**ОГЛАВЛЕНИЕ**

[**ПЕРЕЧЕНЬ УСЛОВНЫХ ОБОЗНАЧЕНИЙ** 7](#_Toc105714304)

[**ВВЕДЕНИЕ** 8](#_Toc105714305)

[**1.** **ПОСТАНОВКА ЗАДАЧИ** 10](#_Toc105714306)

[**1.1.** **Формирование требований к программному продукту** 10](#_Toc105714307)

[**1.1.1.** **Бизнес-требования** 10](#_Toc105714308)

[**1.1.2.** **Пользовательские требования** 10](#_Toc105714309)

[**1.1.3.** **Функциональные требования** 12](#_Toc105714310)

[**1.1.4.** **Нефункциональные требования** 12](#_Toc105714311)

[**1.1.5.** **Ограничения** 13](#_Toc105714312)

[**1.1.6.** **Требование к интерфейсам** 13](#_Toc105714313)

[**1.1.7.** **Требования к данным** 13](#_Toc105714314)

[**1.2.** **Программные средства разработки** 14](#_Toc105714315)

[**1.3.** **Аппаратные средства разработки** 15](#_Toc105714316)

[**2.** **ПРОЕКТИРОВАНИЕ ПРОГРАММНЫХ МОДУЛЕЙ** 16](#_Toc105714317)

[**2.1.** **Архитектура системы (диаграмма компонентов)** 16](#_Toc105714318)

[**2.2.** **Моделирование основных сценариев системы** 18](#_Toc105714319)

[**2.3.** **Проектирование графического интерфейса пользователя** 21](#_Toc105714320)

[**2.4.** **Проектирование и разработка модели данных** 24](#_Toc105714321)

[**3.** **РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА** 26](#_Toc105714322)

[**3.1. Разработка мобильного приложения** 26](#_Toc105714323)

[**3.1.1. Выбранные языки программирования и среда разработки** 26](#_Toc105714324)

[**3.1.2. Выбор и подключение базы данных** 26](#_Toc105714325)

[**3.2. Разработка авторизации и регистрации в мобильном приложении** 28](#_Toc105714326)

[**3.3. Выход из учётной записи пользователя** 31](#_Toc105714327)

[**3.4. Запись на мероприятие** 31](#_Toc105714328)

[**3.5. Отображение всплывающих окон** 31](#_Toc105714329)

[**3.6. Отображение видео** 31](#_Toc105714330)

[**3.7. Разработка карты в мобильном приложении** 32](#_Toc105714331)

[**4.** **ТЕСТИРОВАНИЕ ПРОГРАММНОГО ПРОДУКТА** 34](#_Toc105714332)

[**4.1. История изменений** 34](#_Toc105714333)

[**4.2. Терминология** 35](#_Toc105714334)

[**4.3. Стратегия тестирования программного обеспечения** 35](#_Toc105714335)

[**4.4. Определение объектов тестирования** 36](#_Toc105714336)

[**4.5. Архитектура тестируемой системы.** 36](#_Toc105714337)

[**4.6. Описание процесса тестирования.** 37](#_Toc105714338)

[**5.** **РАЗРАБОТКА ТЕХНОЛОГИЧЕСКОЙ ДОКУМЕНТАЦИИ** 46](#_Toc105714339)

[**5.1. Руководство пользователя** 46](#_Toc105714340)

[**5.2. Руководство администратора** 53](#_Toc105714341)

[**6.** **МЕРОПРИЯТИЯ ПО ТЕХНИКЕ БЕЗОПАСТНОСТИ И ОХРАНЕ ТРУДА** 54](#_Toc105714342)

[**6.1. Характеристика объекта проектирования и условий его эксплуатации** 54](#_Toc105714343)

[**6.2. Мероприятия по технике безопасности** 55](#_Toc105714344)

[**6.3. Противопожарная безопасность** 57](#_Toc105714345)

[**7.** **ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ ПРОЕКТА** 59](#_Toc105714346)

[**7.1.** **Расчет затрат на разработку мобильного приложения** 59](#_Toc105714347)

[**7.1.1.** **Затраты на оплату труда (Зот)** 59](#_Toc105714348)

[**7.1.2.** **Эксплуатационные затраты на оборудование** 60](#_Toc105714349)

[**7.1.3.** **Затраты на материалы** 61](#_Toc105714350)

[**7.2** **Себестоимость разработки программного обеспечения** 62](#_Toc105714351)

[**7.3.** **Расчет экономического эффекта разработчика и пользователя (заказчика) программного обеспечения** 62](#_Toc105714352)

[**7.3.1.** **Экономический эффект у разработчика программного обеспечения** 62](#_Toc105714353)

[**7.3.2.** **Экономический вывод** 63](#_Toc105714354)

[**ЗАКЛЮЧЕНИЕ** 64](#_Toc105714355)

[**СПИСОК ИСТОЧНИКОВ ИНФОРМАЦИИ** 65](#_Toc105714356)

[**ПРИЛОЖЕНИЕ** 66](#_Toc105714357)

# **ПЕРЕЧЕНЬ УСЛОВНЫХ ОБОЗНАЧЕНИЙ**

**БД** – База данных;

**TЗ** – Техническое задание;

**DOM –** Document Object Model;

**JDK** – Java Development Kit;

**Android SDK** – Android Software Development Kit;

**AVD Manager** - Android Visual Device Manager.

# **ВВЕДЕНИЕ**

Данная дипломная работа связана со сферой торговли и спорта. Исследуемой компанией является ООО «Skate.CO», которая предоставляет свои товары и услуги для скейтбордистов. Сфера компании на данный момент направлена на торговлю, которая стремительно развивается, т.к. потребность в качественных товарах и услугах возрастает пропорционально росту ритма и уровня жизни. Глобализация и концентрация знаний и информации в глобальном Интернете вносят свои коррективы в деятельность торговых компаний. Сегодня клиентам становится все труднее найти время, чтобы прийти в магазин компании и получить информацию о доступных товарах. И они предпочитают искать его в Интернете. Также большинство покупателей вынуждены приходить в магазины, чтобы просмотреть товар и приобрести его.

Однако, несмотря на успехи компании в сфере торговли, руководители предприняли решение расширить сферу деятельности с помощью разработки мобильного приложения. Статистические данные показывают, что количество трафика с мобильных устройств растет из года в год. Мобильное приложение позволит компании привлечь ту аудиторию, которая отдает свои предпочтения мобильным гаджетам и все реже пользуется настольными компьютерами и ноутбуками. Наличие мобильного приложения позволит покупателям совершить покупку здесь и сейчас, не откладывая принятие решения на потом, когда будет доступен компьютер. Наличие мобильного приложения повышает эффективность и результативность рекламных кампаний в офф-лайне. Мобильные приложения активно развиваются, многие компании приходят к пониманию необходимости создания собственного приложения, но все же количество приложений мало по сравнению с количеством сайтов в сети интернет. Это дает компании возможность обратиться к потребителю в менее конкурентном пространстве.

На данный момент на российском рынке отсутствуют компании, у которых бы имелось подобное реализуемому приложение. Актуальность создания мобильного приложения для данной компании характеризуется необходимостью скейтбордистов, а именно:

* скейтбордист должен, выйдя в интернет, быстро получить информацию о ближайших выделенных специализированных скейтпарках, локальных парках;
* скейтбордист должен узнать о поступлении новых комплектующих в магазин в новостных разделах;
* создание приложения, работающего при выходе с рынка зарубежных компаний, предоставляющих схожие услуги;
* скейтбордист сможет записаться на актуальные мероприятия, с помощью данного мобильного приложения.

Предметом исследования является мобильное приложение торгово-спортивной компании.

Этот проект связан с поиском и предоставлением необходимой информации пользователям, которые ищут ее в информационном пространстве. Сама по себе проблема поиска информации очень актуальна, и многим пользователям довольно сложно ориентироваться в огромных объемах данных, которые предоставляют информационные системы. Однако реализация результатов работы позволит создать эффективную систему взаимодействия торговой компании с конечным потребителем и выстроить работающую систему продаж продукции компании конечному потребителю. Также выстроить работающее мобильное приложение, представляющее собой личный кабинет потребителя и карту города с отмеченными на ней скейтпарками.

Требуется дать клиенту возможность пользоваться информацией в удобное для него время. Покупателю не нужен стационарный компьютер, достаточно иметь смартфон или планшет, чтобы получить нужную информацию о компании или товаре. Помимо этого, пользователь мобильного приложения сможет записаться на различные скейтбординг-мероприятия, проходящие в городе.

Цель дипломного проекта: разработка мобильного приложения для организации социального взаимодействия участников мероприятий по скейтбордингу.

Задачи дипломного проекта:

* -провести анализ теоретических основ разработки мобильного приложения для организации социального взаимодействия участников мероприятий по скейтбордингу;
* -провести анализ программных сред и средств для разработки мобильного приложения для организации социального взаимодействия участников мероприятий по скейтбордингу;
* разработать модель бизнес-процессов для организации социального взаимодействия участников мероприятий по скейтбордингу;
* провести разработку мобильного приложения для организации социального взаимодействия участников мероприятий по скейтбордингу;
* провести тестирование мобильного приложения для организации социального взаимодействия участников мероприятий по скейтбордингу;
* разработать описание и порядок пользования мобильного приложения для организации социального взаимодействия участников мероприятий по скейтбордингу;
* разработать мероприятия по охране труда и техники безопасности при проектировании и эксплуатации мобильного приложения;
* выполнить расчеты по экономическому обоснованию эффективности разработанного программного продукта для организации социального взаимодействия участников мероприятий по скейтбордингу.

Объект исследования: деятельность по организации социального взаимодействия участников мероприятий по скейтбордингу.

Предмет исследования: мобильные приложения.

Источниковедческая база:

* электронно-библиотечная система СПбГУТ;
* электронно-библиотечная система IPR-books;
* электронно- библиотечная система Лань;
* электронно- библиотечная система ibooks;
* электронно- библиотечная система Юрайт для СПбКТ;
* электронно- библиотечная система Znanium для СПбКТ;
* электронно- библиотечная система book.ru для СПбГУТ и СПбКТ;
* электронная библиотека Polpred.

# **ПОСТАНОВКА ЗАДАЧИ**

Требуется реализовать мобильное приложение на ОС Android для компании Skate.CO. Мобильное приложение должно отвечать следующим требованиям:

* 1. **Формирование требований к программному продукту**
     1. **Бизнес-требования**

Идея создания мобильного приложения возникла после изучения рынка услуг, предоставляемых другими компаниями. Как было выявлено у многих компаний в данной сфере имеется только магазин-сайт, реже имелась группа в популярных социальных сетях. Создание нового мобильного приложения, не имеющего конкуренции на рынке, станет серьёзным шагом для роста количества новых клиентов и пользователей сервиса «Skate.CO».

Бизнес-возможности данного проекта оцениваются как высокие. Внедрение на рынок данного мобильного приложения станет серьёзным шагом в стабилизации на рынке услуг для «Skate.CO». Компании-конкуренты будут вынуждены либо внедрять подобные технологии в свои компании, либо терпеть спад клиентской базы.

Внедрение данного приложения позволит получить выгоды как для компании, так и для рядовых пользователей услуг.

Для компании:

* + 1. в первые месяцы успешной рекламной кампании позволят серьёзно нарастить клиентскую базу, за счёт чего увеличится прибыль, возможны получения денежных средств от инвесторов;
    2. отсутствие конкуренции на рынке позволит в долгосрочной перспективе занимать лидирующие позиции в коммерческой системе.

Для пользователей:

* + 1. рядовые пользователи получают приложение, отвечающее их потребностям;
    2. пользователь сможет получить необходимую информацию находясь не за персональным компьютером, а в любой точке своего города имея при себе мобильное устройство и доступ в интернет;
    3. пользователи сможет обучать, либо учиться катанию на скейтборде.

Рисками данного приложения могут являться:

* + 1. в долгосрочном плане компании-конкуренты будут вынуждены создавать аналогичные или схожие мобильные приложения;
    2. при адаптации и внедрении компаниями-конкурентами новых мобильных приложений существует вероятность потери прибыльности компании «Skate.CO»;
    3. успешность внедрения прямо пропорционально зависит от успешности рекламной компании;
    4. пользователи могут воспринять данное приложение за сложный программный продукт, поэтому следует разработать понятный пользователям интерфейс.

Мобильное приложение должно соответствовать требования федеральных законов:

* №152-ФЗ «О персональных данных» при хранении и обработке персональных данных покупателей;
* №149 ФЗ «Об информации, информационных технологиях и защите информации» при поиске любой информации на сайте.
  + 1. **Пользовательские требования**

В приложении должно быть реализована классификация пользователей, благодаря этому компания сможет более детально и рационально обеспечить выполнение требований пользователей.

Диаграмма сценариев использования представлена на рисунке 1.1.

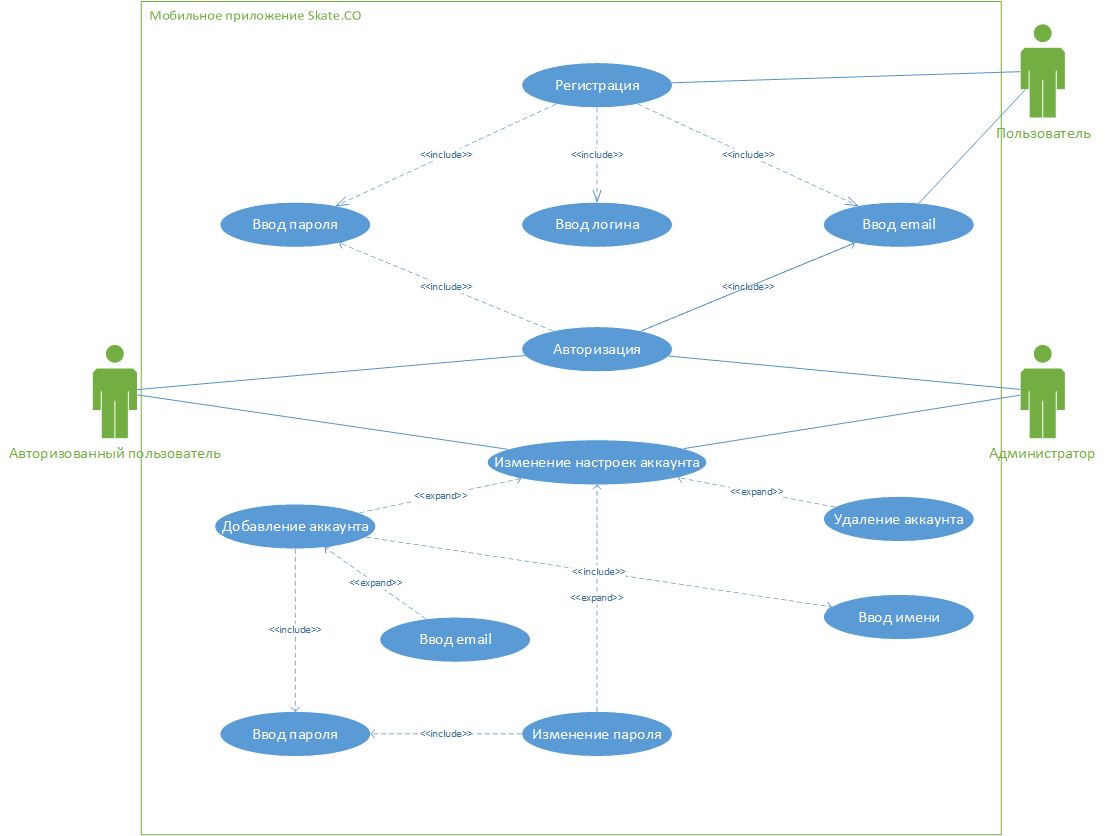


Рисунок 1.1. Сценарий использования мобильного приложения

Классификация выбрана следующая:

* + 1. Неавторизованный пользователь (гость)

Определение требований неавторизованного пользователя (гостя)

Гость имеет ограниченный функционал, гостю потребуется пройти регистрацию. Гость также может обратится в поддержку.

* + 1. Пользователь

Определение требований пользователя

Авторизованный пользователь в мобильном приложении компании «Skate.CO» сможет ознакомиться с новинками на онлайн-полках магазина, просмотреть расположение скейтпарков или локальных местах, обратиться в техническую поддержку, посетить страницы форума.

Одна из главных задач, которую может выполнить пользователь — пользователь может записаться на мероприятия по скейтбордингу.

Пользователь сможет просматривать обучающие видео.

Также пользователь сможет оставлять геолокацию обнаруженного им скейтпарка, которую нужно будет одобрить администрацией.

* + 1. Администратор

Администратор может редактировать данные пользователей напрямую в БД, может удалять или добавлять пользователей.

* + 1. **Функциональные требования**

В мобильном приложении должно быть осуществлено:

1. Страница неавторизованного пользователя, где будет отображаться:
   * Авторизация/регистрация;
   * Поле ввода логина и пароля. Для регистрации необходимо дважды ввести пароль;
   * Кнопка «Вход»;
2. Страница авторизованного пользователя:
   * меню включает:
     + новостную секцию;
     + запись на мероприятие;
     + страницу обучающего контента;
     + личный профиль;
     + карту.
   * Свежие новости компании.
3. В личном кабинете пользователя должно быть реализованы кнопки:
   * изображение пользователя;
   * Редактировать профиль;
   * техническая поддержка;
   * просмотр записи на мероприятие;
   * Выход из учетной записи.
4. Страница технической помощи:
   * вывод номера обращения в техническую поддержку;
   * вывод даты обращения в техническую поддержку;
   * окно сообщений:
     + сообщения технической поддержки отображаются в левой части;
     + сообщения пользователей отображаются в правой части.
   * текстовое окно для ввода сообщений;
   * кнопка отправки сообщения.
5. Страница карты.
6. Страница новостей:
   * отображение новости;
   * кнопки меню.
7. Запись на мероприятия:
   * отображение записи;
   * кнопка записаться
     1. **Нефункциональные требования**

Поскольку современные технологии быстро развиваются, разрабатываемой системе необходимо быть масштабируемой. Масштабируемость – это мера способности системы увеличивать или уменьшать производительность в ответ на изменения в требованиях к обработке приложений и системы. Также, при увеличении числа пользователей база данных должна выдерживать растущее число запросов.

Кроме того, система должна обладать высокой надежностью, т.е. программному обеспечению необходимо сохранять работоспособное состояние в случае обработки неожиданных действий.

Визуальная составляющая мобильного приложения должна быть удобна и понятна пользователю. Также в реалиях 21-го века, мобильное приложение должна иметь способность быть прочтённой человеком с проблемами со зрением, т.е. должна быть реализована версия для слабовидящих пользователей.

Скорость обработки функций приложения должна быть не менее 1000 записей в секунду

При вводе паролей должны быть обработаны исключения, если пользователь ошибётся в вводе букв и цифр.

Пользователь должен иметь мобильное устройство по характеристикам не хуже:

* Версия Android: не ниже 5.0;
* ОЗУ: более 512Мб;
* Объём памяти более 4Гб;
* Разрешение экрана: 1136x640 пикселей.
  + 1. **Ограничения**

Мобильное приложение должно быть написано на языке Java под операционную систему Android.

Адаптивный дизайн приложения исходя из различного количества Android-смартфонов с разными разрешениями экрана, для каждого разрешения нужно:

* растянуть интерфейс и контент под размеры экрана;
* масштабировать изображения без потери качества;
* масштабировать элементы управления, чтобы ими комфортно было пользоваться;
* выдерживать размер шрифтов так, чтобы тексты были читабельны;
* избежать наслоения элементов;
* не дать целевым кнопкам уйти за пределы экрана;
* продумать поведение экранной клавиатуры.
  + 1. **Требование к интерфейсам**

Требования к интерфейсам заключаются в:

1. Стандарты шрифтов:
   * единый шрифт: Roboto 14 Regular;
   * запасной шрифт, если на мобильном устройстве пользователя не поддерживается основной шрифт: Times New Roman 14 Regular.
2. Цветовая схема:
   * основные элементы сайта должны быть оформлены по RGB палитре: 255, 215, 142;
   * задний фон приложения должны быть оформлены по RGB палитре: 245, 245, 245;
   * кнопки приложения должны быть оформлены по RGB палитре: 30, 33, 41.
3. Размеры экранов:
   * минимальный 800х600;
   * максимальный 2440х4800.
     1. **Требования к данным**

Необходимо спроектировать базу данных, в которой будет хранится сведения о пользователях, товарах, видео-контент:

В базе данных должна хранится следующая информация:

Об логинах и паролях пользователей:

* ID пользователя;
* никнейм или ФИО пользователя;
* еmail пользователя;
* логин пользователя;
* пароль пользователя.

О видеоконтенте:

* ID видео;
* ссылка на видео в пространстве хостинга.

Все данные необходимо хранить в структурированном виде под управлением реляционной базы данных. Исключение будут составлять файлы по типу изображений и иконок. Такие файлы должны храниться в файловой системе.

* 1. **Программные средства разработки**

Программные средства разработки – это компьютерные программы, которые используются разработчиками программного обеспечения для создания, редактирования, обслуживания, поддержки и отладки других приложений, сред и программ.

Инструменты разработки могут быть разных форм, таких как компоновщики, компиляторы, редакторы кода, конструктор графического интерфейса пользователя, ассемблеры, отладчик, инструменты анализа производительности и т. д. При выборе соответствующего инструмента разработки необходимо учитывать определенные факторы в зависимости от типа проекта.

Разработка данного программного продукта осуществлялась под управлением операционной системы MS Windows 10. Данная операционная система отличается высокой производительностью, удобством использования и высокий уровень совместимости.

В качестве редактора кода использовалась IntelliJ IDEA. IntelliJ IDEA — интегрированная среда разработки программного обеспечения для многих языков программирования, в частности Java, JavaScript, Python, разработанная компанией JetBrains.

Android Studio — интегрированная среда разработки для работы с платформой Android, анонсированная 16 мая 2013 года на конференции Google I/O

Новые функции появляются с каждой новой версией Android Studio. На данный момент доступны следующие функции:

* расширенный редактор макетов: [WYSIWYG](https://ru.wikipedia.org/wiki/WYSIWYG), способность работать с [UI](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F) компонентами при помощи [Drag-and-Drop](https://ru.wikipedia.org/wiki/Drag-and-drop), функция предпросмотра макета на нескольких конфигурациях экрана;
* сборка приложений, основанная на [Gradle](https://ru.wikipedia.org/wiki/Gradle);
* различные виды сборок и генерация нескольких [.apk](https://ru.wikipedia.org/wiki/.APK) файлов;
* [рефакторинг](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D1%84%D0%B0%D0%BA%D1%82%D0%BE%D1%80%D0%B8%D0%BD%D0%B3) кода;
* статический анализатор кода ([Lint](https://ru.wikipedia.org/wiki/Lint)), позволяющий находить проблемы производительности, несовместимости версий и другое;
* встроенный ProGuard и [утилита](https://ru.wikipedia.org/wiki/%D0%A3%D1%82%D0%B8%D0%BB%D0%B8%D1%82%D0%B0) для подписывания приложений;
* шаблоны основных макетов и компонентов Android;
* поддержка разработки приложений для [Android Wear](https://ru.wikipedia.org/wiki/Android_Wear) и [Android TV](https://ru.wikipedia.org/wiki/Android_TV);
* встроенная поддержка [Google Cloud Platform](https://ru.wikipedia.org/wiki/Google_Cloud_Platform), которая включает в себя интеграцию с сервисами [Google Cloud Messaging](https://ru.wikipedia.org/wiki/Google_Cloud_Messaging) и [App Engine](https://ru.wikipedia.org/w/index.php?title=App_Engine&action=edit&redlink=1);
* Android Studio 2.1 поддерживает [Android N](https://ru.wikipedia.org/wiki/Android_SDK) Preview [SDK](https://ru.wikipedia.org/wiki/SDK), а это значит, что разработчики смогут начать работу по созданию приложения для новой программной платформы;
* новая версия Android Studio 2.1 способна работать с обновленным компилятором Jack, а также получила улучшенную поддержку Java 8 и усовершенствованную функцию Instant Run;
* начиная с Platform-tools 23.1.0 для [Linux](https://ru.wikipedia.org/wiki/Linux) исключительно 64-разрядная.
  1. **Аппаратные средства разработки**

В ходе разработки использовались следующие аппаратные средства:

Персональный компьютер;

В состав системного блока персонального компьютера, на котором происходила разработка, входят:

* процессор Intel Core i7-11700K;
* видеоадаптер Nvidia GeForce RTX 3070 8Гб;
* оперативная память 2 х Crucial Vengeance LPX 8Гб;
* жесткий диск WD Caviar Blue WD10EZEX SATA III;
* SSD Накопитель Apacer Panther 540Гб;
* монитор AOC G2260VWQ6.

Android смартфон.

# **ПРОЕКТИРОВАНИЕ ПРОГРАММНЫХ МОДУЛЕЙ**

* 1. **Архитектура системы (диаграмма компонентов)**

Архитектура мобильного представлена на рисунке 2.1.:

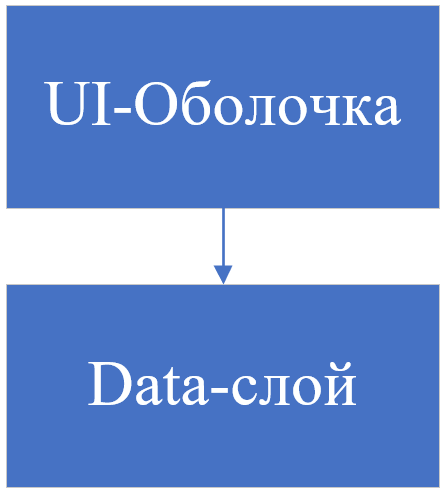


Рисунок 2.1. Архитектура мобильного приложения

UI оболочка

Роль слоя UI (или слоя представления) — отображать на экране данные приложения. Если данные меняются — из-за взаимодействия с пользователем (например, нажатия кнопки) или внешнего воздействия — UI должен обновиться и отразить изменения.

Слой данных

Слой данных в приложении содержит бизнес-логику *—* правила, по которым приложение создаёт, хранит и изменяет данные. Именно благодаря бизнес-логике приложение имеет ценность.

Слой UI состоит из двух частей и представлены на рисунке 2.2.:

* элементов UI, которые отображают данные на экране;
* экземпляров State Holder, которые хранят данные, открывают к ним доступ для UI и работают с логикой.

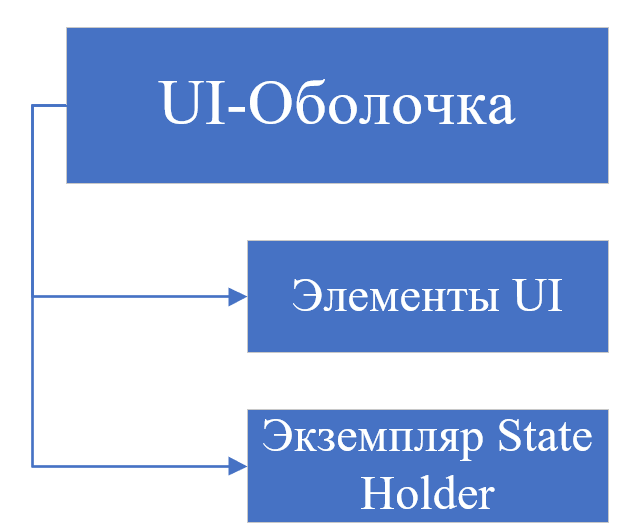


Рисунок 2.2. Диаграмма компонентов UI оболочки

Слой данных состоит из классов Репозитория*.*В каждом может содержаться множество классов DataSource представлен на рисунке 2.3.

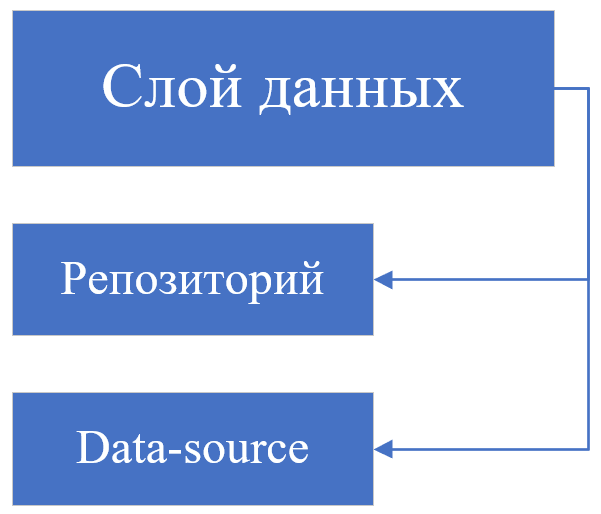


Рисунок 2.3. Архитектура слоя данных

Классы Repository:

* открывают доступ к данным для остальных элементов приложения;
* централизуют изменения в данных;
* разрешают конфликты между несколькими классами DataSource;
* абстрагируют классы DataSource от остальных элементов приложения;
* содержат бизнес-логику.

Каждый DataSource-класс должен отвечать за работу только с одним источником данных. Им может оказаться файл, сетевой источник или локальная база данных. В нашем случае источник — это локальная база данных.

DataSource-классы — связующее звено между приложением и системой для работы с данными.

* 1. **Моделирование основных сценариев системы**

В качестве основных сценариев системы, необходимо рассмотреть, как будет происходить процесс включения или отключения, а также автоматического управления. Для этого мы построим диаграмму последовательности.

Диаграмма последовательности – это диаграммы взаимодействия, в которых подробно описывается, как выполняются операции. Диаграммы последовательности описывают взаимодействие между классами с точки зрения обмена сообщениями во времени. Их также называют диаграммами событий. Диаграмма последовательности — это хороший способ визуализации и проверки различных сценариев выполнения. Они могут помочь предсказать, как будет вести себя система, и определить обязанности, которые могут потребоваться классу в процессе моделирования новой системы.

Актуальностью использования диаграмм последовательностей является удобность восприятия визуальных ассоциаций (картинок, символов и знаков), визуального представления (рисования карты), радиального мышления (создания логической и ассоциативной структуры, имеющей единый центр) достигается эффект глубокого понимания и отличного запоминания информации.

На данном этапе разработки приложения будут созданы две диаграммы последовательности:

* диаграмма последовательности по сценарию авторизации пользователя в мобильном приложении представлена на рисунке 2.4.;
* диаграмма деятельности взаимодействия пользователя со страницами Новости и Форум представлена на рисунке 2.5.

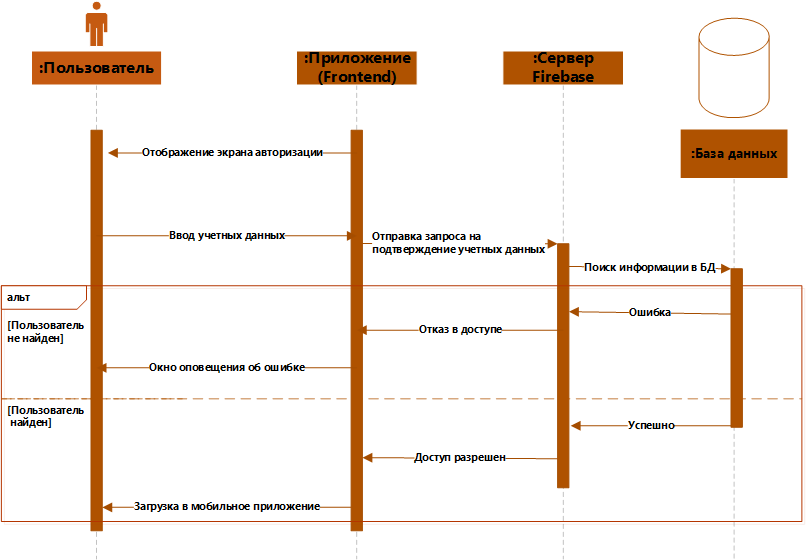


Рисунок 2.4. Диаграмма последовательности, сценарий «Авторизация в приложении»

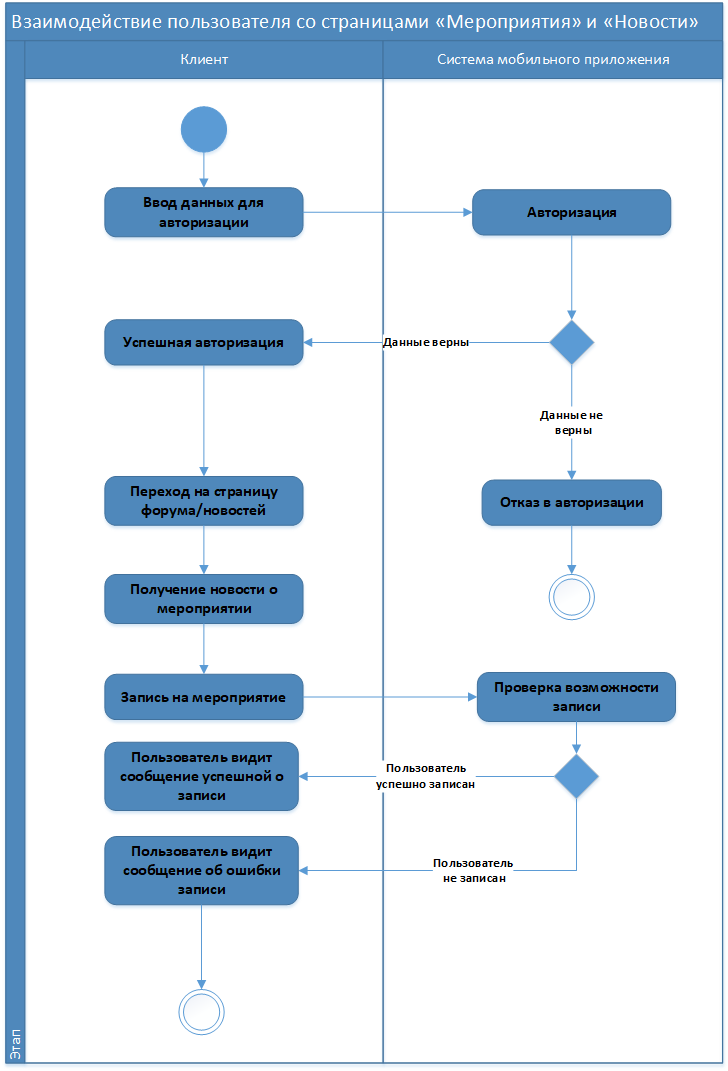


Рисунок 2.5. Диаграмма деятельности работы со страницами «Форум» и «Мероприятия»

* 1. **Проектирование графического интерфейса пользователя**

Графический интерфейс пользователя (GUI – graphic user interface) – это система интерактивных визуальных компонентов для компьютерного программного обеспечения. Графический пользовательский интерфейс отображает объекты, которые передают информацию и представляют действия, которые может предпринять пользователь. Объекты меняют цвет, размер или видимость, когда пользователь взаимодействует с ними.

Графический пользовательский интерфейс был впервые разработан Аланом Кеем, Дугласом Энгельбартом и группой других исследователей в 1981 году. Позже, 19 января 1983 года, Apple представила компьютер Lisa с графическим интерфейсом.

Хороший дизайн пользовательского интерфейса — это понимание потребностей вашего пользователя. В процессе построения дизайны, необходимо думать о задачах, которые пользователь будет выполнять с помощью продукта (потребности пользователя).

Для проектирования графического интерфейса пользователя будет в нашем случае будет использоваться Figma. Figma – это графический онлайн-редактор для совместной работы. В нём можно создать прототип сайта либо интерфейс приложения.

На рисунках 2.6–2.11. приведен макет страниц мобильного приложения.

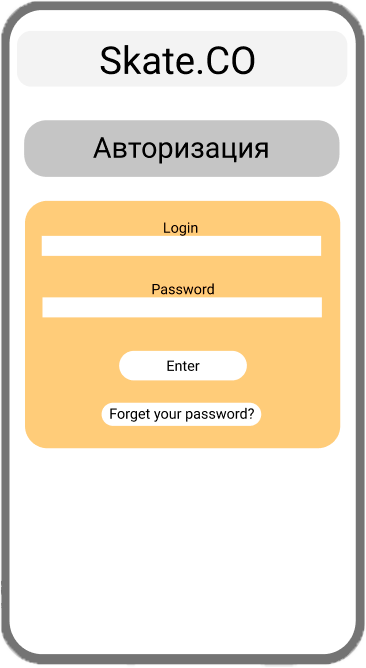


Рисунок 2.6. GUI первой страницы мобильного приложения

В процессе проектирования графического интерфейса пользователя, было принято решение о том, что неавторизованный пользователь увидит только первую страницу, где ему потребуется авторизироваться для дальнейшего доступа к приложению. Данное решение позволит увеличить безопасность мобильного приложения от нежелательных ботов[[1]](#footnote-1).

Новостная секция представляет простые изображения:

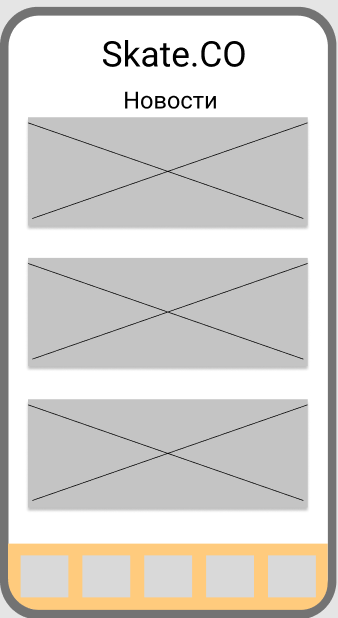


Рисунок 2.7. GUI авторизированного пользователя

В процессе проектирования графического интерфейса пользователя, было принято решение о том, что вся полезная информация будет показана пользователю в первую очередь, также было принято решение, что на нижней части приложения будет располагаться кнопки меню. Данное решение позволит облегчить процесс использования приложения, так как не будет необходимости переходить на какие-либо другие вкладки, когда для перехода между страницами приложения нужно нажать на кнопку в меню.

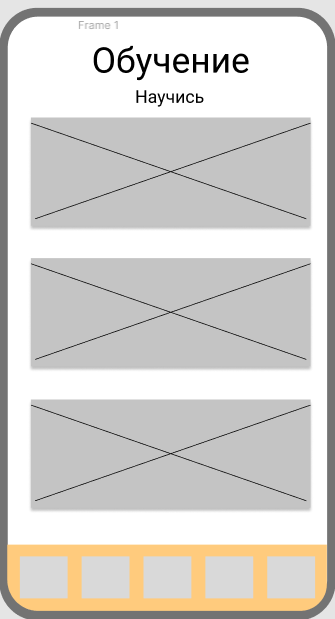


Рисунок 2.8. GUI страницы обучающего форума

Форум представляет собой блоки той или иной темы, к примеру, видео о «Как сделать Олли на скейтборде». Пользователи смогут не только научиться делать какой-либо трюк на скейтборде.

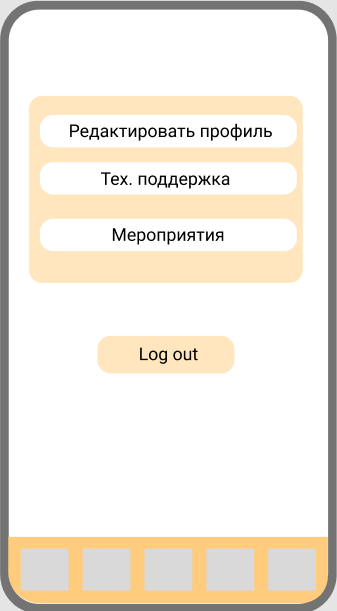


Рисунок 2.9. GUI страницы личного кабинета

Страница личного кабинета пользователя представляет собой основную панель управления, то есть, именно тут пользователь может редактировать информацию, обратиться в техническую поддержку, и проверить актуальные проходящие мероприятия.



Рисунок 2.10. GUI страницы мероприятий

Страница записи на мероприятия представляет собой блок с изображением мероприятия, при клике на изображение пользователь сможет более подробно ознакомиться с мероприятием.

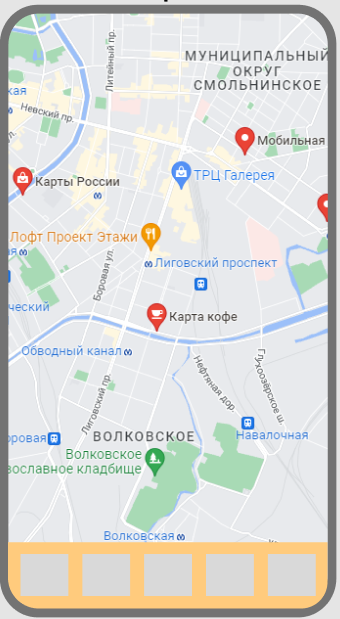


Рисунок 2.11. GUI страницы карты

Страница карты с отображением города Санкт-Петербург, на которой указаны локальные скейтпарки.

* 1. **Проектирование и разработка модели данных**

Модель данных можно рассматривать как блок-схему, которая иллюстрирует объекты данных, их атрибуты и отношения между объектами.

Обеспечивая визуальное представление наборов данных и их бизнес-контекста, он помогает точно определить потребности в информации для различных бизнес‑процессов. Затем он определяет характеристики элементов данных, которые будут включены в приложения и структуры базы данных или файловой системы, используемые для обработки, хранения и управления данными.

Имея детальный план для работы, моделирование данных оптимизирует разработку и снижает риск ошибок проектирования, которые потребуют пересмотра в процессе.

В базе данных должна храниться следующая информация:

О пользователях:

* ID — персональный, уникальный id пользователя;
* Email — личная почта, указанная пользователем при регистрации. Максимальная длина 50 символов;
* Login — логин, указанный пользователем. Максимальная длина 25 символов;
* Password — пароль, указанный пользователем. Максимальная длина 50 символов;
* Category id — id категории пользователя.

О категориях пользователей:

* Category id — задаваемый автоматически при регистрации, если пользователь является администратором, указывается другой id;
* Category description — включает дату создания. Максимальная длина 50 символов;
* Category name — роль пользователя в системе. Максимальная длина 25 символов.

ER-диаграмма базы данных представлена на рисунке 2.12. с её помощью можно выделить ключевые сущности и обозначить связи, которые могут устанавливаться между сущностями в ходе проектирования базы данных мобильного приложения.

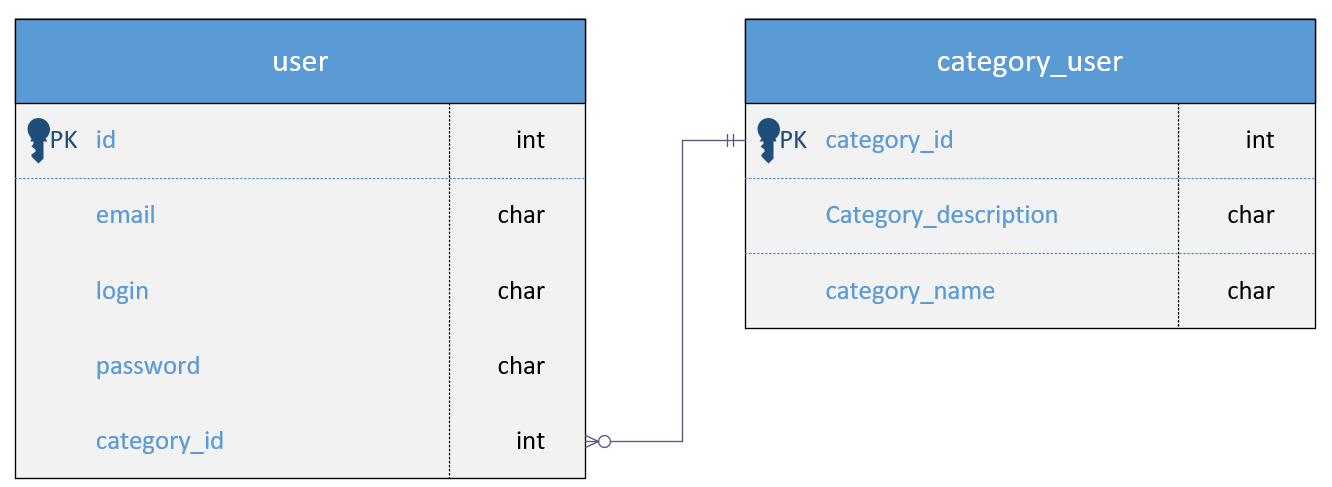


Рисунок. 2.12. ER-диаграмма пользователя

1. **РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА**

**3.1. Разработка мобильного приложения**

**3.1.1. Выбранные языки программирования и среда разработки**

Java — язык программирования общего назначения. Относится к объектно-ориентированным языкам программирования, к языкам с сильной типизацией. Java является объектно-ориентированным языком, относится к языкам программирования с сильной типизацией.

Android — операционная система для смартфонов, планшетов, электронных книг, цифровых проигрывателей, наручных часов, фитнес-браслетов, игровых приставок, ноутбуков, нетбуков, смартбуков, очков Google Glass, телевизоров, проекторов и других устройств.

JDK 18-й версии. JDK является одним из трех основных технологий, используемых в программировании на языке Java. JDK позволяет разработчикам создавать программы, которые могут выполняться и запускаться посредством JVM и JRE.

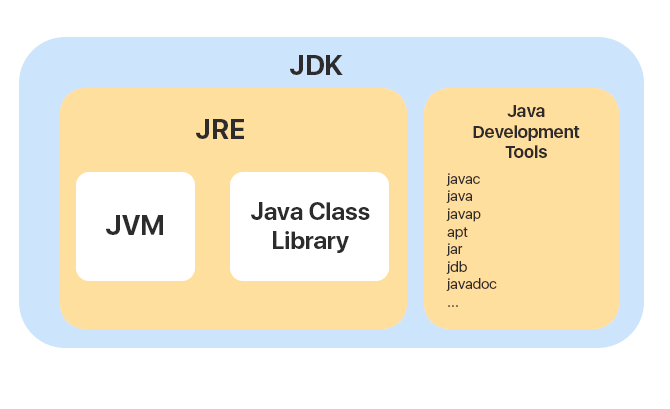


Рисунок 3.1. Что из себя представляет JDK

Android SDK — универсальное средство разработки мобильных приложений для операционной системы Android. Отличительной чертой от обычных редакторов для написания кодов является наличие широких функциональных возможностей, позволяющих запускать тестирование и отладку исходных кодов, оценивать работу приложения в режиме совместимости с различными версиями ОС Android и наблюдать результат в реальном времени (опционально). Поддерживает большое количество мобильных устройств.

AVD Manager — предназначен для создания и настройки виртуальных устройств Android (AVD), запускаемых в Android Emulator. Каждое AVD является конфигурацией эмулятора, имитирующей физическое устройство Android. Это позволяет запускать и тестировать приложение в различных конфигурациях, имитирующих различные физические устройства Android.

**3.1.2. Выбор и подключение базы данных**

Для мобильного приложения Skate.CO была выбрана облачная база данных на основе NoSQL Firebase.

Преимущества данной базы данных было обусловлено простотой установки в данный проект, долговременная поддержка от Google, и высокая степень защищённости.

Для начала работы с Firebase необходимо:

* аккаунт Google;
* пройти авторизацию на сайте Firebase
* загрузить файл-скрип google-services. json для подключения БД;
* имплементировать в файл build.gradle<:app> строки кода;
* синхронизировать Android Studio и Firebase;
* объявить наследование классов Firebase в основных классах кода;

Для имплементации в Android Studio данной базы данных нужно в файл конфигурации build.gradle<:app> вписать зависимости:

*Plugins {*

*id ‘com.android.application’*

*id ‘com.google.gms.google-services’*

*}*

*Dependencies {*

*implementation platform('com.google.firebase:firebase-bom:30.0.1')*

*implementation 'com.google.firebase:firebase-analytics'*

*}*

Выполненная синхронизация Android Studio и Firebase представлена на рисунке 3.2.

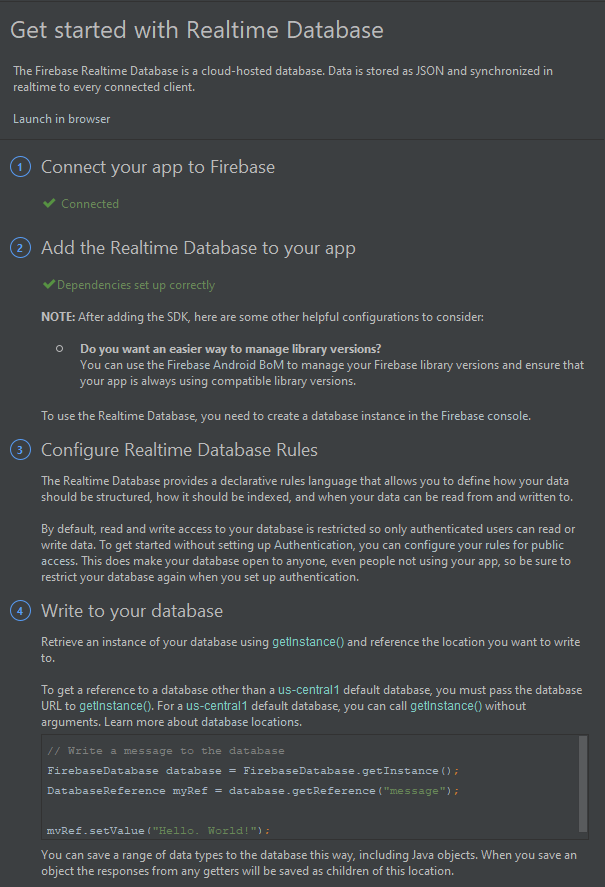


Рисунок 3.2. Выполненная синхронизация Android Studio и Firebase

Дальнейшая работа с данной базой данных реализована в основном классе мобильного приложения MainActivity:

*FirebaseAuth auth;  
FirebaseDatabase db;  
DatabaseReference users;*

*auth = FirebaseAuth.getInstance();  
db = FirebaseDatabase.getInstance();  
users = db.getReference("Users");*

Паттерн get.Instance() или же «синглтон» — это шаблон, который обеспечивает существование одного (реже более одного) экземпляра класса, без возможности прямого создания этого класса. Соответственно, полезен он именно в тех ситуациях, где такое поведение необходимо. Скажем, в приложении может быть только один менеджер работы с базой данных, а создание еще одного не только нецелесообразно, но и вредно.

**3.2. Разработка авторизации и регистрации в мобильном приложении**

При разработке авторизации и регистрации было принято решение, что пользователю при клике по кнопкам «Авторизация» и «Регистрация» будет выдано всплывающее окно поверх основного реализована данная функция добавлением библиотеки «Card View»:

*implementation 'androidx.cardview:cardview:1.0.0'*

У библиотеки «Card View» имеется функция «Card Elevation», которая позволяет настроить отображения всплывающих окон.

*app:cardElevation= «10dp»*

Страница регистрации и авторизации пользователя представлена на рисунке 3.3.

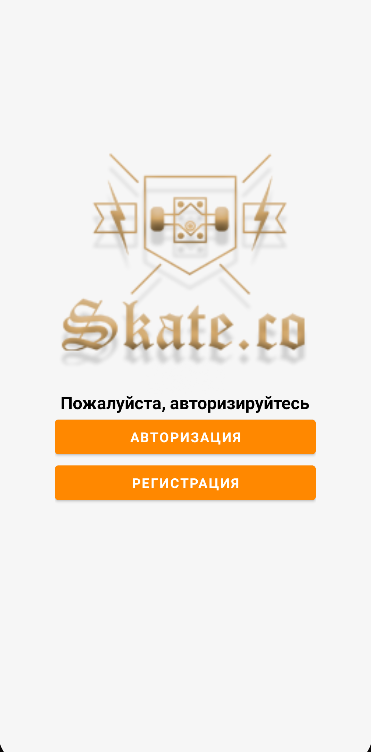


Рисунок 3.3. Главная страница

Для разработки стилизации полей ввода информации в мобильном приложении была подключена сторонняя библиотека.

Подключение библиотеки:

*implementation 'com.rengwuxian.materialedittext:library:2.1.4'*

Данная библиотека позволяет создать визуальный обработчик исключений в форме, т.е. задать условие «пароль должен быть от 8 до 32 символов». Данный функционал представлен на рисунке 3.4.

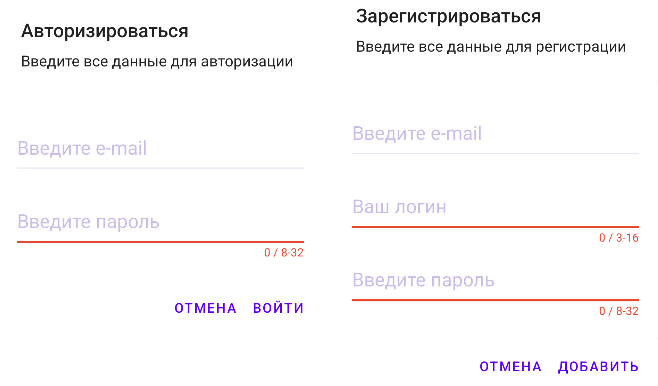


Рисунок 3.4. Стилизованные поля ввода информации

Условия с Рисунка 3.4. реализованы следующим кодом:

*<com.rengwuxian.materialedittext.MaterialEditText*

*android:layout\_width="match\_parent"*

*android:layout\_height="wrap\_content"*

*android:id="@+id/emailField"*

*android:hint="@string/EmailHint"*

*android:inputType="textEmailAddress"*

*android:text=""*

*android:textColor="#ffd88f"*

*android:textColorHint="#1e2129"*

*android:textSize="20sp"*

*app:met\_floatingLabel="highlight"*

*app:met\_baseColor="@color/design\_default\_color\_primary\_dark"*

*app:met\_primaryColor="#1e2129"*

*app:met\_singleLineEllipsis="true"*

*app:met\_minCharacters="3"*

*app:met\_maxCharacters="16"*

*/>*

На всех всплывающих формах при клике «Отмена» обрабатывается лямбда-выражение[[2]](#footnote-2), которая закрывает всплывающее окно:

*dialog.setNegativeButton("Отмена", (dialogInterface, which) -> dialogInterface.dismiss());*

При авторизации следующий код выполняет проверку введённых пользователем данных, затем сравнивает с данными в БД и, если все верно, вызовется обработчик события OnSuccessListener и перенаправит пользователя на другую страницу мобильного приложения методом startActivity:

*auth.signInWithEmailAndPassword(email.getText().toString(), password.getText().toString()).addOnSuccessListener(authResult -> {*

*startActivity(new Intent(MainActivity.this, NewsActivity.class));*

*finish();  
}).addOnFailureListener(e -> Snackbar.make(root, "Вы ввели неверные данные"+ "\n" + e.getMessage(),Snackbar.LENGTH\_SHORT).show());*

Помимо проверки данных, если произойдет ошибка, к примеру, пользователь ввел некорректные данные, то вызовется обработчик события OnFailureListener, который в свою очередь выведет на экран сообщение об ошибке.

При регистрации, если все данные пользователь ввёл верно вызывается обработчик OnSuccessListener, который передаёт в отдельный файл-класса User данные с полей. Регистрация пользователя выполнятся следующим кодом:

*auth.createUserWithEmailAndPassword(email.getText().toString(), password.getText().toString()).addOnSuccessListener(authResult -> {  
 User user = new User();  
 user.setLogin(login.getText().toString());  
 user.setEmail(email.getText().toString());  
 user.setPassword(password.getText().toString());*

*users.child(FirebaseAuth.getInstance().getCurrentUser().getUid()).setValue(user).addOnSuccessListener(unused -> Snackbar.make(root, "Вы успешно зарегистрировались", Snackbar.LENGTH\_SHORT).show());  
}).addOnFailureListener(e -> Snackbar.make(root, "Ошибка регистрации" + e.getMessage(),Snackbar.LENGTH\_SHORT).show());*

Файл User представляет собой обыкновенный конструктор и Getter и Setter:

*public class User {*

*private String login, email, password;*

*public User() {*

*}*

*public User(String login, String email, String password) {*

*this.login = login;*

*this.email = email;*

*this.password = password;*

*}*

*public String getLogin() {*

*return login;*

*}*

*public void setLogin(String login) {*

*this.login = login;*

*}*

*public String getEmail() {*

*return email;*

*}*

*public void setEmail(String email) {*

*this.email = email;*

*}*

*public String getPassword() {*

*return password;*

*}*

*public void setPassword(String password) {*

*this.password = password;*

*}*

*}*

**3.3. Выход из учётной записи пользователя**

Выход из учетной записи реализован следующим кодом:

*auth.signOut();*

*startActivity(new Intent(NewsActivity.this, MainActivity.class));*

**3.4. Запись на мероприятие**

В версии Android 3.0 реализованы классы Fragment (фрагменты). Они обеспечивают поддержку более гибких схем размещения элементов на экранах различных размеров на телефонах и планшетах.

Фрагменты можно считать модулями пользовательского интерфейса. Они позволяют разработчику разделить пользовательский интерфейс на удобные для многократного использования изолированные части, которые могут выполняться в отдельных действиях. Во время выполнения действия сами выбирают, какие фрагменты будут использоваться.

За запись на мероприятие отвечает следующий код:

*RecInButt.setOnClickListener(v -> RecordSuccess());*

*private void RecordSuccess() {*

*Snackbar.make(root, "Вы записались на мероприятие", Snackbar.LENGTH\_SHORT).show();*

*FragmentTransactio ft =getSupportFragmentManager().beginTransaction(); ft.replace(R.id.frameLayout, fragment);*

*ft.commit();*

*}*

Пояснение: при клике на кнопку «записаться», которая находится на странице task.xml происходит отображение на блоке Fragment, который находится на странице record.xml

**3.5. Отображение всплывающих окон**

В приложении реализованы всплывающие окна на многие изображения, за это отвечает следующий код:

*AlertDialog.Builder dialog = new AlertDialog.Builder(this);*

*LayoutInflater inflater = LayoutInflater.from(this);*

*View news\_image\_window = inflater.inflate(R.layout.news\_image\_window, null);*

*dialog.setView(news\_image\_window);*

*dialog.setNegativeButton("Отмена", (dialogInterface, which) ->*

*dialogInterface.dismiss());*

*dialog.show();*

**3.6. Отображение видео**

За отображение видео контейнеров отвечает следующий код:

*VideoView videoPlayer;*

*videoPlayer = findViewById(R.id.videoPlayer1);*

*Uri myVideoUri1= Uri.parse( "android.resource://" + getPackageName() + "/" + R.raw.ollie);*

*videoPlayer.setVideoURI(myVideoUri1);*

*MediaController mediaController = new MediaController(this);*

*videoPlayer.setMediaController(mediaController);*

*mediaController.setMediaPlayer(videoPlayer);*

Ссылка *videoPlayer = findViewById(R.id.videoPlayer1);* отвечает за поиск элемента в xml структуре.

Xml-файл:

*<VideoView*

*android:id="@+id/videoPlayer1"*

*android:layout\_width="328dp"*

*android:layout\_height="195dp"*

*android:layout\_marginTop="120dp"*

*app:layout\_constraintEnd\_toEndOf="parent"*

*app:layout\_constraintHorizontal\_bias="0.507"*

*app:layout\_constraintStart\_toStartOf="parent"*

*app:layout\_constraintTop\_toBottomOf="@+id/textView9"*

*/>*

За получение ссылки на видео, находящегося в проекте, отвечает следующие выражение *Uri myVideoUri1= Uri.parse( "android.resource://" + getPackageName() + "/" + R.raw.ollie);*

За отображение функциональных кнопок в видео отвечает следующий код:

*MediaController mediaController = new MediaController(this);*

*videoPlayer.setMediaController(mediaController);*

*mediaController.setMediaPlayer(videoPlayer);*

**3.7. Разработка карты в мобильном приложении**

Для мобильного приложения был выбран Map.Kit от Яндекс, была проведена регистрация в Яндекс Кабинет Разработчика для получения API ключа.

Ключ API – это уникальный идентификатор, который используется для аутентификации запросов, связанных с вашим проектом. Он нужен для учета использования API.

Добавление Map.Kit в проект:

В файл build.gradle<.project:> добавлены следующие записи:

*buildscript {*

*repositories {*

*mavenCentral()*

*maven {*

*url http://maven.google.com/*

*}*

*}*

В файл build.gradle<.app:> добавлены следующие строки зависимостей для полного стабильной работы:

*dependencies {*

*implementation 'com.yandex.android:maps.mobile:4.1.0-lite'*

*}*

За отображение карты отвечает map.xml:

*<?xml version="1.0" encoding="utf-8"?>*

*<RelativeLayout xmlns:android=http://schemas.android.com/apk/res/android*

*android:layout\_width="match\_parent"*

*android:layout\_height="match\_parent">*

*<com.yandex.mapkit.mapview.MapView*

*android:id="@+id/mapview"*

*android:layout\_width="match\_parent"*

*android:layout\_height="match\_parent"/>*

*</RelativeLayout>*

Программный код для отображения карты:

*public class MapActivity extends AppCompatActivity {*

*private final String MAPKIT\_API\_KEY = "b0dac91b-e31d-4da1-9988-e2e9bddc0548";*

*private static final Point TARGET\_LOCATION = new Point(59.945933, 30.320045);*

*private MapView mapView;*

*@Override*

*protected void onCreate(Bundle savedInstanceState) {*

*MapKitFactory.setApiKey(MAPKIT\_API\_KEY);*

*MapKitFactory.initialize(this);*

*setContentView(R.layout.activity\_map);*

*super.onCreate(savedInstanceState);*

*mapView = (MapView)findViewById(R.id.mapview);*

*mapView.getMap().move(*

*new CameraPosition(TARGET\_LOCATION, 14.0f, 0.0f, 0.0f),*

*new Animation(Animation.Type.SMOOTH, 5),*

*null);*

*}*

*@Override*

*protected void onStop() {*

*mapView.onStop();*

*MapKitFactory.getInstance().onStop();*

*super.onStop();  
 }*

*@Override  
 protected void onStart() {*

*super.onStart();*

*mapView.onStart();*

*MapKitFactory.getInstance().onStart();*

*}  
}*

Отображение карты фиксировано на город Санкт-Петербург за это отвечает переменная *TARGET\_LOCATION*

Функции *onStop() и onStart()* отвечают за закрытие карты при выходе со страницы карты. Иначе MapKit не сможет отобразить карту и остановить обработку карты, когда Activity с картой становится невидимым для пользователя

1. **ТЕСТИРОВАНИЕ ПРОГРАММНОГО ПРОДУКТА**

Тестирование программного обеспечения — это процесс проверки системы с целью выявления любых ошибок, пробелов или отсутствующих требований по сравнению с фактическими требованиями.

**4.1. История изменений**

В ходе создания программного продукта, было введено определенное количество изменений в коде. Большей части изменений была подвержена xml структура страниц мобильного приложения. Также изменения касались рефакторинга кода, для увеличения скорости его работы, проверки работы на других версиях OC Android.

В таблице 4.1 отображены все изменения в ходе разработки веб‑приложения.

Основные изменения в ходе разработки мобильного приложения

Таблица 4.1

| Дата | Автор | Описание изменений |
| --- | --- | --- |
| 1.05.2022 | Шандер Илья | Работа над макетом мобильного приложения |
| 2.05.2022 | Шандер Илья | Поиск и сбор технологий разработки |
| 2.05.2022 | Шандер Илья | Разработка по первому макету приложения |
| 3.05.2022 | Шандер Илья | Разработка по второму макету приложения |
| 4.05.2022 | Шандер Илья | Старт разработки backend кода |
| 5.05.2022 | Шандер Илья | Разработка регистрация по email и паролю |
| 5.05.2022 | Шандер Илья | Работа над ошибками приложения Android Studio |
| 6.05.2022 | Шандер Илья | Оптимизация кода |
| 6.05.2022 | Шандер Илья | Переход на NoSQL базу данных Firebase |
| 6.05.2022 | Шандер Илья | Настройка подключений к базе данных |
| 6.05.2022 | Шандер Илья | Проверка возможности добавлений данных при регистрации |
| 6.05.2022 | Шандер Илья | Разработка аунтификации |
| 7.05.2022 | Шандер Илья | Тестирование аунтификации и тестирование обработчика ошибок |
| 8.05.2022 | Шандер Илья | Разработка блока новостей |
| 8.05.2022 | Шандер Илья | Разработка кнопок меню |
| 8.05.2022 | Шандер Илья | Тестирование кнопок меню |
| 8.05.2022 | Шандер Илья | Разработка личного кабинета |
| 9.05.2022 | Шандер Илья | Разработка функционала записи на мероприятия и страницы мероприятий |
| 10.05.2022 | Шандер Илья | Разработка отображений успешной записи на мероприятие |
| 11.05.2022 | Шандер Илья | Тестирование личного кабинета, записи на мероприятия |
| 11.05.2022 | Шандер Илья | Разработка обучающего форума |
| 12.05.2022 | Шандер Илья | Тестирование обучающего форума |
| 12.05.2022 | Шандер Илья | Разработка и внедрение карты в мобильное приложение |
| 13.05.2022 | Шандер Илья | Полное тестирование приложения |
| 13.05.2022 | Шандер Илья | Рефакторинг кода |

**4.2. Терминология**

* тестирование стабильности – это проведение тестирования со средней нагрузкой в течение длительного времени. Выявляет проблемы связанные с утечками памяти и некорректными настройками ПО;
* нагрузочное тестирование – это тип теста производительности, который проверяет верхние пределы системы, тестируя ее при экстремальных нагрузках. При помощи данных тестов исследуется, как система ведет себя при интенсивных нагрузках и как она восстанавливается при возвращении к обычному использованию;
* тестирование стабильности – это проведение тестирования со средней нагрузкой в течение длительного времени. Выявляет проблемы связанные с утечками памяти и некорректными настройками ПО.

**4.3. Стратегия тестирования программного обеспечения**

Тестирование — это наблюдение за работой приложения в разных искусственно созданных ситуациях. Таким образом тестируют производительность, уровень защиты и удобство пользования программой.

Тестирование программного обеспечения можно разделить на два типа — функциональное тестирование и нефункциональное тестирование. Тестирование следует начинать как можно раньше, чтобы сократить затраты и время на доработку и создание программного обеспечения без ошибок.

В ходе тестирования необходимо выполнить каждый из следующих этапов:

* написать тест;
* добиться его безошибочной компиляции;
* запустить тест и убедиться, что он потерпел неудачу;
* добиться успешного выполнения теста;
* устранить дублирование.

Функциональное тестирование — это тип тестирования программного обеспечения, который используется для проверки функциональности программного приложения, работает ли функция в соответствии со спецификацией требований. При функциональном тестировании каждая функция тестируется путем предоставления значения, определения выходных данных и проверки фактического результата с ожидаемым значением. Функциональное тестирование, выполняемое как тестирование «черного ящика», которое проводится для подтверждения того, что функциональные возможности приложения или системы ведут себя так, как мы ожидаем. Целью такого тестирования является проверка работоспособности приложения.

Нефункциональное тестирование проверяет все аспекты, не охваченные функциональными тестами. Он включает в себя производительность, удобство использования, масштабируемость и надежность программного обеспечения.

Тестирование UX —это свойство продукта, определяющее насколько он удобен, понятен и легок в эксплуатации пользователями. Юзабилити-тестирование или UI/UX тестирование проводится при создании нового ПО и его доработке. Главная цель такого тестирования в поиске решений для создания наиболее дружелюбного интерфейса программного продукта и уменьшения времени на решение задач пользователей.

Стратегия тестирования — это план определения подхода к жизненному циклу тестирования программного обеспечения. Он помогает определить покрытие тестами и область тестирования. Это позволяет тестировщикам получить четкое представление о проекте в любой момент. Вероятность пропуска какой-либо тестовой активности очень мала при наличии надлежащей стратегии тестирования.

Во время тестирования основными целями будут являться тестирование пользовательского интерфейса, а также модуля автоматического управления для того, чтобы убедиться, что все требования были корректно реализованы, а также чтобы идентифицировать дефекты и убедиться, что они максимально выявлены еще до развертывания системы.

**4.4. Определение объектов тестирования**

В таблице 4.2 обозначено какие тесты необходимы для покрытия различных видов, типов и областей тестирования.

Необходимые тесты

Таблица 4.2

|  |  |
| --- | --- |
| Тесты | Объект тестирования |
| Функциональное тестирование | Формы ввода в модальных окнах; форма авторизации; добавление пользователя в БД |
| Нагрузочное тестирование | Firebase сервер |
| Тестирование стабильности | Несколько разных Android устройств, разные версии ОС Android |
| Тестирование интерфейса пользователя | Мобильное приложение |
| Тестирование безопасности | Мобильное приложение |
| Позитивные тесты | Мобильная страница пользователя |
| Негативные тесты | Формы ввода в модальных окнах; форма авторизации, регистрации |

**4.5. Архитектура тестируемой системы.**

Конфигурация Android-эмулятора и тестового стендов.

Характеристики систем приведены в таблицах 4.3. и 4.4.

Конфигурация Android эмулятора

Таблица 4.3.

| **Ресурсы** | **Детали** |
| --- | --- |
| **Pixel 2** | |
| CPU | Google Play Intel Atom (x86) |
| CPU Cores | 2 |
| RAM | 1536Мб |
| SD Card | 512Мб |
| API-Version | 30 |
| Android Version | 11 |
| Разрешение | 1920х1080 |

Конфигурация тестовых стендов

Таблица 4.4.

| Ресурсы | Детали |
| --- | --- |
| **Xiaomi MI A1** | |
| CPU | Soc Qualcomm Snapdragon 625 |
| GPU | Andreno 506 |
| GPU Cores | 8 ARM Cortex-A53 2,0Гц |
| RAM | 4Гб |
| SD | 64 Гб |
| Ресурсы | Детали |
| Android Version | 10.04 Root |
| Разрешение | 1920х1080 |

Продолжение таблицы 4.4.

|  |  |
| --- | --- |
| **Pixel 2** | |
| CPU | Google Play Intel Atom (x86) |
| CPU Cores | 2 |
| RAM | 1536Мб |
| SD Card | 512Мб |
| API-Version | 30 |
| Android Version | 11 |
| Разрешение | 1920х1080 |

**4.6. Описание процесса тестирования.**

В ходе описания процесса тестирования, необходимо отобразить тестовые сценарии с указанием конкретных тестовых наборов.

Чек‑лист тестирования и тест‑кейсы для веб‑приложения отображены в таблице 4.5.

Чек-лист тестирования и тест-кейсы

Таблица 4.5.

| Что проверяется | Проверка | Детализация/Ожидаемое | Результат/  Ошибка | Приоритет | Рез-т тестирования |
| --- | --- | --- | --- | --- | --- |
| Отображение на различных разрешениях | 800х600 | Отображение элементов мобильного приложения корректно | Отображается корректно | Средний | Пройден |
| Отображение на различных разрешениях | 1024x600 | Отображение элементов мобильного приложения корректно | Отображается корректно | Средний | Пройден |
| Отображение на различных разрешениях | 1024x768 | Отображение элементов мобильного приложения корректно | Отображается корректно | Средний | Пройден |

Продолжение таблицы 4.5.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Что проверяется | Проверка | Детализация/Ожидаемое | Результат/  Ошибка | Приоритет | Рез-т тестирования |
| Отображение на различных разрешениях | 1152x864 | Отображение элементов мобильного приложения корректно | Отображается корректно | Средний | Пройден |
| Отображение на различных разрешениях | 1280x800 | Отображение элементов мобильного приложения корректно | Отображается корректно | Средний | Пройден |
| Отображение на различных разрешениях | 1280x1024 | Отображение элементов мобильного приложения корректно | Отображается корректно | Средний | Пройден |
| Отображение на различных разрешениях | 1440x900 | Отображение элементов мобильного приложения корректно | Отображается корректно | Средний | Пройден |
| Отображение на различных разрешениях | 1680x1050 | Отображение элементов мобильного приложения корректно | Отображается корректно | Средний | Пройден |
| Отображение на различных разрешениях | 1920x1080 | Отображение элементов мобильного приложения корректно | Отображается корректно | Средний | Пройден |
| Регистрация пользователя | Вводятся заведомо корректные данные | Успешная регистрация нового пользователя | Регистрация успешна | Высокий | Пройден |

Продолжение таблицы 4.5.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Что проверяется | Проверка | Детализация/Ожидаемое | Результат/  Ошибка | Приоритет | Рез-т тестирования |
| Регистрация пользователя | Работоспособность кнопки «Отмена» | При клике на кнопку «Отмена» закрывается окно регистрации | Окно закрывается, работает корректно | Высокий | Пройден |
| Регистрация пользователя | Работоспособность кнопки «Добавить» | При вводе корректных данных и клике на кнопку «Добавить» успешно происходит добавление данных в БД и отображается всплывающая подсказка «Вы успешно зарегистрировались» | Регистрация успешна, отображения всплывающая подсказка «Вы успешно зарегистрировались» функционирует | Высокий | Пройден |
| Регистрация пользователя | Не вводятся никаких данных в поля ввода | Регистрации не произойдёт, будет объявлена и показана пользователю ошибка с пояснением | В БД не было добавлены некорректные данные, ошибка с пояснением отображается | Высокий | Пройден |
| Регистрация пользователя | Вводятся некорректные данные в поле вода e-mail | Регистрации не произойдёт, будет объявлена и показана пользователю ошибка с пояснением | В БД не было добавлены некорректные данные, ошибка с пояснением отображается | Высокий | Пройден |
| Регистрация пользователя | Вводятся некорректные данные в поля Login | Регистрации не произойдёт, будет объявлена и показана пользователю ошибка с пояснением | В БД не было добавлены некорректные данные, ошибка с пояснением отображается | Высокий | Пройден |
| Регистрация пользователя | Вводится слишком короткий пароль (менее 8 символов) | Регистрации не произойдёт, будет объявлена и показана пользователю ошибка с пояснением | В БД не было добавлены некорректные данные, ошибка с пояснением отображается | Высокий | Пройден |
| Регистрация пользователя | Вводится слишком длинный пароль более 32 символов | Регистрации не произойдёт, будет объявлена и показана пользователю ошибка с пояснением | В БД не было добавлены некорректные данные, ошибка с пояснением отображается | Высокий | Пройден |

Продолжение таблицы 4.5.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Что проверяется | Проверка | Детализация/Ожидаемое | Результат/  Ошибка | Приоритет | Рез-т тестирования |
| Авторизация пользователя | Ввод заведомо корректных данных в поля | Авторизация успешна, авторизированного пользователя перенаправляет на другую страницу приложения | Авторизация успешна, пользователь перенаправлен на следующую страницу | Высокий | Пройден |
| Авторизация пользователя | Вводятся некорректные данные в поле вода e-mail | Авторизация не произойдёт, будет объявлена и показана пользователю ошибка с пояснением | Авторизация не пройдена, ошибка с пояснением отображается | Высокий | Пройден |
| Авторизация пользователя | Вводится слишком короткий пароль (менее 8 символов) | Авторизация не произойдёт, будет объявлена и показана пользователю ошибка с пояснением | Авторизация не пройдена, ошибка с пояснением отображается | Высокий | Пройден |
| Авторизация пользователя | Вводится слишком длинный пароль более 32 символов | Авторизация не произойдёт, будет объявлена и показана пользователю ошибка с пояснением | Авторизация не пройдена, ошибка с пояснением отображается | Высокий | Пройден |
| Авторизация пользователя | Работоспособность кнопки «Отмена» | При клике на кнопку «Отмена» закрывается окно авторизации | Окно закрывается, работает корректно | Высокий | Пройден |
| Возможность выйти из учетной записи | Возможность выйти из учетной записи | При клике на кнопку «Logout» происходит выход из учетной записи пользователя и переадресация на страницу с авторизацией | Выход из учетной записи произведён, переадресация на страницу авторизации успешна | Высокий | Пройден |
| Корректность отображения элементов на странице Новости | Все элементы отображаются корректно | Переход при успешной авторизации на страницу новостей и успешное её отображение | Отображается корректно. | Высокий | Пройден |

Продолжение таблицы 4.5.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Что проверяется | Проверка | Детализация/Ожидаемое | Результат/  Ошибка | Приоритет | Рез-т тестирования |
| Отображение детальной информации о новости при клике на изображение | Проверка отображения детальной информации о новости при клике на изображение | При клике на изображение пользователь обязан успешно прочитать детальную информацию о новости | При клике отображается всплывающее окно с детальной информацией о новости | Высокий | Пройден |
| Функциональность кнопок меню | Успешности перехода по страницам | При клике по кнопкам меню должен происходить переход на соответствующую страницу | При клике по кнопкам в меню происходит переход на другую страницу | Высокий | Пройден |
| Индикация в меню при переходе на другую страницу | Проверка индикации в меню на какой странице находится пользователь | При переходе по страницам в меню должно менять индикацию кнопки на другой оттенок, для визуального понимания нахождения на странице | Индикация функционирует | Высокий | Пройден |
| Возможность перехода со страницы новостей на страницу мероприятий | Возможности перехода со страницы новостей на страницу мероприятий | Успешный переход со страницы на страницу | При клике в меню на кнопку соответствующей страницы мероприятие происходит переход на страницу. | Высокий | Пройден |
| Отображение детальной информации о новости при клике на изображение | Проверка отображения детальной информации при клике на изображение мероприятия | При клике на изображение пользователь обязан успешно прочитать детальную информацию о мероприятии | При клике отображается всплывающее окно с детальной информацией о мероприятии | Высокий | Пройден |

Продолжение таблицы 4.5.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Что проверяется | Проверка | Детализация/Ожидаемое | Результат/  Ошибка | Приоритет | Рез-т тестирования |
| Возможность записаться на мероприятие | При клике на кнопку «записаться» происходит запись на мероприятие | При клике на кнопку «записаться» происходит запись на мероприятие | При клике на кнопку «записаться» происходит запись на мероприятие | Высокий | Пройден |
| Отображение всплывающей подсказки о успешной записи на мероприятие | Всплывающая подсказка | При клике на кнопку «записаться» происходит визуальное отображение всплывающей подсказки об успешности записи | При клике на кнопку отображается подсказка | Низкий | Пройден |
| Функциональность кнопок меню | Переход со страницы новости на страницу обучающего форума | При клике по кнопкам меню должен происходить переход на соответствующую страницу | При клике по кнопкам в меню происходит переход на другую страницу | Высокий | Пройден |
| Функциональность кнопок меню | Переход со страницы мероприятий на страницу обучающего форума | При клике по кнопкам меню должен происходить переход на соответствующую страницу | При клике по кнопкам в меню происходит переход на другую страницу | Высокий | Пройден |
| Корректность отображения элементов на странице обучающего форума | Корректность отображения при переходе с другой страницы на страницу форума | Корректность отображения при переходе с другой страницы на страницу форума | Корректное отображения при переходе с другой страницы на страницу форума | Высокий | Пройден |

Продолжение таблицы 4.5.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Что проверяется | Проверка | Детализация/Ожидаемое | Результат/  Ошибка | Приоритет | Рез-т тестирования |
| Возможность воспроизведения видео при клике на него | Возможность воспроизведения видео при клике на него | При клике по видео происходит успешное воспроизведение видео | При клике по видео происходит успешное воспроизведение видео | Высокий | Пройден |
| Отображение мультимедийных кнопок под видео | Отображения мультимедийных кнопок под видео | Кнопки отображаются | Кнопки отображаются | Высокий | Пройден |
| Возможность паузы видео | Возможности поставить видео на паузу | Видео успешно приостановлено | Видео успешно приостановлено | Высокий | Пройден |
| Возможность возобновить видео после паузы | Возможность возобновить видео после паузы | Воспроизведение видео после паузы успешно | Воспроизведение видео после паузы успешно | Высокий | Пройден |
| Функциональность кнопок меню | Переход со страницы обучающего форума на другие и обратно | При клике по кнопкам меню должен происходить переход на соответствующую страницу и обратно | При клике по кнопкам меню должен происходить переход на соответствующую страницу | Высокий | Пройден |
| Корректность отображения элементов страницы | Корректность отображения элементов страницы | Корректное отображения элементов страницы | Корректное отображения элементов страницы | Высокий | Пройден |

Продолжение таблицы 4.5.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Что проверяется | Проверка | Детализация/Ожидаемое | Результат/  Ошибка | Приоритет | Рез-т тестирования |
| Возможность перехода на страницу просмотра записей | При клике по кнопке «просмотреть запись» должно перенаправить на страницу просмотра | Возможность перехода на страницу просмотра записей | Корректный переход | Высокий | Пройден |
| Проверка события, что пользователь не записался на мероприятие | Если записи нет – нет отображения записей | Записей нет. Вывод сообщения об необходимости записи | Записей нет, выводится на страницу сообщения об необходимости записи | Высокий | Пройден |
| Проверка события, что пользователь записался на мероприятие | При успешной записи на странице мероприятий, должно быть отображение в личном кабинете | После записи корректное отображение краткой информации об мероприятии на странице пользователя | Корректное отображение мероприятия | Высокий | Пройден |
| Возможность отписаться от мероприятия | Возможность отписаться от мероприятия | Возможность отписаться от мероприятия | При клике по кнопке «отменить» происходит отписка от мероприятия. | Высокий | Пройден |
| Отображение всплывающей подсказки об отмене записи | Отображение всплывающей подсказки об отмене записи | Отображение всплывающей подсказки об отмене записи | Отображение успешно | Низкий | Пройден |

Продолжение таблицы 4.5.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Что проверяется | Проверка | Детализация/Ожидаемое | Результат/  Ошибка | Приоритет | Рез-т тестирования |
| Возможность перехода на страницу просмотра карты | Возможность перехода на страницу просмотра карты | Возможность перехода на страницу просмотра карты | Переход успешен | Высокий | Пройден |
| Возможность запуска приложения на версии Android 13 | Запуска приложения на Android 13 | Запуск тестового стенда на эмуляторе с версией ОС Android 13 | Запуск успешен | Средний | Пройден |
| Android 12 | Запуска приложения на Android 12 | Запуск тестового стенда на эмуляторе с версией ОС Android 12 | Запуск успешен | Средний | Пройден |
| Android 11 | Запуска приложения на Android 11 | Запуск тестового стенда на эмуляторе с версией ОС Android 11 | Запуск успешен | Средний | Пройден |
| Android 10 | Запуска приложения на Android 10 | Запуск тестового стенда на эмуляторе с версией ОС Android 10 | Запуск успешен | Средний | Пройден |
| Android 9 | Запуска приложения на Android 9 | Запуск тестового стенда на эмуляторе с версией ОС Android 9 | Запуск успешен | Средний | Пройден |
| Android 8 | Запуска приложения на Android 8 | Запуск тестового стенда на эмуляторе с версией ОС Android 8 | Запуск успешен | Средний | Пройден |
| Android 7 | Запуска приложения на Android 7 | Запуск тестового стенда на эмуляторе с версией ОС Android 7 | Запуск успешен | Средний | Пройден |

1. **РАЗРАБОТКА ТЕХНОЛОГИЧЕСКОЙ ДОКУМЕНТАЦИИ**

**5.1. Руководство пользователя**

* назначение мобильного приложения:

Мобильное приложение компании Skate.CO предназначена для пользователей, занимающихся хобби, а именно скейтбордингом.

В данном приложении пользователь после недолгой регистрации и авторизации сможет узнать свежие новости в мире скейтбординга, узнать о новых товарах, посмотреть карту города с отмеченными скейт-площадками и записаться на скейт-мероприятия, проходящие в городе.

* подготовка мобильного приложения к использованию:

Для обеспечения корректного функционирования мобильного приложения необходим доступ в интернет.

* описание использования мобильного приложения:

При первом посещении мобильного приложения пользователь должен авторизоваться в нем. Первая страница мобильного приложения представлена на рисунке 5.1.

Если же пользователь не зарегистрирован, то необходимо воспользоваться кнопкой «регистрация». Функционал регистрации отображен на рисунке 5.2.

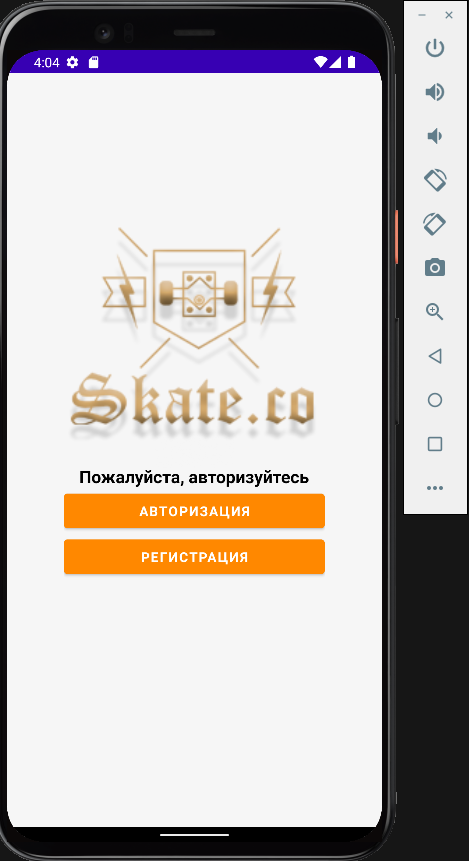


Рисунок 5.1. Первая страница мобильного приложения

На первой странице пользователь видит только логотип, надпись «пожалуйста, авторизуйтесь, две кнопки, при нажатии на которые откроются окна (рисунок 5.2. и 5.3.).

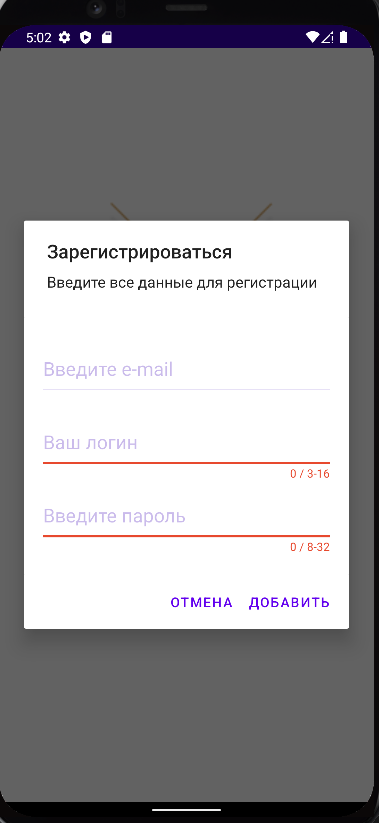


Рисунок 5.2. Регистрация

Как видно из рисунка, пользователь должен вписать в поля «Введите email», «Ваш логин» и «Введите пароль» свои данные. Также пользователь не сможет ввести меньшую длину логина или пароля, чем указана на подсказке. Если же пользователь введёт неверные данные, то приложение выведет на экран ошибку с текстом пояснения.

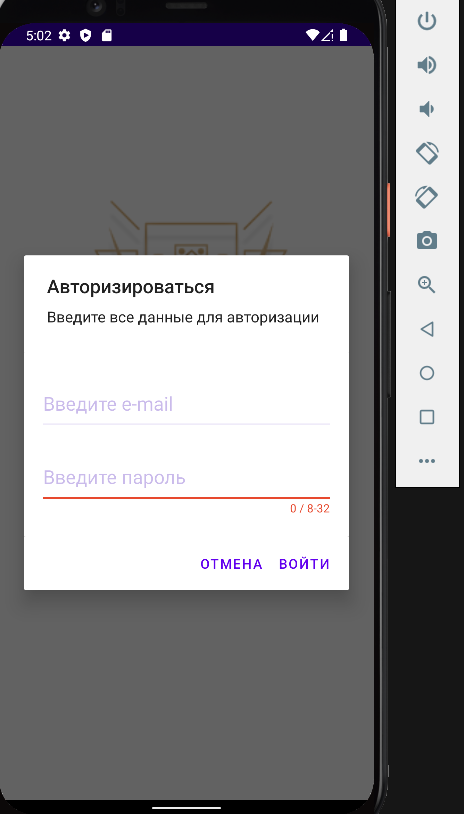


Рисунок 5.3. Авторизация

На странице авторизации, пользователь должен ввести свой email и пароль. Если пользователь введет пароль менее 8 символов или более 32-х, то система уведомит его об этом.

После авторизации в приложении, пользователь может просмотреть новости, нажимая на картинки (рисунки 5.4. - 5.5).

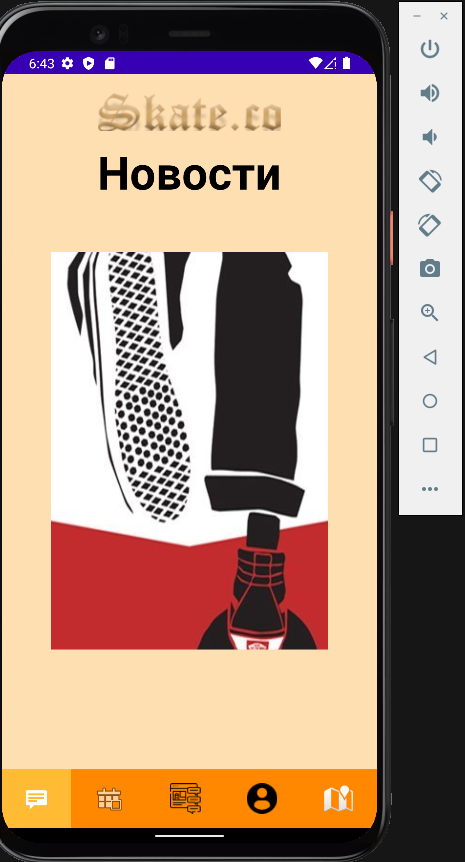


Рисунок 5.4. Страница авторизованного пользователя

Для навигации по мобильному приложению предусмотрено меню находящиеся в нижней части приложения. Для упрощенного понимания на какой странице находится пользователь, индикация в меню меняет цвет.

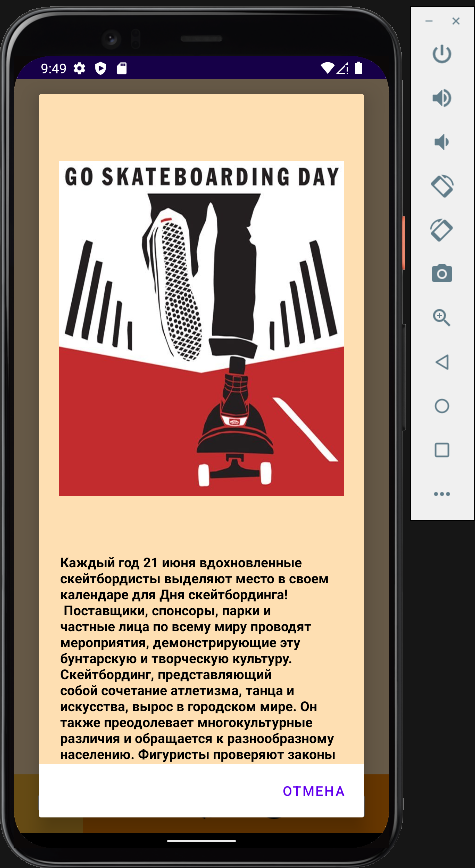


Рисунок 5.5. Отображение новости

При клике на новость, пользователь может более подробно с ней ознакомиться. При клике на кнопку «Отмена» - новость закроется.

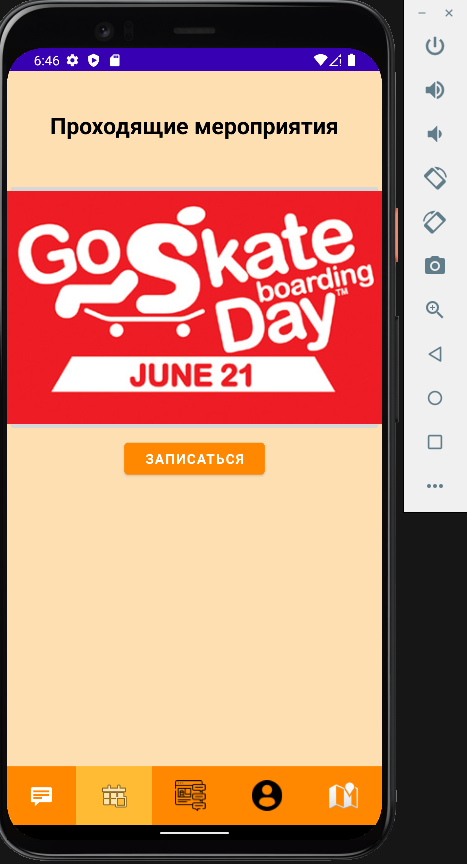


Рисунок 5.6. Окно мероприятий

В данном окне пользователь может просмотреть проходящие мероприятия.

При клике на изображение пользователю будет показана более подробная информация о мероприятии (рисунок 5.7).



Рисунок 5.7. Подробная информация о мероприятии

При клике на кнопку «записаться» пользователь запишется на мероприятие, о чем будет выведено сообщение на экран (рисунок 5.8.).

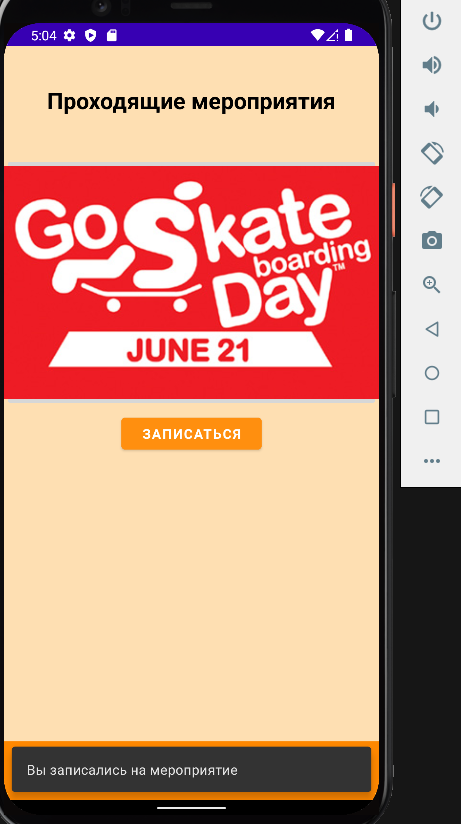


Рисунок 5.8. Отображение записи

При клике в меню по кнопке профиль, пользователь будет перенаправлен на страницу личного кабинета (рисунок 5.9.)

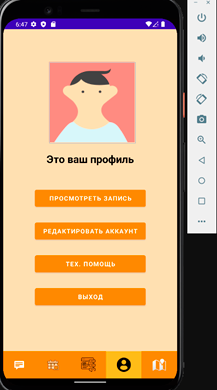


Рисунок 5.9. Профиль пользователя

На данной странице пользователь может просмотреть активные записи на мероприятия (рисунок 5.10-12.), редактировать свой аккаунт (рисунок 5.13.), обратится в техническую поддержку (рисунок 5.14.) и выполнить выход из своей учетной записи.

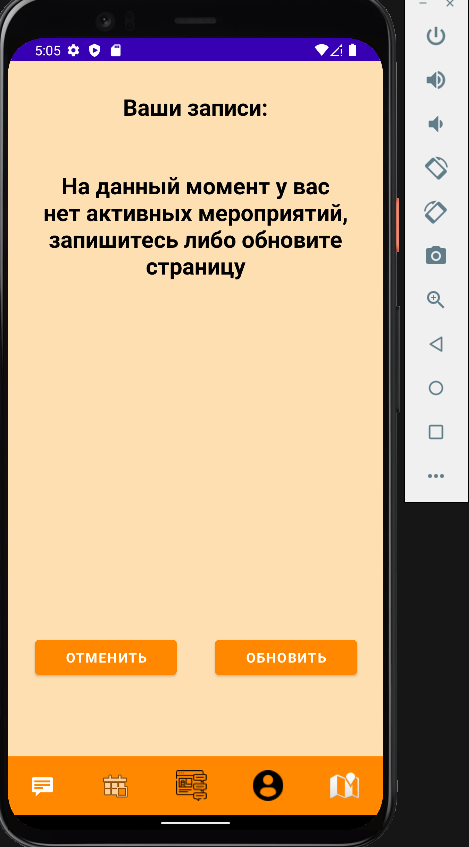


Рисунок 5.10. записи на мероприятия

После записи на мероприятие (рисунок 5.8.) активная запись будет показана пользователю, однако, для её успешного отображения, нужно нажать на кнопку «обновить».

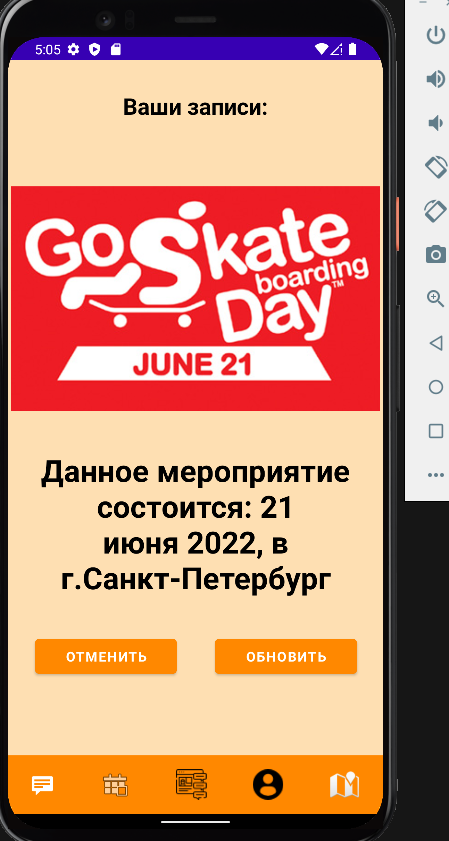


Рисунок 5.11. Отображение записи на мероприятие

Если пользователь передумал о посещении мероприятия, то он может отписаться от него, нажав соответствующую кнопку (рисунок 5.12.).

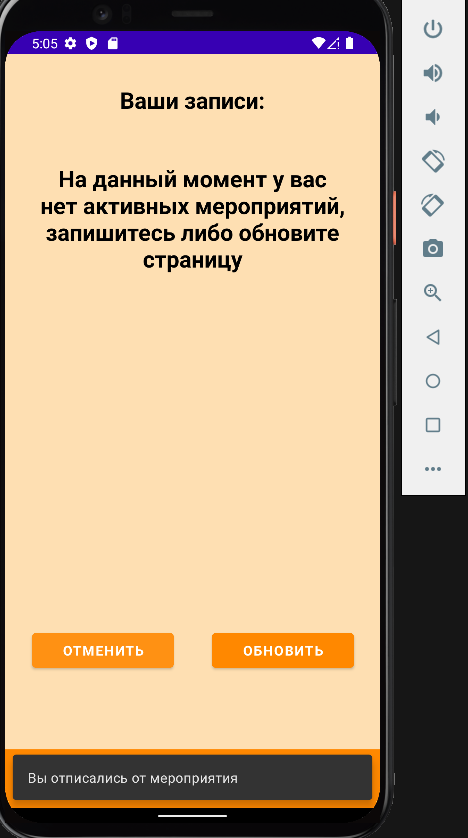


Рисунок 5.12. Отписка от мероприятия

Перейдя на страницу обучающего форума, пользователь сможет просмотреть видеофайлы кликнув по ним (рисунок 5.13.)

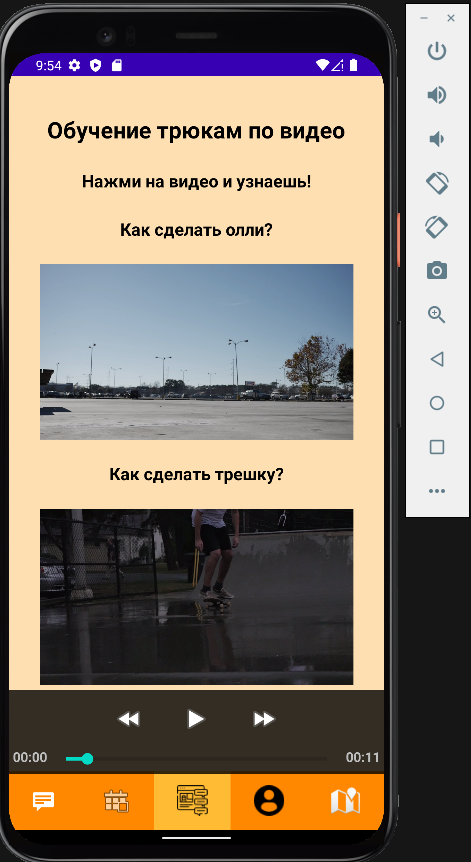


Рисунок 5.13. Обучающий форум

На странице «Карта» (рисунок 5.14.) пользователь может просмотреть карту города

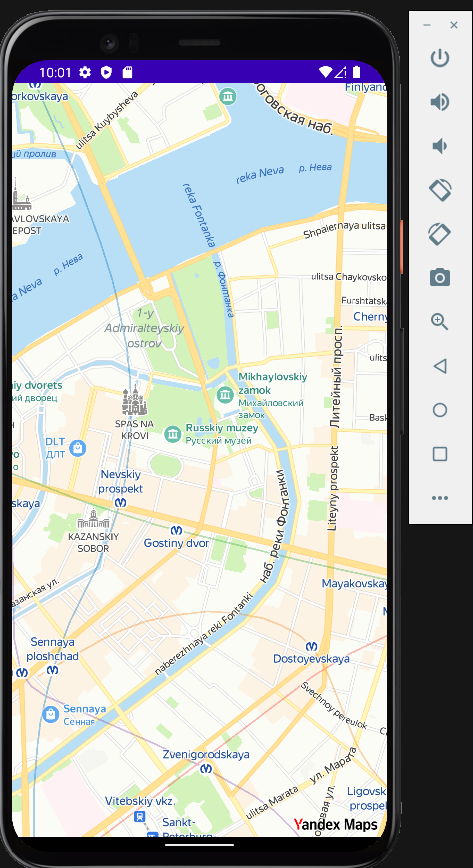


Рисунок 5.14. Карта

**5.2. Руководство администратора**

Администратор приложения может авторизоваться в Firebase для редактирования и удаления информации пользователей.

Для этого администратору нужно авторизоваться в личном кабинете Firebase, перейти в RealtimeDatabase, открыть базу данных и выбрать таблицу users.

1. **МЕРОПРИЯТИЯ ПО ТЕХНИКЕ БЕЗОПАСТНОСТИ И ОХРАНЕ ТРУДА**

# **6.1. Характеристика объекта проектирования и условий его эксплуатации**

**Технические данные проектного устройства**

Целью проекта является организация беспроводной сети в Санкт-Петербургской государственной академии ветеринарной медицины, для предоставления выхода в интернет с высокой скоростью.

Согласно ГОСТ 12.1.005-88 ССБТ “Общие санитарно-гигиенические требования к воздуху рабочей зоны” нужно создать нормальный климатические условия для работников, которые будут обслуживать сеть.

В помещении технического персонала, в холодный период года должны соблюдаться оптимальные величины температуры воздуха 22-24°С (допустимая 18-26°С), его относительной влажности 60-40% и скорости движения (не более 0,1 м/с).

В теплый период года должны соблюдаться оптимальные величины температуры воздуха 23-25°С (допустимая 20-30°С), его относительной влажности 60-40% и скорость движения воздуха 0,1 м/с и допустимая 0,1-0,2 м/с

Так как технический персонал большую часть работы проводит за ПК, для них требуется спроектированное рабочее место. Необходимы стул, на котором они смогут регулировать высоту для комфортной работы, подставка для ног с регулировкой высоты, монитор рекомендуется располагать на расстоянии 60-100 см. Сам персонал должен соблюдать правильную осанку при работе за компьютером для исключения проблем со спиной.

Также при проектировании рабочего места следует помнить, что физиологически не оправдана фиксированная рабочая поза, т.к. она может привести к профессиональным заболеваниям. Поэтому целесообразно предусматривать возможность работы стоя и сидя.

**Условия эксплуатации проектного объекта**

Помещения, где используются ПК, должны иметь естественное и искусственное освещение. Расположение рабочих мест за мониторами для взрослых пользователей в подвальных помещениях не допускается. Естественное освещение обеспечивается через оконные проемы с коэффициентом естественного освещения КЕО не ниже 1,2% в зонах с устойчивым снежным покровом и не ниже 1,5% на остальной территории.

Площадь на одно рабочее место с компьютером для взрослых пользователей должна составлять не менее 4,5 м2, если монитор жидкокристаллический, и не менее 6 м2 для ЭЛТ-монитора. Объем на одно рабочее место - не менее 20 м3.

Помещения с компьютерами должны оборудоваться системами отопления, кондиционирования воздуха или эффективной приточно-вытяжной вентиляцией.

Для внутренней отделки интерьера помещений с компьютерами должны использоваться диффузно-отражающие материалы с коэффициентом отражения для потолка — 0,7-0,8; для стен — 0,5-0,6; для пола — 0,3-0,5.

Поверхность пола в помещениях эксплуатации компьютеров должна быть ровной, без выбоин, нескользкой, удобной для очистки и влажной уборки, обладать антистатическими свойствами.

В помещении должны находиться аптечка первой медицинской помощи, углекислотный огнетушитель для тушения пожара.

Работая с ПК, оператор подвергается воздействию следующих психофизических факторов:

* умственное перенапряжение;
* перенапряжение зрительных и слуховых анализаторов;
* монотонность труда;
* эмоциональные перегрузки.

Кроме того, работа с ПК связана с воздействием таких вредных и опасных факторов, как:

* повышенный уровень шума;
* повышенная температура внешней среды;
* недостаточная освещенность рабочей зоны;
* электрический ток;
* статическое электричество;
* электромагнитное излучение и др.

Влияние выше приведенных факторов приводит к утомлению и снижению работоспособности. Появление и развитие утомления вызывает изменения в центральной нервной системе человека. В результате длительного нахождения человека в зоне комбинированного воздействия вредных различных и опасных факторов может привести к профессиональному заболеванию.

**Требования к обслуживающему персоналу**

Технический персонал будет состоять их двух сотрудников: технический специалист и диспетчер поддержки и мониторинга беспроводной сети. Образование у сотрудников должно быть среднее профессиональное или выше. Также должна быть третья квалификационная группа по электробезопасности.

Особые требования к персоналу:

* знание и умение работать с электронным оборудованием;
* знание пожарной безопасности;
* желателен опыт работы.

# **6.2. Мероприятия по технике безопасности**

**Организационно- технические мероприятия по охране труда**

При работе за ПК требуется соблюдать правила:

* правильно сидеть для избежания переутомления и проблем со спиной;
* держать глаза на расстоянии 60-100 см. от монитора;
* не хранить воду рядом с электрическими приборами;
* не допускать большое скопление пыли на приборах и в помещении.

При работе с ПК защитные средства не требуются, кроме случаев, если у сотрудника плохое зрение и ему нужно носить очки.

В помещении или рядом с помещением должна быть схема здания, если понадобится пожарная эвакуация.

**Требования электробезопасности при работе с ПК**

Необходимо постоянно следить на рабочем месте за исправным состоянием электропроводки, выключателей, штепсельных розеток, при помощи которых оборудование включается в сеть, и заземления. При обнаружении неисправности немедленно обесточить электрооборудование, оповестить администрацию. Продолжение работы возможно только после устранения неисправности.

Во избежание повреждения изоляции проводов и возникновения коротких замыканий не разрешается:

* вешать что-либо на провода;
* закрашивать и белить шнуры и провода;
* закладывать провода и шнуры за газовые и водопроводные трубы, за батареи отопительной системы;
* выдергивать штепсельную вилку из розетки за шнур, усилие должно быть приложено к корпусу вилки.

Для исключения поражения электрическим током запрещается:

* часто включать и выключать компьютер без необходимости;
* прикасаться к экрану и к тыльной стороне блоков компьютера;
* работать на средствах вычислительной техники и периферийном оборудовании мокрыми руками;
* работать на средствах вычислительной техники и периферийном оборудовании, имеющих нарушения целостности корпуса, нарушения изоляции проводов, неисправную индикацию включения питания, с признаками электрического напряжения на корпусе;
* класть на средства вычислительной техники и периферийном оборудовании посторонние предметы.

Запрещается:

* под напряжением очищать от пыли и загрязнения электрооборудование;
* проверять работоспособность электрооборудования в неприспособленных для эксплуатации помещениях с токопроводящими полами, сырых, не позволяющих заземлить доступные металлические части.

Ремонт электроаппаратуры производится только специалистами-техниками с соблюдением необходимых технических требований.

Во избежание поражения электрическим током, при пользовании электроприборами нельзя касаться одновременно каких-либо трубопроводов, батарей отопления, металлических конструкций, соединенных с землей.

При пользовании электроэнергией в сырых помещениях соблюдать особую осторожность. При обнаружении оборвавшегося провода необходимо немедленно сообщить об этом администрации, принять меры по исключению контакта с ним людей.

Во всех случаях поражения человека электрическим током немедленно вызывают врача. До прибытия врача нужно, не теряя времени, приступить к оказанию первой помощи пострадавшему. Искусственное дыхание пораженному электрическим током производится вплоть до прибытия врача.

**Требования защиты от излучения**

Источниками электромагнитного излучения, при работе пользователя за ПК, являются компоненты внутри корпуса и дисплея. Электромагнитное излучение нормируется согласно ГОСТ 12.1.006-84 «ССБТ. Электромагнитные поля радиочастот. Допустимые уровни на рабочих местах и требования к проведению контроля». Согласно СанПиН 2.2.2./2.5.1340-03 «Гигиенические требования к видеодисплейным терминалам, персональным ЭВМ и организации работы» конструкция ПЭВМ должна обеспечивать мощность экспозиционной дозы рентгеновского излучения в любой точке на расстоянии 0,05 м от экрана при любых положениях регулировочных устройств не должна превышать 1 мкЗв/час (100 мкР/час).

Допустимые значения неионизирующих электромагнитных излучений:

* напряженность электромагнитного поля по электрической составляющей на расстоянии 50 см от поверхности видеомонитора не более 10 В/м или 0,3 А/м;
* напряженность электростатического поля 20 кВ/м.

Согласно СанПиН 2.2.4/2.1.8.055-96 «Электромагнитные излучения радиочастотного диапазона», для рабочего места оператора ЭВМ или программиста при восьмичасовой рабочей неделе существуют следующие ограничения на уровень электромагнитного излучения: ‒ напряженность магнитного поля не должна превышать 500 В/м; ‒ напряженность электрического поля не должна превышать 50 А/м.

**Требования защиты от шума**

Уровни шума на рабочих местах пользователей персональных компьютеров не должны превышать значений, установленных СанПиН 2.2.4/2.1.8.562-96 «Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки» и составляют не более 50 дБА.

На рабочих местах в помещениях для размещения шумных агрегатов уровень шума не должен превышать 75 дБА.

Снизить уровень шума в помещениях можно использованием звукопоглощающих материалов с максимальными коэффициентами звукопоглощения в области частот 63-8000 Гц для отделки стен и потолка помещений. Дополнительный звукопоглощающий эффект создают однотонные занавески из плотной ткани, повешенные в складку на расстоянии 15-20 см от ограждения. Ширина занавески должна быть в 2 раза больше ширины окна.

# **6.3. Противопожарная безопасность**

Причинами пожара могут быть:

* возгорание элементов аппаратуры;
* несоблюдение режимов эксплуатации оборудования;
* неправильные действия персонала.

Помещения, в которых располагаются ЭВМ, по пожарной опасности относятся к категории пожароопасности «В», также нужно учитывать высокую стоимость электронного оборудования. Поэтому для изготовления строительных конструкций используется кирпич, железобетон, стекло, металлы и другие негорючие материалы. Применение дерева ограничено, но в случае его использования оно пропитывается огнезащитным веществом (антипиреном).

Воздух в помещении должен очищаться от пыли, так как она может воспламениться при попадании в воздух искр. В соответствии с правилами пожарной безопасности на каждые 100 м2 должен находиться углекислотные огнетушители ОУ-5, с помощью которых можно тушить возгорание различных материалов и установок напряжением до 1000В.

При возникновении пожара, в здании должна быть предусмотрена эвакуация людей, при которой люди смогут в кратчайшие сроки покинуть горящее здание. Помещение должно быть оборудовано противопожарной сигнализацией.

В помещениях применяются дымовые пожарные сигнализации. В местах, практически недоступных для тушения пожаров первичными средствами или передвижной техникой пожарной охраны, применяют установки автоматического пожаротушения. Для хранения носителей информации используются несгораемые металлические шкафы, двери в хранилище также должны быть несгораемыми. Для противодействия пожарам применяются специальные мероприятия, направленные на предотвращение и эффективную борьбу с пожарами, такие как установка противопожарных сигнализаций, установка схем для эвакуации и установка огнетушителей.

Для устранения локальных возгораний на рабочих местах в легкодоступном месте находятся ручные огнетушители марки ОУ-25, а также применяется специализированная система автоматического пожаротушения на основе газообразного фреона.

Для помещения, в котором есть вычислительная техника, устанавливается категория пожарной безопасности “Д” по НПБ 105 05.

ЭВМ нельзя устанавливать вблизи источников тепла или термоизлучателей. Персоналу нельзя хранить воду рядом с техникой или проводами.

Чтобы предотвратить пожары в помещении, необходимо максимально сократить количество предметов из горючих материалов или разместить их на безопасном расстоянии.

В качестве первичного средства тушения пожара используется ОУ-2 являющийся углекислотным огнетушителем. Ручные огнетушители устанавливаются из расчета один огнетушитель на 40-50 м2, но не меньше двух в помещении. Также в здании должны быть инструкции по пожарной безопасности и планы эвакуации, которые будут показывать кратчайшие выходы из здания в случае пожара.

Противопожарная безопасность при работе с персональным компьютером

На рабочем месте запрещается иметь огнеопасные вещества.

В помещениях запрещается:

* зажигать огонь;
* включать электрооборудование, если в помещении пахнет газом;
* курить;
* сушить что-либо на отопительных приборах;
* закрывать вентиляционные отверстия в электроаппаратуре

Источниками воспламенения являются:

* искра при разряде статического электричества;
* искры от электрооборудования;
* искры от удара и трения;
* открытое пламя.

При возникновении пожароопасной ситуации или пожара персонал должен немедленно принять необходимые меры для его ликвидации, одновременно оповестить о пожаре администрацию.

Помещения с электрооборудованием должны быть оснащены огнетушителями типа ОУ-2 или ОУБ-3.

**Вывод**

В данной части выпускной квалификационной работы были рассмотрены мероприятия по технике безопасности и выделены требования охраны труда.

Так же были рассмотрены мероприятия противопожарной защиты, были определены категория и степень пожарной безопасности объекта и приняты решения для уменьшения вероятности пожара и его предотвращение.

1. **ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ ПРОЕКТА**

Экономическая целесообразность разработки и внедрения программного обеспечения определяется экономическим эффектом, который будет получен производителями при их реализации и потребителями при их использовании. По величине ожидаемого экономического эффекта принимается решение о целесообразности инвестиций в разработку того или иного программного продукта.

По характеру объекта вложений инвестиции в разработку программного обеспечения относят к интеллектуальным инвестициям.

При создании программного продукта важно оценить его себестоимости (затраты на разработку).

Определение стоимости разработки программного обеспечения (ПО) – себестоимости ПО

* 1. **Расчет затрат на разработку мобильного приложения**
     1. **Затраты на оплату труда (Зот)**

Затраты на оплату труда разработчика ПО складывается из двух составляющих: основной заработной платы и дополнительной заработной платы

Расчет текущих тарифных ставок оплаты труда персонала

Таблица 7.1.

|  |  |  |  |
| --- | --- | --- | --- |
| Категории работников- исполнителей работ | Тарифная ставка руб./чел.-ч | Количество рабочих часов | Всего руб. |
| Разработчик мобильного приложения | 427,43 | 192 | 82 067 |
| US – дизайнер (инженер) | 394,75 | 16 | 6 316 |
| Аналитик программного обеспечения | 361,22 | 12 | 4 335 |
| Специалист -тестировщик | 328,54 | 36 | 11 827 |
| Администратор баз данных | 295,85 | 4 | 1 183 |
| IT-администратор | 213,72 | 16 | 3 420 |
| ИТОГО | х | ∑t = 276 | Зосн= 86 158 |

∑t =tрмп + tдиз + tапо + tст + tабд +tit-а, (1)

t = 192+16+12+36+4+16=276 ч.

Основная заработная плата рассчитывается в руб. по формуле (2)

Зосн = ∑ t ⋅ (ТСМЕС / КЧР) ×ТК (2)

Зосн = 276 ⋅ (20 500 / 176) ⋅ 2,68 = 276 ⋅ 116,48 ⋅ 2,68 = 86 158 руб.

где

∑ t – суммарные затраты труда, вычисляемые по формуле (1), час.;

ТСМЕС – месячная тарифная ставка 1 разряда, руб.;

КЧР – среднемесячная расчетная норма рабочего времени

(среднее количество часов работы в месяц), час.;

ТК – тарифный коэффициент, соответствующий разряду работ разработчика ПО, (2,68)

Дополнительная заработная плата составляет 20% от основной заработной платы, рассчитывается по формуле (3):

Здоп = (20 /100) ⋅ Зосн, (3)

Здоп = 0,2 ⋅ 86 158 = 17 232руб.

Затраты на оплату труда разработчика ПО в рублях (ФОТ – фонд оплаты труда) представляет собой сумму основной и дополнительной заработной платы и рассчитывается по формуле (4):

ФОТ = Зосн + Здоп, (4)

ФОТ = 86 158 + 17 232= 103 390руб.

Отчисления от фонда оплаты труда включает страховые взносы:

Страховые взносы с зарплаты

Таблица 7.2

|  |  |
| --- | --- |
| Взносы | Ставка |
| Пенсионное страхование | 22% |
| Социальное страхование | 2,9% |
| Медицинское страхование | 5,1% |
| Всего | 30% |

Отчисления от фонда оплаты труда рассчитываются по формуле (5):

ОФОТ = (30 / 100) ⋅ ФОТ, (5)

ОФОТ = 0,3 ⋅ 103 390 = 31 017 руб.

Затраты на оплату труда разработчика ПО рассчитывается по формуле (6):

ЗОТ = ФОТ + ОФОТ, (6)

ЗОТ = 103 390 + 31 017 = 134 407 руб.

# **Эксплуатационные затраты на оборудование**

Суммарная годовая стоимость эксплуатационных затрат Cэз рассчитывается по формуле (7):

Cэз = СТО +СМК +СЭЭ + Агод, (7)

Сэз = 1 523 + 1 015 + 505 + 10 150 = 13 193руб.,

где

СТО - затраты на техническое обслуживание и ремонт оборудования, руб.,

СМК – затраты на материалы и комплектующие, руб.,

СЭЭ - годовая стоимость электроэнергии, руб.,

Агод - годовые амортизационные отчисления, руб.

Затраты на техническое обслуживание и ремонт (СТО) составляют 3 % от стоимости оборудования по формуле (8):

СТО = 0,03 ⋅ Собор, (8)

Сто = 0,03 ⋅ 50 750 ≈ 1 523 руб.

Затраты на материалы и комплектующие принимаются в размере 2% от стоимости оборудования по формуле (9):

СМК = 0,02 ⋅ Собор, (9)

СМК = 0,02 ⋅ 50 750 = 1 015руб

Амортизационные отчисления производятся по установленным нормам амортизации, выражаются, в процентах к стоимости оборудования и рассчитываются по формуле (10):

Агод = Собор ⋅ (НА / 100), (10

Агод = 50 750 ⋅ (20 / 100) = 10 150 руб

где

Cобор – стоимость компьютера;

НА – норма амортизации, которая рассчитывается по формуле (11):

НА = 100 / Тнорм,, (11)

Тнорм – нормативный срок службы (для персонального компьютера Тнорм = 5 лет)

Стоимость электроэнергии вычисляется по формуле (12):

СЭЭ = M ⋅ kз ⋅ Fэф ⋅ Сквт.ч ⋅ КС, (12)

СЭЭ = 0,045 \* 0,8 \* 1976 \* 6,76 \* 1,05 ≈ 505 руб

где

M – мощность компьютера, Квт (емкость аккумулятора);

kз – коэффициент загрузки, учитывающий использование оборудования по

времени (0,8);

Fэф – эффективный фонд рабочего времени;

Cквт.ч – стоимость 1 квт-час электроэнергии (от 6,76 - 7,03- 8,84 - 9,2 руб./квт по состоянию на март 2022 года);

КС - коэффициент, учитывающий потери в сети (Kc=1,05)

Fэф – эффективный фонд рабочего времени, рассчитывается по формуле (13):

Fэф = Дном ⋅ d ⋅ (1 – (f / 100)) = 247 \* 8 \* (1 - (0,02 / 100)) = 1 976 ч (13)

где

Дном = 247 – номинальное число рабочих дней в году при пятидневной рабочей неделе;

d = 8 – продолжительность рабочего дня, час;

f = 2% – планируемый процент времени на ремонт оборудования

Рассчитываем суммарную годовую стоимость эксплуатационных затрат Cэз используя формулу 7:

Cэз = СТО +СМК +СЭЭ + Агод

Cэз - суммарная годовая стоимость эксплуатационных затрат, вычисленный по формуле (7).

Полученная стоимость эксплуатационных затрат – это значения годовых расходов. Необходимо их скорректировать в соответствии с временным коэффициентом (так как оборудование будет эксплуатироваться не весь год, а только в течение времени Σt смотреть табл 1) который определяется исходя из суммарных годовых эксплуатационных затрат, которые рассчитываются по формуле (14):

Эз = ∑t х (Сэз / Fэф), (14)

Эз = 276 \* (13 193/1 976) ≈ 1 843руб.

где

Эз - стоимость эксплуатационных затрат;

Fэф – эффективный фонд рабочего времени, вычисленный по формуле (13);

Σt – общее время использования оборудования для решения задачи (см. таблицу 7.1.).

# **Затраты на материалы**

Возможен расчет затрат на материалы (ЗМ ) прямым счетом (по факту).

Расчет затрат на материалы

Таблица 7.3

|  |  |  |  |
| --- | --- | --- | --- |
| Материалы | Количество, шт. | Цена за единицу, руб. | Всего, руб. |
| Оплата услуг провайдера | 2 | 600 | 1 200 |
| Бумага | 95 | 1,2 | 114,00 |
| Картридж | 1 | 550 | 550,00 |
| Итого (Зм) | | | 1 864,00 |

**Накладные расходы**, связанные с управлением, организационными расходами и прочими дополнительными затратами, составляют 40% от фонда оплаты труда, вычисляются по формуле (15):

Cнакл = 0,4× ФОТ, (15)

Cнакл = 0,4 \* 103 390 = 41 356 руб.

# **Себестоимость разработки программного обеспечения**

Стоимость программного обеспечения

Таблица 7.4

|  |  |  |  |
| --- | --- | --- | --- |
| **№**  **п/п** | Наименование статей затрат | Обозначение | Сумма, руб |
| 1 | Затраты на материалы | ЗМ | 1 864 |
| 2 | Эксплуатационные затраты (ф.12) | Эз | 1 843 |
| 3 | Фонд оплаты труда (ф.2) | ФОТ | 103 390 |
| 4 | Отчисления от фонда оплаты труда (ф.3) | ОФОТ | 31 017 |
| 5 | Накладные расходы (ф.14) | Cнакл | 41 356 |
| 6 | Себестоимость ПО без учета расходов на сопровождение и адаптацию | п.1+п.2+п.3+п.4+п.5 | 180 070 |
| 7 | Расходы на сопровождение и адаптацию | 10% от п. 6 | 18 007 |
| 8 | Полная (плановая) себестоимость ПО | Спол.= п.6 + п.7 | 198 077 |

# **Расчет экономического эффекта разработчика и пользователя (заказчика) программного обеспечения**

# **Экономический эффект у разработчика программного обеспечения**

Заказчик оплачивает разработчику всю сумму расходов по проекту (полная себестоимость ПО из таблицы 7.4 с учетом прибыли разработчика и налога на добавленную стоимость с учетом качества, потребительских свойств продукции (ПО) и конъюнктуры рынка.

Таким образом, в дипломном проекте отпускная цена программного обеспечения, представляет собой не цену за единицу продукции, а цену проекта вместе с его исходными кодами и документацией, за которую его можно продать и получить определенную выгоду. Прогнозируемая отпускная цена ПО (ЦПО) с учетом НДС рассчитывается по формуле (16):

ЦПО = , (16)

ЦПО = (198 077+ 39 615) \* (100 + 20) / 100 = 285 214руб.,

где

ЦПО -отпускная цена программного обеспечения;

Сполн – полная (плановая) себестоимость ПО, руб., (из таблицы 7.4);

П – прибыль разработчика ПО, руб.;

СТНДС – ставка налога на добавленную стоимость (=20%), в %.

Прибыль закладывается в цену исходя из уровня рентабельности (устанавливается студентом самостоятельно), расчет производится по формуле (17):

П = , (17)

П = 20\*198 077/100 = 39 615руб.,

где

П – прибыль, руб.;

R – уровень рентабельности, % (в рамках дипломного проекта рекомендуемый уровень рентабельности ≈ 20%).

Сполн – полная (плановая) себестоимость ПО, руб., (из таблицы 7.4).

Ввиду того, что программное обеспечение разрабатывается для одного объекта, в качестве экономического эффекта разработчика от реализованного программного обеспечения можно рассматривать чистую прибыль (ЧП), которая рассчитывается по формуле (18):

ЧП = , (18)

ЧП = 39 615\* (100 – 20) / 100 = 31 692руб.,

где

ЧП - чистая прибыль, руб.;

П – прибыль, руб.;

СТП – ставка налогообложения прибыли составляет 20%.

# **Экономический вывод**

**Вывод:** таким образом, разработчик программного обеспечения может продать заказчику программное обеспечение по рассчитанной цене 285 214 руб., что покроет затраты на разработку ПО - 198 077 руб. и обеспечит получение чистой прибыли при его реализации заказчику 31 692 руб.

**ЗАКЛЮЧЕНИЕ**

В процессе разработки дипломного проекта и разработки мобильного приложения в соответствии с поставленным заданием удалось достичь поставленных задач:

* проведен анализ предметной области;
* проведен обзор актуальности программного продукта;
* сформированы требования к программному продукту;
* сформированы бизнес-требования программного продукта;
* сформированы пользовательские требования программного продукта;
* сформированы функциональные требования программного продукта:
* сформированы нефункциональные требования программного продукта;
* сформированы требования к интерфейсам программного продукта;
* сформированы требования к данным;
* проведен обзор архитектуры системы;
* смоделирован основной сценарий программного продукта:
* произведено проектирование графического интерфейса;
* произведено проектирование и разработка модели данных;
* разработан программный продукт;
* произведено тестирование программного продукта;
* определены объекты тестирования;
* описаны объекты тестирования;
* разработана технологическая документация программного продукта;
* описаны мероприятия по технике безопасности и охране труда;
* выполнено технико-экономическое обоснование проекта.

В результате выполненной работы разработано мобильное приложение на операционной системе Android, дающее клиенту возможность пользоваться информацией в удобное для него время. Клиенту не нужен стационарный компьютер, достаточно будет иметь смартфон или планшет, чтобы получить нужную информацию о компании или товаре. Помимо этого, клиент мобильного приложения сможет записаться на различные скейтбординг-мероприятия, проходящие в городе.

Таким образом, выполнена цель данного дипломного проекта – разработка мобильного приложения для организации социального взаимодействия участников мероприятий по скейтбордингу и выполнены все задачи проекта.

# **СПИСОК ИСТОЧНИКОВ ИНФОРМАЦИИ**

1. Вигерс Карл И., Битти Джой. «Разработка требований к программному обеспечению». Печатное издание. Год издания 2017;
2. Герберт Шилдт «Java 8. Полное руководство». Печатное издание Год издания 2017.
3. Гриффитс Дон, Гриффитс Дэвид. «Head First. Программирование для Android». 3-е печатное издание. Год издания 2018;
4. Дарвин Ян. Ф. Android. «Сборник рецептов. Задачи и решения для разработчиков приложений». Печатное издание. Год издания 2018;
5. Джошуа Блох. «Java. Эффективное программирование». Печатное издание. Год издания 2018;
6. Кузнецов С.Д. Основы баз данных. - 2-е изд. - М.: "Интернет-университет информационных технологий - ИНТУИТ. ру", 2018;
7. Кори Сандлер, Баджетт Том «Искусство тестирования программ». Год выпуска 2020. 3-е печатное издание
8. Норенков И. П. Основы автоматизированного проектирования: Учеб. для вузов. 2-е изд., перераб. и доп. — М.: Изд-во МГТУ им. Н. Э. Баумана, 2018;
9. Постановление правительства РФ от 1 ноября 2012 г. No 1119 Об утверждении требований к защите персональных данных при их обработке в информационных системах персональных данных;
10. Постановление правительства РФ от 21 марта 2012 г. No 211 Об утверждении перечня мер, направленных на обеспечение выполнения обязанностей, предусмотренных федеральным законом "О персональных данных" и принятыми в соответствии с ним нормативными правовыми актами, операторами, являющимися государственными или муниципальными органами;
11. Филипс Билл, Стюарт К. «Android.Программирование для профессионалов» 3-е печатное издание. Год издания 2017;
12. Федорова Г.Н. «Разработка, внедрение и адаптация программного обеспечения отраслевой направленности»: Учебное пособие / Г.Н. Федорова. – Москва: КУРС: ИНФРА – М, 2021;
13. Федеральный закон Российской Федерации от 27 июля 2006 г. No 152-ФЗ “О персональных данных”;
14. Федеральный закон Российской Федерации No 124871-4 “О служебной тайне”;
15. Приложения к методическим рекомендациям для организации защиты информации при обработке персональных данных в учреждениях здравоохранения, социальной сферы, труда и занятости;
16. Ян Клифтон. «Проектирование пользовательских интерфейсов». Печатное издание. Год издания 2017;

# **ПРИЛОЖЕНИЕ**

User.java

package com.example.skateco.models;  
  
public class User {  
 private String login, email, password;  
  
 public User() {  
 }  
  
 public User(String login, String email, String password) {  
 this.login = login;  
 this.email = email;  
 this.password = password;  
 }  
  
 public String getLogin() {  
 return login;  
 }  
  
 public void setLogin(String login) {  
 this.login = login;  
 }  
  
 public String getEmail() {  
 return email;  
 }  
  
 public void setEmail(String email) {  
 this.email = email;  
 }  
  
 public String getPassword() {  
 return password;  
 }  
  
 public void setPassword(String password) {  
 this.password = password;  
 }  
}

Авторизация и регистрация. MainActivity

package com.example.skateco;  
  
  
import androidx.annotation.RequiresApi;  
import androidx.appcompat.app.AlertDialog;  
import androidx.appcompat.app.AppCompatActivity;  
  
  
import android.content.Intent;  
import android.os.Build;  
import android.os.Bundle;  
import android.text.TextUtils;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.widget.Button;  
import android.widget.RelativeLayout;  
  
import com.example.skateco.models.User;  
  
import com.google.android.material.snackbar.Snackbar;  
  
import com.google.firebase.auth.FirebaseAuth;  
import com.google.firebase.database.DatabaseReference;  
import com.google.firebase.database.FirebaseDatabase;  
import com.rengwuxian.materialedittext.MaterialEditText;  
  
  
public class MainActivity extends AppCompatActivity {  
  
 Button btnSingIn, btnRegister;  
 RelativeLayout root;  
  
 FirebaseAuth auth;  
 FirebaseDatabase db;  
 DatabaseReference users;  
  
 @RequiresApi(api = Build.VERSION\_CODES.*KITKAT*)  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 btnSingIn = findViewById(R.id.*btnSignIn*);  
 btnRegister = findViewById(R.id.*btnRegister*);  
 root = findViewById(R.id.*root\_element*);  
 btnRegister.setOnClickListener(v -> showRegisterWindow());  
 btnSingIn.setOnClickListener(v -> showSingInWindow());  
  
 auth = FirebaseAuth.*getInstance*();  
 db = FirebaseDatabase.*getInstance*();  
 users = db.getReference("Users");  
 }  
  
 private void showSingInWindow(){  
 AlertDialog.Builder dialog = new AlertDialog.Builder(this);  
 dialog.setTitle("Авторизироваться");  
 dialog.setMessage("Введите все данные для авторизации");  
  
 LayoutInflater inflater = LayoutInflater.*from*(this);  
 View sign\_in\_window = inflater.inflate(R.layout.*sign\_in\_window*, null);  
 dialog.setView(sign\_in\_window);  
  
 final MaterialEditText email = sign\_in\_window.findViewById(R.id.*emailField*);  
 final MaterialEditText password = sign\_in\_window.findViewById(R.id.*passwordField*);  
  
 dialog.setNegativeButton("Отмена", (dialogInterface, which) -> dialogInterface.dismiss());  
 dialog.setPositiveButton("Войти", (dialogInterface, which) -> {  
 if (TextUtils.*isEmpty*(email.getText().toString())){  
 Snackbar.*make*(root, "Введите ваш email", Snackbar.*LENGTH\_SHORT*).show();  
 return;  
 }  
 if (TextUtils.*isEmpty*(password.getText().toString())){  
 Snackbar.make(root, "Введите пароль", Snackbar.LENGTH\_SHORT).show();  
 return;  
 }  
 //Авторизация пользователя  
 auth.signInWithEmailAndPassword(email.getText().toString(), password.getText().toString()).addOnSuccessListener(authResult -> {  
 startActivity(new Intent(MainActivity.this, NewsActivity.class));  
 finish();  
 }).addOnFailureListener(e -> Snackbar.make(root, "Вы ввели неверные данные"+ "\n" + e.getMessage(),Snackbar.LENGTH\_SHORT).show());  
  
 });  
 dialog.show();  
 }  
 private void showRegisterWindow() {  
 AlertDialog.Builder dialog = new AlertDialog.Builder(this);  
 dialog.setTitle("Зарегистрироваться");  
 dialog.setMessage("Введите все данные для регистрации");  
  
 LayoutInflater inflater = LayoutInflater.from(this);  
 View register\_window = inflater.inflate(R.layout.register\_window, null);  
 dialog.setView(register\_window);  
  
 final MaterialEditText email = register\_window.findViewById(R.id.emailField);  
 final MaterialEditText login = register\_window.findViewById(R.id.loginField);  
 final MaterialEditText password = register\_window.findViewById(R.id.passwordField);  
  
 dialog.setNegativeButton("Отмена", (dialogInterface, which) -> dialogInterface.dismiss());  
 dialog.setPositiveButton("Добавить", (dialogInterface, which) -> {  
 if (TextUtils.isEmpty(email.getText().toString())){  
 Snackbar.make(root, "Введите вашу почту", Snackbar.LENGTH\_SHORT).show();  
 return;  
 }  
 if (TextUtils.isEmpty(login.getText().toString())){  
 Snackbar.make(root, "Введите ваш логин", Snackbar.LENGTH\_SHORT).show();  
 return;  
 }  
 if (TextUtils.isEmpty(password.getText().toString())){  
 Snackbar.make(root, "Введите пароль", Snackbar.LENGTH\_SHORT).show();  
 return;  
 }  
  
 if (password.getText().toString().length() < 8){  
 Snackbar.make(root, "Введите пароль более 8 символов", Snackbar.LENGTH\_SHORT).show();  
 }  
 //Регистрация пользователя  
 auth.createUserWithEmailAndPassword(email.getText().toString(), password.getText().toString()).addOnSuccessListener(authResult -> {  
 User user = new User();  
 user.setLogin(login.getText().toString());  
 user.setEmail(email.getText().toString());  
 user.setPassword(password.getText().toString());  
  
 users.child(FirebaseAuth.getInstance().getCurrentUser().getUid()).setValue(user).addOnSuccessListener(unused -> Snackbar.make(root, "Вы успешно зарегистрировались", Snackbar.LENGTH\_SHORT).show());  
 }).addOnFailureListener(e -> Snackbar.make(root, "Ошибка регистрации" + e.getMessage(),Snackbar.LENGTH\_SHORT).show());  
 });  
 dialog.show();  
 }  
}

News.Activity

package com.example.skateco;  
  
import androidx.appcompat.app.AlertDialog;  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.annotation.SuppressLint;  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.widget.Button;  
import android.widget.ImageButton;  
  
import com.example.skateco.models.User;  
import com.google.firebase.auth.FirebaseAuth;  
import com.google.firebase.database.DatabaseReference;  
import com.google.firebase.database.FirebaseDatabase;  
  
public class NewsActivity extends AppCompatActivity {  
 ImageButton ImgNewsButt;  
 ImageButton ImgTackButt;  
 ImageButton ImgUserButt;  
 ImageButton ImgForumBut;  
 ImageButton ImgMapButt;  
 FirebaseAuth auth;  
 FirebaseDatabase db;  
 DatabaseReference users;  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_news*);  
 ImgNewsButt = findViewById(R.id.*skateGOday*);  
 ImgTackButt = findViewById(R.id.*tasksBut*);  
  
 ImgForumBut = findViewById(R.id.*forumBut*);  
  
 ImgUserButt = findViewById(R.id.*userAccBut*);  
 ImgMapButt = findViewById(R.id.*mapButton*);  
  
  
 auth = FirebaseAuth.*getInstance*();  
 db = FirebaseDatabase.*getInstance*();  
 users = db.getReference("Users");  
  
  
  
 ImgNewsButt.setOnClickListener( v -> showNewsOneWindow());  
 ImgForumBut.setOnClickListener(v -> showForumWindow());  
 ImgUserButt.setOnClickListener(v -> showUserLKWindow());  
 ImgTackButt.setOnClickListener(v -> showTaskWindow());  
 ImgMapButt.setOnClickListener(v -> showMapWindow());  
  
 }  
  
 private void showForumWindow() {  
 startActivity(new Intent(NewsActivity.this, ScrActivityForum.class));  
 }  
  
 private void showTaskWindow() {  
 startActivity(new Intent(NewsActivity.this, tackActivity.class));  
 }  
  
 private void showUserLKWindow() {  
 startActivity(new Intent(NewsActivity.this, user\_activity\_lk.class));  
 }  
  
 private void showMapWindow() {  
 startActivity(new Intent(NewsActivity.this, MapActivity.class));  
 }  
  
 private void showNewsOneWindow() {  
 AlertDialog.Builder dialog = new AlertDialog.Builder(this);  
 LayoutInflater inflater = LayoutInflater.*from*(this);  
 View news\_image\_window = inflater.inflate(R.layout.*news\_image\_window*, null);  
 dialog.setView(news\_image\_window);  
 dialog.setNegativeButton("Отмена", (dialogInterface, which) -> dialogInterface.dismiss());  
 dialog.show();  
 }  
}

Map.activity

package com.example.skateco;  
  
import android.os.Bundle;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import com.yandex.mapkit.Animation;  
import com.yandex.mapkit.MapKitFactory;  
import com.yandex.mapkit.geometry.Point;  
import com.yandex.mapkit.map.CameraPosition;  
  
import com.yandex.mapkit.mapview.MapView;  
  
public class MapActivity extends AppCompatActivity {  
  
 private final String MAPKIT\_API\_KEY = "b0dac91b-e31d-4da1-9988-e2e9bddc0548";  
 private static final Point *TARGET\_LOCATION* = new Point(59.945933, 30.320045);  
 private MapView mapView;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
  
 MapKitFactory.*setApiKey*(MAPKIT\_API\_KEY);  
  
  
 MapKitFactory.*initialize*(this);  
 setContentView(R.layout.*activity\_map*);  
 super.onCreate(savedInstanceState);  
  
 mapView = (MapView)findViewById(R.id.*mapview*);  
 mapView.getMap().move(  
 new CameraPosition(*TARGET\_LOCATION*, 14.0f, 0.0f, 0.0f),  
 new Animation(Animation.Type.*SMOOTH*, 5),  
 null);  
  
 }  
 @Override  
 protected void onStop() {  
 // Вызов onStop нужно передавать инстансам MapView и MapKit.  
 mapView.onStop();  
 MapKitFactory.*getInstance*().onStop();  
 super.onStop();  
 }  
  
 @Override  
 protected void onStart() {  
 // Вызов onStart нужно передавать инстансам MapView и MapKit.  
 super.onStart();  
 mapView.onStart();  
 MapKitFactory.*getInstance*().onStart();  
 }  
}

record.Activity

package com.example.skateco;  
  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.fragment.app.Fragment;  
import androidx.fragment.app.FragmentTransaction;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.widget.Button;  
import android.widget.FrameLayout;  
import android.widget.ImageButton;  
import android.widget.RelativeLayout;  
  
import com.google.android.material.snackbar.Snackbar;  
  
public class recordActivity extends AppCompatActivity {  
 RelativeLayout root;  
 ImageButton ImgNewsButt;  
 ImageButton ImgTackButt;  
 ImageButton ImgForumBut;  
 ImageButton ImgUserButt;  
 ImageButton ImgMapButt;  
  
 Button DismissFrameButt;  
 Button UpdateFrameButt;  
 private final RecOneFragment FirstFragment = new RecOneFragment();  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_record*);  
 root = findViewById(R.id.*recordRoot*);  
 /\*Кнопки меню\*/  
 ImgNewsButt = findViewById(R.id.*newsButt*);  
 ImgTackButt = findViewById(R.id.*tasksBut*);  
 ImgForumBut = findViewById(R.id.*forumBut*);  
 ImgUserButt = findViewById(R.id.*userAccBut*);  
 ImgMapButt = findViewById(R.id.*mapButton*);  
 /\*Кнопки фрагмента\*/  
 DismissFrameButt = findViewById(R.id.*DismissFrameButt*);  
 UpdateFrameButt = findViewById(R.id.*UpdateFrameButt*);  
 /\*Листенги кнопкок\*/  
 ImgNewsButt.setOnClickListener( v -> showNewsOneWindow());  
 ImgForumBut.setOnClickListener(v -> showForumWindow());  
 ImgUserButt.setOnClickListener(v -> showUserLKWindow());  
 ImgTackButt.setOnClickListener(v -> showTaskWindow());  
 ImgMapButt.setOnClickListener(v -> showMapWindow());  
  
 /\*Инициализация фреймов, подгрузка сразу первого\*/  
  
 setNewFragment(FirstFragment);  
 /\*Листенги кнопок фрейма\*/  
 DismissFrameButt.setOnClickListener(v -> FragOneShow());  
 UpdateFrameButt.setOnClickListener(v -> FragTwoShow());  
  
 }  
  
 private void showForumWindow() {  
 startActivity(new Intent(recordActivity.this, ScrActivityForum.class));  
 }  
  
  
 private void FragOneShow() {  
 Snackbar.*make*(root, "Вы отписались от мероприятия", Snackbar.*LENGTH\_SHORT*).show();  
 setNewFragment(FirstFragment);  
 }  
 private void FragTwoShow() {  
 RecTwoFragment SecondFragment = new RecTwoFragment();  
 setNewFragment(SecondFragment);  
 }  
  
 private void setNewFragment(Fragment fragment) {  
 FragmentTransaction ft = getSupportFragmentManager().beginTransaction();  
 ft.replace(R.id.*frameLayout*, fragment);  
 ft.commit();  
 }  
  
 private void showMapWindow() {  
 startActivity(new Intent(recordActivity.this, MapActivity.class));  
 }  
  
 private void showTaskWindow() {  
 startActivity(new Intent(recordActivity.this, tackActivity.class));  
 }  
  
 private void showUserLKWindow() {  
 startActivity(new Intent(recordActivity.this, user\_activity\_lk.class));  
 }  
  
 private void showNewsOneWindow() {  
 startActivity(new Intent(recordActivity.this, NewsActivity.class));  
 }  
}

task.activity

package com.example.skateco;  
  
import androidx.appcompat.app.AlertDialog;  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.widget.Button;  
import android.widget.ImageButton;  
import android.widget.RelativeLayout;  
  
import com.google.android.material.snackbar.Snackbar;  
  
public class tackActivity extends AppCompatActivity {  
  
 ImageButton ImgNewsButt;  
 ImageButton ImgMapButt;  
 ImageButton ImgUserButt;  
 ImageButton ImgForumBut;  
 Button RecInButt;  
  
 ImageButton RecNews;  
  
 RelativeLayout root;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_tack*);  
  
 ImgMapButt = findViewById(R.id.*mapButton*);  
 ImgUserButt = findViewById(R.id.*userAccBut*);  
 ImgNewsButt = findViewById(R.id.*newsButt*);  
 ImgForumBut = findViewById(R.id.*forumBut*);  
 RecInButt = findViewById(R.id.*ButtRecIn*);  
 RecNews = findViewById(R.id.*RecNews*);  
 root = findViewById(R.id.*RecRoot*);  
  
  
 ImgNewsButt.setOnClickListener(v -> showNewsActivity());  
 ImgMapButt.setOnClickListener(v -> showMapWindow());  
 ImgUserButt.setOnClickListener(v -> showUserLKWindow());  
 ImgForumBut.setOnClickListener(v -> showForumWindow());  
 RecInButt.setOnClickListener(v -> RecordSuccess());  
 RecNews.setOnClickListener(v -> RecordDialogShow());  
 }  
  
 private void showForumWindow() {  
 startActivity(new Intent(tackActivity.this, ScrActivityForum.class));  
 }  
  
 private void RecordDialogShow() {  
 AlertDialog.Builder dialog = new AlertDialog.Builder(this);  
 LayoutInflater inflater = LayoutInflater.*from*(this);  
 View task\_news\_window = inflater.inflate(R.layout.*task\_news\_window*, null);  
 dialog.setView(task\_news\_window);  
 dialog.setNegativeButton("Закрыть", (dialogInterface, which) -> dialogInterface.dismiss());  
 dialog.show();  
 }  
  
 private void RecordSuccess() {  
 /\*Запись человека на мероприятие\*/  
 Snackbar.*make*(root, "Вы записались на мероприятие", Snackbar.*LENGTH\_SHORT*).show();  
  
 }  
  
 private void showUserLKWindow() {  
 startActivity(new Intent(tackActivity.this, user\_activity\_lk.class));  
 }  
  
 private void showMapWindow() {  
 startActivity(new Intent(tackActivity.this, MapActivity.class));  
 }  
  
 private void showNewsActivity() {  
 startActivity(new Intent(tackActivity.this, NewsActivity.class));  
 }  
}

strings.xml

<resources>  
 <string name="app\_name">Skate.CO</string>  
 <string name="auto">Пожалуйста, авторизируйтесь</string>  
 <string name="singIn">Авторизация</string>  
 <string name="Register">Регистрация</string>  
 <string name="PassHint">Введите пароль</string>  
 <string name="PassConf">Подтвердите пароль</string>  
 <string name="EmailHint">Введите e-mail</string>  
 <string name="Login">Ваш логин</string>  
 <string name="title\_activity\_news">NewsActivity</string>  
 <string name="navigation\_drawer\_open">Open navigation drawer</string>  
 <string name="navigation\_drawer\_close">Close navigation drawer</string>  
 <string name="nav\_header\_title">Android Studio</string>  
 <string name="nav\_header\_subtitle">android.studio@android.com</string>  
 <string name="nav\_header\_desc">Navigation header</string>  
 <string name="action\_settings">Settings</string>  
 <string name="News">Новости</string>  
 <string name="menu\_home">Home</string>  
 <string name="logout">Выход</string>  
 <string name="menu\_gallery">Gallery</string>  
 <string name="menu\_slideshow">Slideshow</string>  
 <string name="title\_activity\_maps">Map</string>  
 <string name="textLK">Это ваш профиль</string>  
 <string name="record">Просмотреть запись</string>  
 <string name="redactor">Редактировать аккаунт</string>  
 <string name="technical">Тех. помощь</string>  
 <string name="zapisatsya">Записаться</string>  
 <string name="go\_skateboarding\_day">Go Skateboarding Day!</string>  
 <string name="aboutGoSkbDay">Go Skateboarding Day, или День скейтбординга, это всемирный праздник всех скейтбордистов. Данное мероприятие пройдет 21 июня 2022 года в г.Санкт-Петербург</string>  
 <string name="rec\_now">Проходящие мероприятия</string>  
 <!-- *TODO: Remove or change this placeholder text* -->  
 <string name="hello\_blank\_fragment">Hello blank fragment</string>  
 <string name="rec\_title">Ваши записи:</string>  
 <string name="skbinfo">Данное мероприятие состоится: 21 июня 2022, в г.Санкт-Петербург</string>  
 <string name="title\_activity\_scr\_forum">ScrollingActivity</string>  
 <string name="news\_info\_window">Каждый год 21 июня вдохновленные скейтбордисты выделяют место в своем календаре для Дня скейтбординга!\n Поставщики, спонсоры, парки и частные лица по всему миру проводят мероприятия, демонстрирующие эту бунтарскую и творческую культуру. Скейтбординг, представляющий собой сочетание атлетизма, танца и искусства, вырос в городском мире. Он также преодолевает многокультурные различия и обращается к разнообразному населению.</string>  
</resources>

Activity.main

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:background="@drawable/bg\_whitesmoke"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity"  
 android:id="@+id/root\_element"  
 >  
  
 <LinearLayout  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="150dp"  
 android:gravity="center"  
 android:layout\_centerHorizontal="true"  
 android:orientation="vertical">  
  
 <ImageView  
 android:id="@+id/logo"  
 android:layout\_width="273dp"  
 android:layout\_height="240dp"  
 android:layout\_marginBottom="20dp"  
 app:srcCompat="@drawable/logo\_pic" />  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:fontFamily="@font/roboto\_bold"  
 android:text="@string/auto"  
 android:textColor="@color/black"  
 android:textSize="18sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.5"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.113" />  
  
 <Button  
 android:id="@+id/btnSignIn"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:backgroundTint="@android:color/holo\_orange\_dark"  
 android:fontFamily="@font/roboto\_bold"  
 android:text="@string/singIn"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.473"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.483" />  
  
 <Button  
 android:id="@+id/btnRegister"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:backgroundTint="@android:color/holo\_orange\_dark"  
 android:fontFamily="@font/roboto\_bold"  
  
 android:text="@string/Register" />  
 </LinearLayout>  
  
</RelativeLayout>

Activity\_map.xml

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <com.yandex.mapkit.mapview.MapView  
 android:id="@+id/mapview"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"/>  
</RelativeLayout>

Activity\_news.xml

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".NewsActivity"  
 android:background="@drawable/bg\_whitesmoke">  
  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="@drawable/bg\_color"  
  
 android:orientation="horizontal">  
  
 <ImageView  
 android:id="@+id/logo"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:paddingTop="20dp"  
 android:paddingBottom="20dp"  
 app:srcCompat="@drawable/logo\_word" />  
 </LinearLayout>  
  
 <TextView  
 android:id="@+id/textNews"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="70dp"  
 android:fontFamily="@font/roboto\_bold"  
 android:paddingBottom="20dp"  
 android:text="@string/News"  
 android:textColor="@color/black"  
 android:textSize="48sp" />  
  
 <ImageButton  
 android:id="@+id/skateGOday"  
 android:layout\_width="293dp"  
 android:layout\_height="505dp"  
 android:layout\_centerInParent="true"  
 android:paddingTop="20dp"  
 app:srcCompat="@drawable/skate\_go\_day" />  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentBottom="true"  
 >  
  
 <ImageButton  
 android:id="@+id/newsButt"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_light"  
 app:srcCompat="@android:drawable/stat\_notify\_chat" />  
  
 <ImageButton  
 android:id="@+id/tasksBut"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@android:drawable/ic\_menu\_today" />  
  
 <ImageButton  
 android:id="@+id/forumBut"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@drawable/chat" />  
  
 <ImageButton  
 android:id="@+id/userAccBut"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="61dp"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@drawable/lk\_icon"  
 />  
  
 <ImageButton  
 android:id="@+id/mapButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@android:drawable/ic\_dialog\_map" />  
  
 </LinearLayout>  
  
  
</RelativeLayout>

Activity.record.xml

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".recordActivity"  
 android:background="@drawable/bg\_color"  
 android:id="@+id/recordRoot">  
  
  
 <androidx.constraintlayout.widget.ConstraintLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <TextView  
 android:id="@+id/textView2"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="32dp"  
 android:fontFamily="@font/roboto\_bold"  
 android:text="@string/rec\_title"  
 android:textAlignment="center"  
 android:textColor="@color/black"  
 android:textSize="24sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <FrameLayout  
 android:id="@+id/frameLayout"  
 android:layout\_width="386dp"  
 android:layout\_height="508dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.476"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView2" />  
  
 <LinearLayout  
 android:layout\_width="377dp"  
 android:layout\_height="63dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.529"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/frameLayout"  
 app:layout\_constraintVertical\_bias="0.185">  
  
 <Button  
 android:id="@+id/DismissFrameButt"  
 android:layout\_width="25sp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:fontFamily="@font/roboto\_bold"  
 android:backgroundTint="@android:color/holo\_orange\_dark"  
 android:layout\_marginHorizontal="20dp"  
 android:text="Отменить" />  
  
 <Button  
 android:id="@+id/UpdateFrameButt"  
 android:layout\_width="25sp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:fontFamily="@font/roboto\_bold"  
 android:backgroundTint="@android:color/holo\_orange\_dark"  
 android:layout\_marginHorizontal="20dp"  
 android:text="Обновить" />  
 </LinearLayout>  
  
 </androidx.constraintlayout.widget.ConstraintLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentBottom="true"  
 >  
  
 <ImageButton  
 android:id="@+id/newsButt"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@android:drawable/stat\_notify\_chat" />  
  
 <ImageButton  
 android:id="@+id/tasksBut"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@android:drawable/ic\_menu\_today" />  
  
 <ImageButton  
 android:id="@+id/forumBut"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@drawable/chat" />  
  
 <ImageButton  
 android:id="@+id/userAccBut"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="61dp"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@drawable/lk\_icon"  
 />  
  
 <ImageButton  
 android:id="@+id/mapButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@android:drawable/ic\_dialog\_map" />  
  
 </LinearLayout>  
  
  
</RelativeLayout>

Activity.record.xml

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".recordActivity"  
 android:background="@drawable/bg\_color"  
 android:id="@+id/recordRoot">  
  
  
 <androidx.constraintlayout.widget.ConstraintLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <TextView  
 android:id="@+id/textView2"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="32dp"  
 android:fontFamily="@font/roboto\_bold"  
 android:text="@string/rec\_title"  
 android:textAlignment="center"  
 android:textColor="@color/black"  
 android:textSize="24sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <FrameLayout  
 android:id="@+id/frameLayout"  
 android:layout\_width="386dp"  
 android:layout\_height="508dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.476"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView2" />  
  
 <LinearLayout  
 android:layout\_width="377dp"  
 android:layout\_height="63dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.529"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/frameLayout"  
 app:layout\_constraintVertical\_bias="0.185">  
  
 <Button  
 android:id="@+id/DismissFrameButt"  
 android:layout\_width="25sp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:fontFamily="@font/roboto\_bold"  
 android:backgroundTint="@android:color/holo\_orange\_dark"  
 android:layout\_marginHorizontal="20dp"  
 android:text="Отменить" />  
  
 <Button  
 android:id="@+id/UpdateFrameButt"  
 android:layout\_width="25sp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:fontFamily="@font/roboto\_bold"  
 android:backgroundTint="@android:color/holo\_orange\_dark"  
 android:layout\_marginHorizontal="20dp"  
 android:text="Обновить" />  
 </LinearLayout>  
  
 </androidx.constraintlayout.widget.ConstraintLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentBottom="true"  
 >  
  
 <ImageButton  
 android:id="@+id/newsButt"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@android:drawable/stat\_notify\_chat" />  
  
 <ImageButton  
 android:id="@+id/tasksBut"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@android:drawable/ic\_menu\_today" />  
  
 <ImageButton  
 android:id="@+id/forumBut"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@drawable/chat" />  
  
 <ImageButton  
 android:id="@+id/userAccBut"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="61dp"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@drawable/lk\_icon"  
 />  
  
 <ImageButton  
 android:id="@+id/mapButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@android:drawable/ic\_dialog\_map" />  
  
 </LinearLayout>  
  
  
</RelativeLayout>

Activity.task.xml

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".tackActivity"  
 android:id="@+id/RecRoot"  
 android:background="@drawable/bg\_color">  
  
 <androidx.constraintlayout.widget.ConstraintLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical">  
  
  
 <TextView  
 android:id="@+id/textView5"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/rec\_now"  
 android:textSize="24sp"  
 android:textColor="@color/black"  
 android:fontFamily="@font/roboto\_bold"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.498"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.054" />  
  
 <ImageButton  
 android:id="@+id/RecNews"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="20dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView5"  
 app:layout\_constraintVertical\_bias="0.053"  
 app:srcCompat="@drawable/goskate\_zapiss" />  
  
 <Button  
 android:id="@+id/ButtRecIn"  
 android:layout\_width="147dp"  
 android:layout\_height="45dp"  
 android:layout\_marginTop="4dp"  
 android:backgroundTint="@android:color/holo\_orange\_dark"  
 android:fontFamily="@font/roboto\_bold"  
 android:text="@string/zapisatsya"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/RecNews" />  
  
 </androidx.constraintlayout.widget.ConstraintLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentBottom="true"  
 >  
  
 <ImageButton  
 android:id="@+id/newsButt"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@android:drawable/stat\_notify\_chat" />  
  
 <ImageButton  
 android:id="@+id/tasksBut"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_light"  
 app:srcCompat="@android:drawable/ic\_menu\_today" />  
  
 <ImageButton  
 android:id="@+id/forumBut"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@drawable/chat" />  
  
 <ImageButton  
 android:id="@+id/userAccBut"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="61dp"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@drawable/lk\_icon"  
 />  
  
 <ImageButton  
 android:id="@+id/mapButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:background="@android:color/holo\_orange\_dark"  
 app:srcCompat="@android:drawable/ic\_dialog\_map" />  
  
 </LinearLayout>  
</RelativeLayout>

Register\_window.xml

<?xml version="1.0" encoding="utf-8"?>  
<androidx.cardview.widget.CardView  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
  
 app:cardElevation="10dp"  
 >  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="20dp"  
 android:orientation="vertical">  
 <!--Поле email-->  
 <com.rengwuxian.materialedittext.MaterialEditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/emailField"  
 android:hint="@string/EmailHint"  
 android:inputType="textEmailAddress"  
 android:text=""  
 android:textColor="#ffd88f"  
 android:textColorHint="#1e2129"  
 android:textSize="20sp"  
  
 app:met\_floatingLabel="highlight"  
 app:met\_baseColor="@color/design\_default\_color\_primary\_dark"  
 app:met\_primaryColor="#1e2129"  
 app:met\_singleLineEllipsis="true"  
 />  
 <!--Login-->  
 <com.rengwuxian.materialedittext.MaterialEditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/loginField"  
 android:hint="@string/Login"  
 android:inputType="text"  
 android:text=""  
 android:textColor="#ffd88f"  
 android:textColorHint="#1e2129"  
 android:textSize="20sp"  
  
 app:met\_floatingLabel="highlight"  
 app:met\_baseColor="@color/design\_default\_color\_primary\_dark"  
 app:met\_primaryColor="#1e2129"  
 app:met\_minCharacters="3"  
 app:met\_maxCharacters="16"  
 />  
 <!--Password-->  
 <com.rengwuxian.materialedittext.MaterialEditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/passwordField"  
 android:hint="@string/PassHint"  
 android:inputType="textPassword"  
 android:text=""  
 android:textColor="#ffd88f"  
 android:textColorHint="#1e2129"  
 android:textSize="20sp"  
  
 app:met\_floatingLabel="highlight"  
 app:met\_baseColor="@color/design\_default\_color\_primary\_dark"  
 app:met\_primaryColor="#1e2129"  
 app:met\_minCharacters="8"  
 app:met\_maxCharacters="32"  
  
 />  
  
 </LinearLayout>  
  
</androidx.cardview.widget.CardView>

Sign.in.window.xml

<?xml version="1.0" encoding="utf-8"?>  
<androidx.cardview.widget.CardView  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
  
 app:cardElevation="10dp"  
 >  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="20dp"  
 android:orientation="vertical">  
 <!--Поле email-->  
 <com.rengwuxian.materialedittext.MaterialEditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/emailField"  
 android:hint="@string/EmailHint"  
 android:inputType="textEmailAddress"  
 android:text=""  
 android:textColor="#ffd88f"  
 android:textColorHint="#1e2129"  
 android:textSize="20sp"  
  
 app:met\_floatingLabel="highlight"  
 app:met\_baseColor="@color/design\_default\_color\_primary\_dark"  
 app:met\_primaryColor="#1e2129"  
 app:met\_singleLineEllipsis="true"  
 />  
 <!--Password-->  
 <com.rengwuxian.materialedittext.MaterialEditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/passwordField"  
 android:hint="@string/PassHint"  
 android:inputType="textPassword"  
 android:text=""  
 android:textColor="#ffd88f"  
 android:textColorHint="#1e2129"  
 android:textSize="20sp"  
  
 app:met\_floatingLabel="highlight"  
 app:met\_baseColor="@color/design\_default\_color\_primary\_dark"  
 app:met\_primaryColor="#1e2129"  
 app:met\_minCharacters="8"  
 app:met\_maxCharacters="32"  
  
 />  
  
 </LinearLayout>  
  
</androidx.cardview.widget.CardView>

Task\_news\_window.xml

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="@drawable/bg\_color"  
 >  
  
 <ImageView  
 android:id="@+id/imageView3"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="64dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.454"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:srcCompat="@drawable/goskate\_zapiss" />  
  
 <TextView  
 android:id="@+id/textView4"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="36dp"  
 android:text="@string/go\_skateboarding\_day"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 android:fontFamily="@font/roboto\_bold"  
 android:textColor="@color/black"  
 android:textSize="32sp"  
 app:layout\_constraintTop\_toBottomOf="@+id/imageView3" />  
  
 <TextView  
 android:id="@+id/textView6"  
 android:layout\_width="388dp"  
 android:layout\_height="154dp"  
 android:layout\_marginTop="32dp"  
 android:fontFamily="@font/roboto\_bold"  
 android:textColor="@color/black"  
 android:textAlignment="center"  
 android:text="@string/aboutGoSkbDay"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.695"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView4"  
 app:layout\_constraintVertical\_bias="0.01" />  
</androidx.constraintlayout.widget.ConstraintLayout>

1. Бот - Робот, или бот, а также интернет-бот — специальная программа, выполняющая автоматически и/или по заданному расписанию какие-либо действия через интерфейсы, предназначенные для людей. Зачастую данные боты приводят к DDOS атакам, которые могут нанести серьёзный ущерб. [↑](#footnote-ref-1)
2. Лямбда-выражение – Лямбда представляет набор инструкций, которые можно выделить в отдельную переменную и затем многократно вызвать в различных местах программы. Основу лямбда-выражения составляет **лямбда-оператор**, который представляет стрелку **->**. Этот оператор разделяет лямбда-выражение на две части: левая часть содержит список параметров выражения, а правая, собственно, представляет тело лямбда-выражения, где выполняются все действия. [↑](#footnote-ref-2)