

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

КУРСОВАЯ РАБОТА
по дисциплине «Разработка приложений для мобильных платформ»
Тема: Агрегатор квестов

Студенты гр. 6304

Пискунов Я.А.

Некрасов Н.А.

Прозорова А.Д.

Преподаватель

Заславский М.М.

Санкт-Петербург

2020

ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ (КУРСОВОЙ ПРОЕКТ)

Студенты Пискунов Я.А., Некрасов Н.А., Прозорова А.Д.

Группа 6304

Тема работы: Агрегатор квестов

Исходные данные:

Необходимо реализовать мобильное приложение для агрегации и удобного предоставления информации по квестурам города.

Содержание пояснительной записки:

«Содержание», «Введение», «Сценарии использования», «Пользовательский интерфейс», «Модель данных», «Разработанное приложение»,
«Последовательность действий для осуществления сценариев использования»,
«Заключение», «Список использованных источников», «Приложения»

Предполагаемый объем пояснительной записки:

Не менее 22 страниц.

Дата выдачи задания:

Дата сдачи реферата:

Дата защиты реферата:

Студенты

Пискунов Я.А.

Некрасов Н.А.

Преподаватель

Прозорова А.Д.

Заславский М.М.

АННОТАЦИЯ

Выбрана тема курсовой работы «Агрегатор квестов». Спроектирован интерфейс приложения, который предполагает отображение информации о местонахождении, содержании, цене квеста. Сформированы сценарии использования. Разработано мобильное приложение для платформы Android, которое позволяет искать квесты с помощью фильтров, а также с помощью интерактивной карты. Также в приложении предусмотрена возможность сохранять понравившиеся квесты и совершать их бронирование. Для приложения написаны модульные тесты и тесты пользовательского интерфейса.

SUMMARY

The theme of the course work "Quest Aggregator" was selected. An application interface has been designed, which involves displaying information about the location, content, price of the quest. Use cases generated. A mobile application for the Android platform has been developed that allows you to search for quests using filters, as well as using an interactive map. The application also provides the ability to save your favorite quests and make their reservation. Unit tests and user interface tests are written for the application.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	6
1. СЦЕНАРИИ ИСПОЛЬЗОВАНИЯ	7
1.1. Основной сценарий.....	7
1.2. Альтернативные сценарии	7
1.3. Опциональные шаги	8
2. ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС.....	9
3. МОДЕЛЬ ДАННЫХ.....	11
4. РАЗРАБОТАННОЕ ПРИЛОЖЕНИЕ.....	12
5. ПОСЛЕДОВАТЕЛЬНОСТЬ ДЕЙСТВИЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ СЦЕНАРИЕВ ИСПОЛЬЗОВАНИЯ	14
ЗАКЛЮЧЕНИЕ	21
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	22

ВВЕДЕНИЕ

В обычной жизни людям иногда хочется отдохнуть и провести время в компании друзей. Одним из занимательных способов времяпрепровождения является прохождение квестов.

Цель работы – создать приложение, которое будет хранить информацию по предложениям квеструмов, предоставлять ее пользователю по необходимости, а также давать возможность, не выходя из приложения произвести бронирование понравившегося предложения.

Предлагается решение в виде мобильного приложения под платформу Android, которое использует базу данных под управлением firestore для хранения данных пользователей и информации о предложениях квеструмов.

Данное приложение разрабатывается как мобильное и под платформу Android по ряду причин:

- На данный момент жизнь большинства современных людей неразрывна связана с использованием мобильных устройств.
- Kotlin, будучи языком разработки мобильных приложений под Android, является наиболее известным из всех языков мобильной разработки для авторов.
- На рынке мобильных приложений подобного рода решения представлены мало, а их качество не всегда соответствует ожиданиям пользователей.
- Некоторая часть решений по посещению квестов принимается спонтанно, во время нахождения вдали от стационарных ПК. Пользователям не всегда хочется искать их для осуществления задуманного.

1. СЦЕНАРИИ ИСПОЛЬЗОВАНИЯ

1.1. Основной сценарий

Действующим лицом сценариев является пользователь мобильного приложения (далее – пользователь).

- 1) Пользователь запускает приложение.
- 2) Пользователь проходит процедуру авторизации, вводя EMail и пароль.
- 3) Пользователь переходит на страницу поиска.
- 4) Отображаются параметры поиска. Пользователь вводит требуемые от него данные.
- 5) Пользователь просматривает предложенные варианты и выбирает подходящий.
- 6) Пользователь изучает подробную информацию по квесту и нажимает "забронировать".
- 7) Пользователь вводит необходимые данные для бронирования.
- 8) Пользователь переходит по на страницу оплаты.
- 9) Пользователь вводит необходимые данные для оплаты.
- 10) Пользователь переходит на страницу забронированных заказов.

1.2. Альтернативные сценарии

1.1 Перед шагом 2: Пользователь проходит процедуру регистрации, вводя EMail, пароль и логин => возврат к шагу 3 основного сценария.

2.1 Перед шагом 2: Пользователь проходит процедуру авторизации с использованием социальных сетей => возврат к шагу 3 основного сценария.

3.1 Перед шагом 3: Пользователь переходит на страницу забронированных заказов => завершение сценария.

3.2 Пользователь просматривает подробную информацию о забронированном квесте => возврат к шагу 8 основного сценария.

3.3 Пользователь отменяет бронирование квеста => возврат к шагу 8 основного сценария.

4.1 Перед шагом 3: Пользователь переходит на страницу избранных квестов и выбирает квест => возврат к шагу 6 основного сценария.

4.2 Пользователь убирает квест из избранного => возврат к шагу 3 основного сценария.

5.1 Перед шагом 3: Пользователь переходит на страницу с картой => возврат к шагу 5 основного сценария.

6.1 Перед шагом 3: Пользователь переходит на страницу с профилем.

6.2 Пользователь производит настройку уведомлений и прочего => переход к шагу 3 основного сценария.

6.3 Пользователь производит изменения имени, EMail, даты рождения, пароля => возврат к шагу 3 основного сценария.

1.3. Опциональные шаги

1) Перед шагом 5, перед шагом 6, перед шагом 3.3. Пользователь добавляет квест в избранное.

2) Перед шагом 5, перед шагом 6, перед шагом 3.3. Пользователь убирает квест из избранного, если он уже находится в списке избранных.

2. ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС

Детальный макет интерфейса, а также интерактивная модель взаимодействия представлены в [1]. На рис. 1 и рис. 2 представлены графы переходов.

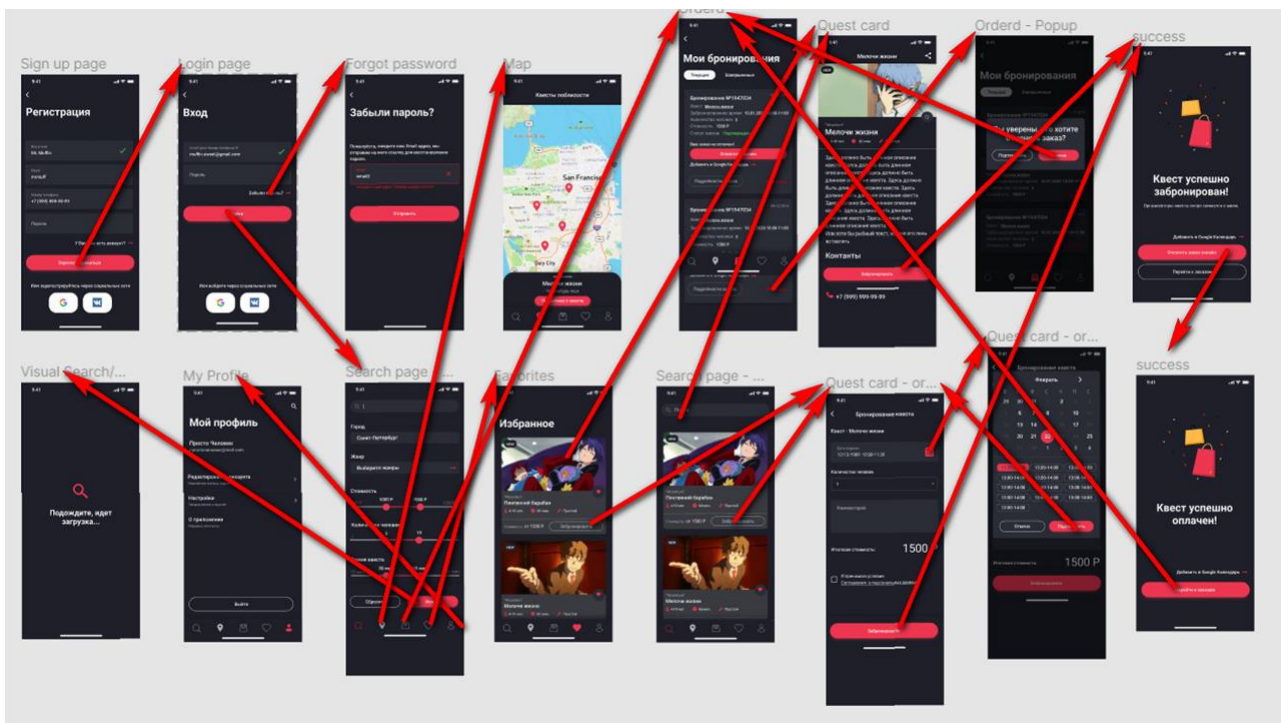


Рисунок 1 – Основной граф переходов

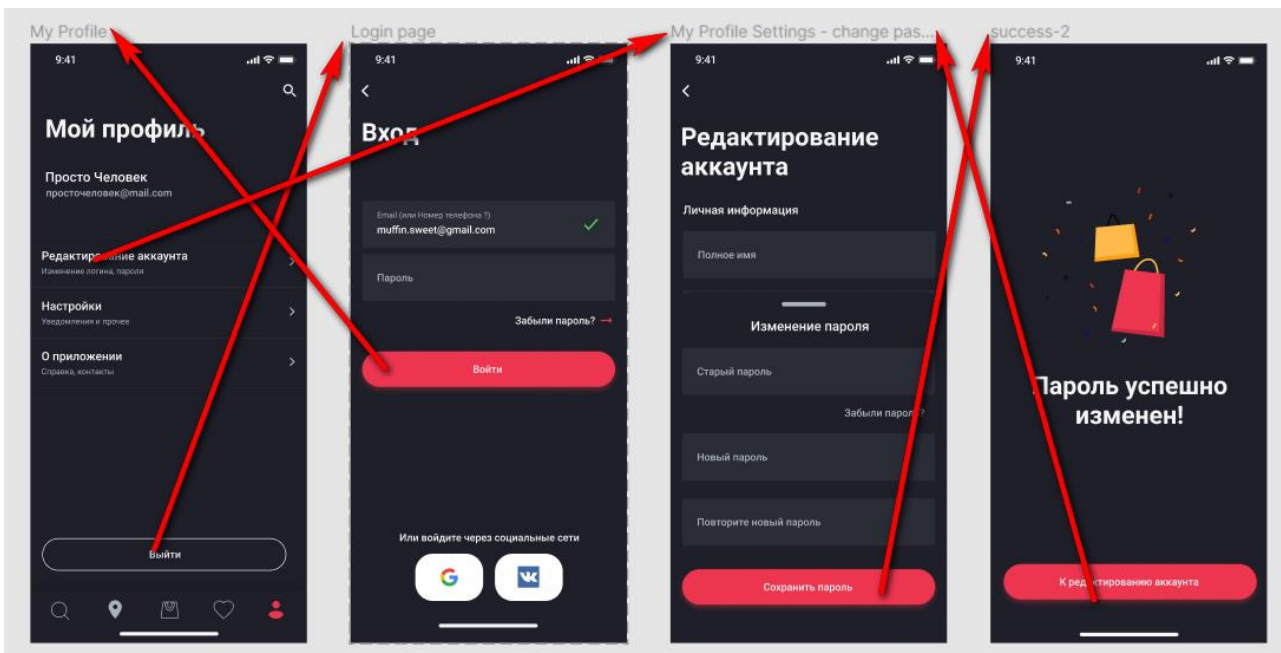


Рисунок 2 – Граф переходов

Учитывая специфику приложения, оно подойдет как для смартфонов, так и для иных устройств на Android. Единственными условиями комфортного

использования являются наличие доступа в интернет (скорость доступа влияет лишь на скорость получения информации, т.е. пользователь в данном случае сам для себя определяет уровень комфортности) и достаточно большого экрана (диагональ не менее 3 дюймов) для удобства взаимодействия.

3. МОДЕЛЬ ДАННЫХ

На данный момент в качестве информации используются случайно сгенерированные данные. В дальнейшем имеется возможность заполнить базу данными из открытых источников.

Центральная сущность – «пользователь» (User). Олицетворяет собой пользователя. Также имеются сущности «заказ» (Order) и «Квест» (Quest).

«Пользователь» необходима для хранения информации о зарегистрированных пользователях, в том числе и информацию о квестах, добавленных в избранное и сделанных заказов. В «заказ» хранится информация о сделанном заказе. Для связи с определенным квестом тут же хранится Id квеста. В «Квест» хранится информация о предлагаемом квесте. Графическое изображение модели данных представлено на рис. 3.

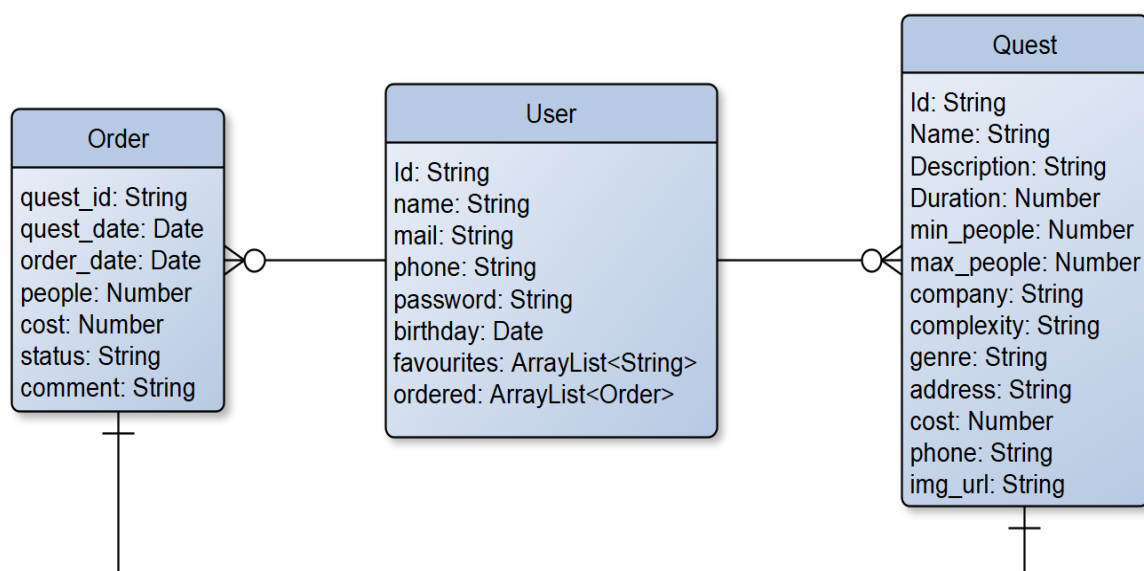


Рисунок 3 – Модель данных

4. РАЗРАБОТАННОЕ ПРИЛОЖЕНИЕ

Разработано мобильное приложение для платформы Android, с помощью которого можно за некоторое количество простейших действий, которые не сильно отличаются от простого нажатия, просмотреть, выбрать, забронировать квест. Выбор можно осуществлять несколькими способами: с использованием фильтра и текстового поиска или карты. Код приложения представлен в [2].

В качестве библиотек платформы Android использованы:

- org.jetbrains.kotlin:kotlin-stdlib-jdk7
- androidx.appcompat:appcompat:1.1.0
- androidx.core:core-ktx:1.2.0
- com.google.android.material:material:1.1.0
- androidx.constraintlayout:constraintlayout:1.1.3
- com.android.support:multidex:1.0.3
- com.google.firebase:firebase-analytics:17.2.2
- com.google.firebase:firebase-database-ktx:19.2.1
- androidx.navigation:navigation-fragment-ktx:2.2.1
- androidx.navigation:navigation-ui-ktx:2.2.1
- androidx.legacy:legacy-support-v4:1.0.0
- com.google.firebase:firebase-database:19.2.1
- com.google.firebase:firebase-firestore:21.4.2
- junit:junit:4.12
- androidx.test.ext:junit:1.1.1
- androidx.test.espresso:espresso-core:3.2.0
- com.yandex.android:mapkit:3.5.0
- com.yandex.android:directions:3.5.0
- com.yandex.android:places:3.5.0
- com.yandex.android:search:3.5.0
- com.yandex.android:transport:3.5.0

Снимки экранов представлены в приложении А, архитектура приложения представлена на рис. 4.

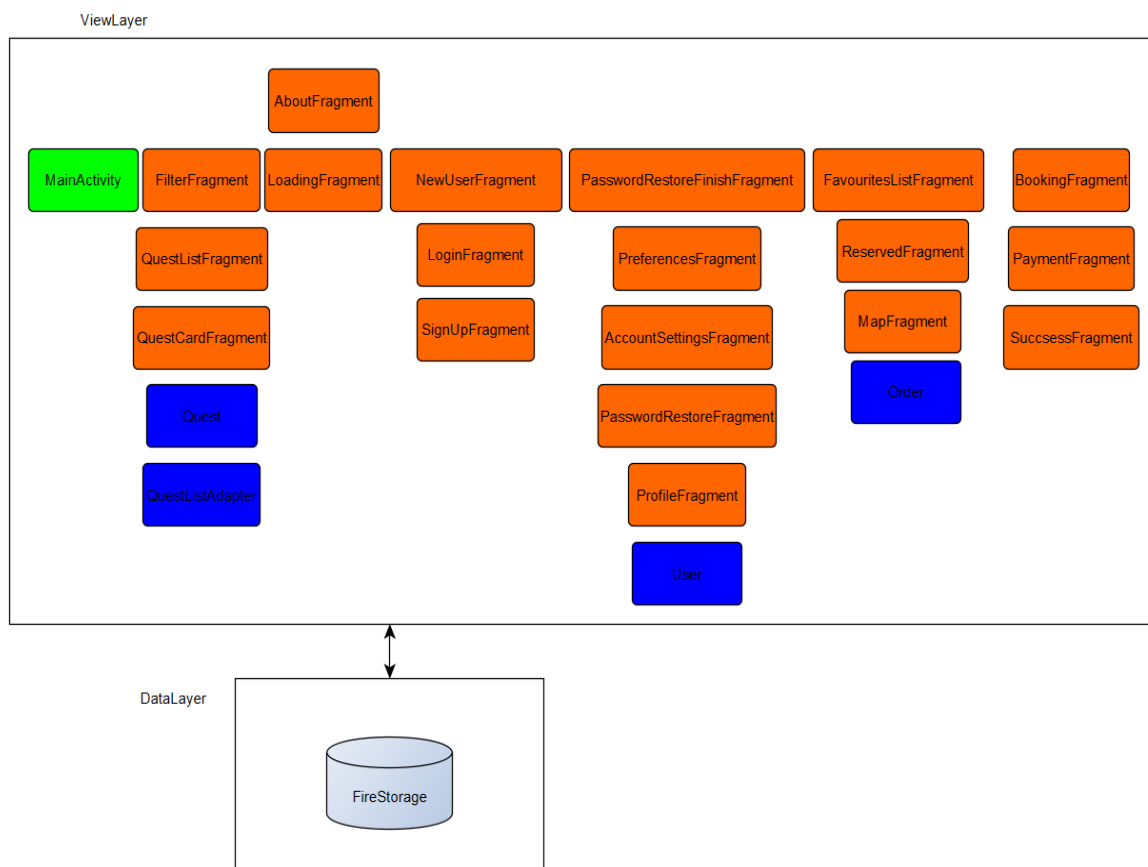


Рисунок 4 – Архитектура приложения

Учитывая, что уже произведена разработка приложения для Android без использования паттернов, библиотек и фреймворков кроссплатформенной разработки, то наименее трудозатратным будет разработать для каждой платформы отдельное приложение. В противном случае придется переделывать уже созданное приложение, обесценивая проделанный ранее труд, а время важнее всего.

5. ПОСЛЕДОВАТЕЛЬНОСТЬ ДЕЙСТВИЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ СЦЕНАРИЕВ ИСПОЛЬЗОВАНИЯ

Произведем расчет сложности интерфейса для основной последовательности действий из сценариев использования. Для конкретики будем считать, что пользователь запускает приложение впервые и не использует альтернативные сценарии. Результат подсчета количества определенных действий представлен в табл. 1.

Таблица 1. Подсчет количества действий основной последовательности

Пункт сценария использования	Действие	Количество
Запуск приложения		
	Нажатие	1
Авторизация		
	Ввод текста	2
	Нажатие	1
Переход на страницу поиска		
	Нажатие	1
Ввод данных для поиска		
	Ввод текста	2
	Выбор нескольких из списка	1
	Выбор из диапазона	3

Пункт сценария использования	Действие	Количество
	Нажатие	2
Выбор подходящего квеста		
	Нажатие	1
Изучение информации и начало бронирования		
	Нажатие	1
Ввод необходимых для бронирования данных		
	Ввод текста	3
	Выбор даты	1
	Выбор из выпадающего списка	1
	Нажатие	2
Переход к странице оплаты бронирования		
	Нажатие	1
Ввод необходимых данных для оплаты		

Пункт сценария использования	Действие	Количество
	Ввод текста	4
	Нажатие	2
	Выбор нескольких из списка	1
Переход на страницу забронированных		
	Нажатие	1

В качестве альтернативы разрабатываемому приложению рассмотрим реально существующее приложение для Android "Топ квестов - квесты в реальности". Произведем аналогичный расчет количества действий от первого открытия приложения до перехода на страницу с забронированными заказами после оплаты. Результаты подсчета представлены в табл. 2.

Таблица 2. Подсчет действий основной последовательности конкурента

Пункт Сценария использования	Действие	Количество
Запуск приложения		
	Нажатие	1
Ввод данных для поиска		
	Выбор из выпадающего списка	3
	Выбор нескольких	1

Пункт Сценария использования	Действие	Количество
	из списка	
	Нажатие	1
Ввод дополнительных данных фильтрации		
	Выбор нескольких из списка	1
	Ввод текста	1
	Нажатие	3
Выбор подходящего квеста		
	Нажатие	1
Изучение информации и начало бронирования		
	Нажатие	2
Ввод необходимых для бронирования данных		
	Ввод текста	3
	Нажатие	3
Ввод необходимых для оплаты данных		
	Ввод текста	4
	Нажатие	1

Пункт Сценария использования	Действие	Количество
Переход на страницу забронированных		
	Нажатие	1

Следует отметить, что выбор даты в приложении конкурента реализован, по сути, через выбор из огромного списка и производится через просто нажатия. Однако, при бронировании на долгий срок вперед количество дополнительных нажатий резко возрастает.

К сожалению, из-за ситуации в городе квеструмы не принимают заказы и нет возможности воспроизвести основную последовательность действий в приложении для снятия скриншотов.

Произведем подсчет общего числа действий для обоих приложений. Кроме того, для каждого действия добавим оценку сложности. Чем меньше значение - тем более простым это действие является для пользователя. Результаты представлены в табл. 3.

Таблица 3. Сравнение общего числа действий

Действие	Оценка сложности	Количество	Количество у конкурента
Нажатие	1	13	13
Ввод текста	5	11	8
Выбор из диапазона	4	3	0
Выбор нескольких из списка	3	2	2
Выбор из выпадающего списка	2	1	3

Действие	Оценка сложности	Количество	Количество у конкурента
Выбор даты	2	1	0

Как можно отметить, число действий не сильно отличается. Можно вычислить сравнительную сложность последовательностей, умножив оценку сложности на количество производимых действий, после чего сложить. Таким образом, получится, что сложность создаваемого приложения составляет 90, а у конкурента - 65. Что не является особо большой разницей, учитывая факт необходимости выбора даты большим количеством нажатий у конкурента при определенных условиях.

Возможные улучшения интерфейса. Очевидным улучшением является добавление авторизации через социальные сети для сокращения числа действий. Данная функция уже предусмотрена и в макете интерфейса имеются соответствующие изображения.

Другим сложным действием является выбор из диапазона. Он реализован как выбор с использованием двойного слайдера. Для упрощения его можно заменить на выбор с использованием одинарного слайдера и возможностью задать границы для более точного ввода. Пример представлен на рис. 1. Сложность такого действия можно оценить в 2. Таким образом, общая сложность интерфейса снизится до 84.

Также можно несколько изменить ввод данных для фильтрации, заменив ввод текста на выбор из списка, в котором город пользователя можно выводить наверх, используя данные геолокации при дозволении пользователя. Это снизит сложность до 81.

Указанные выше изменения успешно внедрены в интерфейс на стадии его разработки. Учитывая, что в большинстве случаев пользователь не ищет конкретный квест, поле названия останется пустым, а действия ввода текста

производиться не будет, что также потенциально снижает сложность интерфейса до 76.

ЗАКЛЮЧЕНИЕ

Разработано мобильное приложение для платформы Android, с помощью которого можно за некоторое количество простейших действий, которые не сильно отличаются от простого нажатия, просмотреть, выбрать, забронировать квест. Выбор можно осуществлять несколькими способами: с использованием фильтра и текстового поиска или карты.

К недостаткам решения можно отнести непривлекательный дизайн, который создает впечатление недоделанного продукта, необходимость авторизации для доступа к большинству функционала.

В качестве развития решения предлагается прежде всего улучшение интерфейса в сторону его большей привлекательности, что также подразумевает вероятное изменение способов взаимодействия пользователя с приложением. Очевидным способом развития решения является его портирование на iOS, что позволило бы покрыть практически весь спектр устройств, а, соответственно, и пользователей.

Одним из более широких развитий решения может быть расширение в сторону агрегации не только квестов, но и иных предложений по развлечениям.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Real Quests - Figma // figma.com. URL:
<https://www.figma.com/file/bzU10p0wrbR5N5uBc3sWAK/Real-quests?node-id=0%3A1> (дата обращения: 20.04.2020).
2. Репозиторий приложения // github.com. URL:
<https://github.com/moevm/adfmp20-quests/tree/master> (дата обращения: 21.04.2020).