интуитивно понятен и максимально автоматизирован, чтобы минимизировать усилия пользователя.

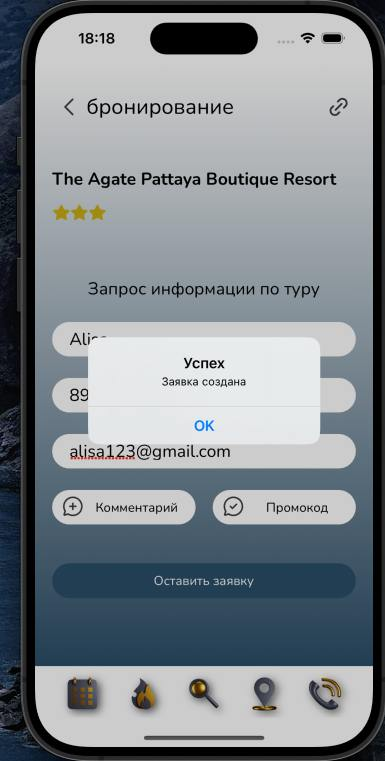
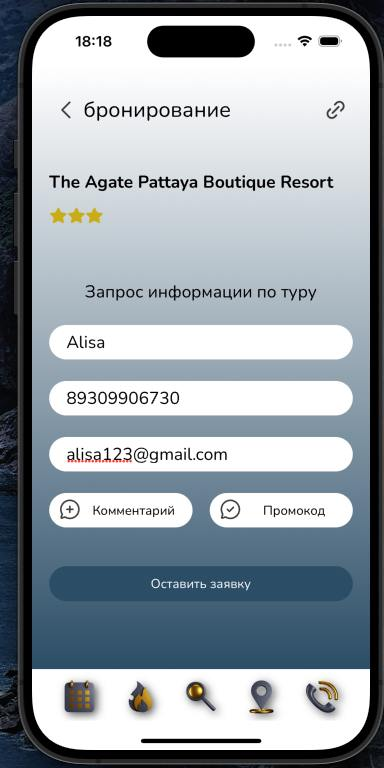


Рис.7, 8 – Интерфейс бронирования и заполнение формы

Основные функции бронирования:

1. Выбор тура:

• Пользователь выбирает интересующий тур из каталога.

• На странице выбранного тура отображается подробная информация, включая доступные даты, условия проживания и стоимость.

2. Заполнение данных:

• В форме бронирования пользователь указывает:

• Личные данные (имя, фамилия, номер телефона, email).

• Может добавить комментарии или промокод на скидку

3. Подтверждение бронирования:

• После заполнения всех данных, для подтверждения бронирования отображается кнопка “Оставить заявку”.

4. Получение подтверждения:

• После успешного оформления пользователь администратор получает заявку на тур:

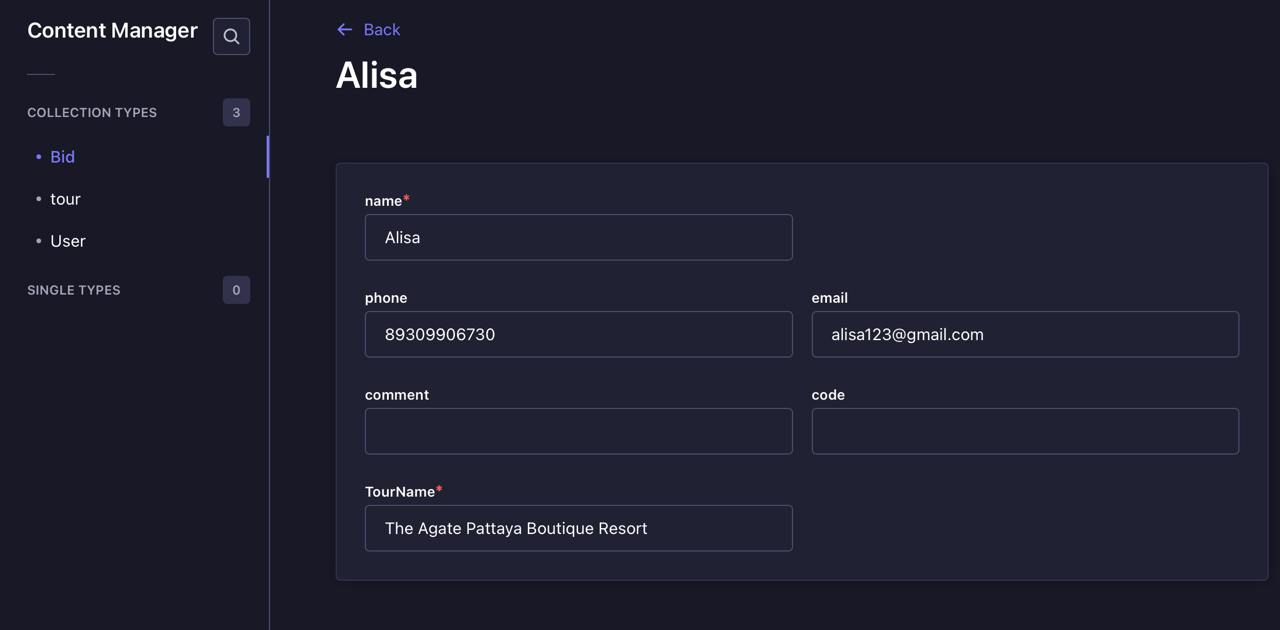


Рис.9 – Заявка на бронирование тура

• Далее администратор свяжется с клиентом для подтверждения бронирования и оплаты.

Ниже приведен пример кода создания заявки:

import axios from 'axios';

export interface Bid {

name: string;

phone: string;

email: string;

comment: string;

code: string;

TourName: string;

}

export async function createBid(bid: Bid): Promise<void | null> {

try {

const response = await axios.post(

'http:/localhost:1337/api/bids',

{

data: bid,

}

);

return response.data;

} catch (error) {

return null;

}

}

Функционал бронирования делает приложение удобным инструментом для планирования путешествий, упрощая процесс выбора и оформления туров.

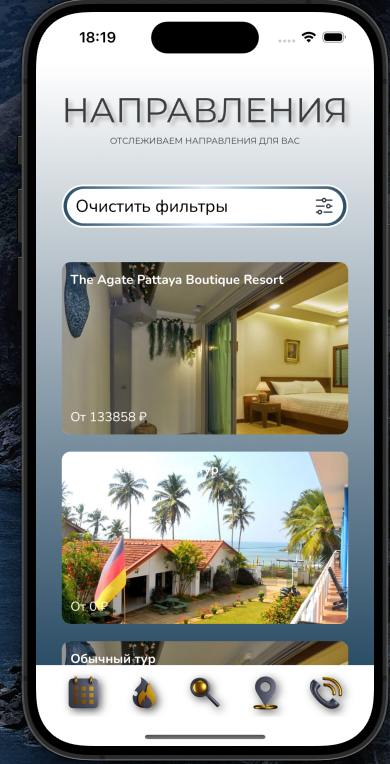


Рис.10, 11 – Интерфейс раздела акций и направлений

Раздел “Акции”

Раздел “Акции” в мобильном приложении туристического агентства предназначен для отображения специальных предложений и скидок, которые привлекают внимание пользователей и стимулируют их к бронированию туров. Этот раздел помогает агентству выделить выгодные предложения и повысить интерес к определённым направлениям.

Основные функции раздела “Акции”:

1. Отображение текущих акций:

• Пользователь видит список туров со скидками или другими выгодными предложениями.

• Для каждого предложения отображаются:

• Название тура.

• Размер скидки (например, “Скидка 30%”).

• Условие акции (например, “Бронирование до 31 декабря”).

• Цена со скидкой и оригинальная цена.

2. Фильтрация и сортировка акций:

• Возможность сортировать предложения по размеру скидки, популярности или сроку действия.

• Фильтрация по типу акций, например:

• “Раннее бронирование”.

• “Горящие туры”.

• “Семейные туры”.

3. Детализация предложений:

• При выборе акции пользователь попадает на карточку тура с подробным описанием:

• Условия действия (даты, количество участников).

• Услуги, включённые в тур (перелёт, проживание, экскурсии).

• Отзывы и рейтинги.

4. Интеграция с бронированием:

• Кнопка “Забронировать сейчас” для оформления тура прямо из раздела “Акции”.

• Информация о наличии мест в рамках акционного предложения.

5. Уведомления об акциях:

• Push-уведомления о новых скидках или приближении окончания акции для зарегистрированных пользователей.

Раздел “Направления”

Раздел “Направления” предоставляет пользователю возможность изучить основные туристические локации, предлагаемые агентством. Этот раздел помогает организовать информацию о странах, городах и курортах, делая выбор путешествия удобным и понятным.

Основные функции раздела “Направления”:

1. Каталог направлений:

• Отображается список стран или регионов с их популярными курортами.

• Каждый элемент включает:

• Название страны/города.

• Иконку или изображение, ассоциирующееся с направлением.

• Краткое описание (например, “Турция: пляжный отдых и древние достопримечательности”).

2. Фильтрация и поиск:

• Возможность фильтровать направления по типу отдыха:

• Пляжный.

• Экскурсионный.

• Активный (горные лыжи, треккинг).

• Поиск направлений по ключевым словам (например, “Испания”, “горнолыжный курорт”).

3. Подробное описание направления:

• При выборе направления пользователь видит развёрнутое описание, включая:

• Особенности региона (климат, основные достопримечательности).

• Лучшие сезоны для посещения.

• Популярные туры в этом направлении с их кратким описанием и ценой.

4. Рекомендации и популярные туры:

• Раздел “Популярные туры” на странице направления предлагает готовые варианты путешествий в выбранный регион.

5. Интеграция с картой:

• Возможность открыть карту и посмотреть местоположение направления, расстояние до него и ближайшие интересные локации.

6. Информирование о визовых требованиях:

• Краткая информация о необходимости визы и условиях её получения.

Преимущества разделов “Акции” и “Направления”:

• “Акции”: Увеличивают мотивацию пользователей, предлагая выгодные условия, и стимулируют импульсные бронирования.

• “Направления”: Обеспечивают структурированную информацию о доступных местах отдыха, упрощают выбор и помогают пользователю спланировать путешествие.

Оба раздела интегрированы с системой бронирования, что делает процесс выбора и оформления тура быстрым и удобным.

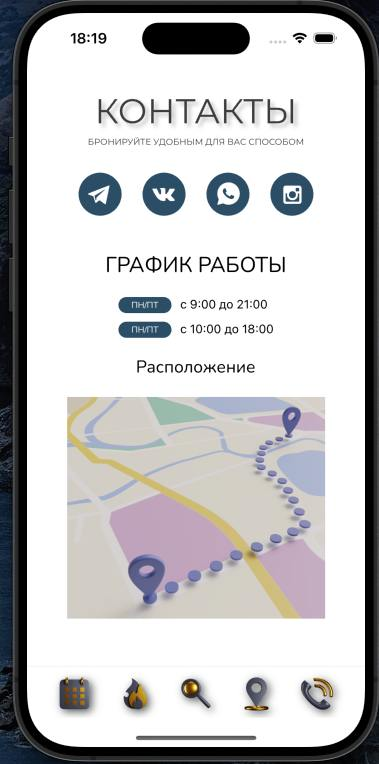


Рис.10, 11 – Интерфейс раздела акций и направлений

Раздел “Контакты” в мобильном приложении туристического агентства обеспечивает удобный доступ пользователей к информации о связи с агентством. Этот раздел позволяет клиентам быстро найти необходимые контактные данные, получить помощь или задать вопросы по интересующим их турам.

Основные элементы раздела “Контакты”:

1. Информация о компании:

• Название туристического агентства.

• Юридический адрес и местоположение офиса.

• Рабочие часы (например, “Пн-Пт с 09:00 до 18:00”).

2. Телефоны:

• Основной номер телефона для связи.

• Номера телефонов для разных отделов, например:

• Бронирование: +7 (123) 456-78-90.

• Вопросы о турах: +7 (123) 456-78-91.

• Служба поддержки: +7 (123) 456-78-92.

• Возможность позвонить прямо из приложения (кнопка “Позвонить”).

3. Электронная почта:

• Основной адрес электронной почты (например, info@touragency.com).

• Специальные адреса для отдельных услуг:

• Для бронирований: bookings@touragency.com.

• Для вопросов и обратной связи: support@touragency.com.

• Кнопка “Отправить письмо”, открывающая почтовый клиент пользователя.

4. Социальные сети и мессенджеры:

• Ссылки на официальные страницы в социальных сетях (Instagram, VK, Facebook).

• QR-коды для добавления в мессенджеры (WhatsApp, Telegram).

• Кнопки для перехода к чатам или мессенджерам прямо из приложения.

5. Форма обратной связи:

• Поля для ввода имени, контактных данных и сообщения.

• Возможность выбора темы обращения (например, “Вопрос о туре”, “Жалоба”, “Обратная связь”).

• Кнопка “Отправить”, с подтверждением успешной отправки.

6. Карта с местоположением офиса:

• Интерактивная карта, отображающая офис агентства.

• Возможность построить маршрут до офиса через сторонние приложения (например, Google Maps или Яндекс.Карты).

7. Часто задаваемые вопросы (FAQ):

• Отдельный раздел с ответами на популярные вопросы, такие как:

• “Как оформить бронирование?”

• “Какие документы нужны для поездки?”

• “Как отменить или изменить бронирование?”

Преимущества раздела “Контакты”:

• Упрощает процесс связи с агентством.

• Повышает доверие пользователей благодаря доступности информации.

• Способствует оперативному решению вопросов и проблем клиентов.

Этот раздел позволяет клиентам быстро и удобно связываться с агентством, получать ответы на свои вопросы и чувствовать себя уверенно при взаимодействии с сервисом.

# **Заключение**

В процессе разработки мобильного приложения для туристического агентства была выполнена полная реализация проекта: от постановки задач и анализа требований до создания работающего программного продукта. Приложение включает все ключевые функции, необходимые для удобного поиска туров, фильтрации по параметрам и оформления бронирований.

Достигнутые результаты:

* Анализ предметной области и существующих решений:
* Проведено исследование современных тенденций в цифровом туризме, что позволило выделить основные требования к приложению.
* Учет актуальных трендов обеспечил создание продукта, соответствующего ожиданиям пользователей.

Проектирование и реализация приложения:

* Разработаны интерфейсы для работы с каталогом туров, фильтрацией, поиском и системой бронирования.
* Реализована возможность авторизации для безопасного доступа к персональным данным.
* Интегрирована административная панель для управления турами и обработкой заявок.

Функциональные особенности:

* Пользователи могут просматривать каталог туров, настраивать фильтры и бронировать туры.
* Администраторы имеют возможность управлять ассортиментом туров и следить за статусами бронирований.

Практическая значимость:

* Разработанное приложение может быть использовано туристическими агентствами как готовое решение для автоматизации процесса бронирования и улучшения пользовательского опыта.
* Приложение помогает повысить эффективность работы агентства, улучшить взаимодействие с клиентами и увеличить продажи.

# **Список литературы**

1. Шимкин, Д. А. Основы программирования на Java для Android. — М.: БХВ-Петербург, 2021. — 280 с.
2. Фрэнк, М. Мобильные приложения для Android: от идеи до реализации. — М.: Основание, 2020. — 256 с.
3. Android Developers. Документация по Android [Электронный ресурс]. — Режим доступа: https://developer.android.com/docs. — Дата обращения: 15.11.2023.
4. Закутин, В. Б. Проектирование пользовательского интерфейса в мобильных приложениях. — М.: Наука, 2021. — 215 с.
5. Кузнецов, А. В. Лучшие практики разработки Android-приложений. — М.: Альпина Паблишер, 2023. — 320 с.