

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ «НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ЯДЕРНЫЙ
УНИВЕРСИТЕТ «МИФИ»
(НИЯУ МИФИ)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
по разработке мобильного приложения для планирования выполнения
прыжков с парашютом

Выполнил:
студент группы Б17-В71
Герилив Роман

Москва, 2020

СОДЕРЖАНИЕ

Список терминов и сокращений.....	4
Введение	5
1. Обзорная часть	0
1.1. Анализ функционала существующих мобильных приложений в сфере парашютного спорта.....	0
1.2. Обзор структуры для реализации приложения	0
1.3. Обзор технологий для создания мобильного приложения.....	0
1.4. Актуальность	0
2. Расчётно-конструкторская часть	0
2.1. Требования к функционалу приложения	0
2.2. Исследование стека используемых технологий	0
2.3. Исследование математической модели прыжка с парашютом	0
2.4. Разработка структуры базы данных	0
2.5. Разработка структуры серверной части приложения	0
2.6. Разработка структуры клиентской части приложения	0
2.7. Разработка интерфейса пользователя	0
2.7.1. Регистрация и авторизация	0
2.7.2. Главная страница	0
2.7.3. Погода	0
2.7.4. Математическая модель прыжка	0
2.7.5. Информация о пользователе	0
2.7.6. История парашютного спорта	0
2.7.7. Карта с аэродромами	0
2.7.8. Календарь прыжков	0
2.8. Разработка плана тестирования приложения	0
3. Экспериментальная часть	0
3.1. Структура проекта	0
3.2. Реализация серверной части приложения.....	0
3.3. Реализация базы данных в Oracle SQLDeveloper	0
3.4. Реализация клиентской части приложения (интерфейсы и алгоритмы).....	0
3.4.1. Регистрация и авторизация	0
3.4.2. Главная страница	0
3.4.3. Погода	0
3.4.4. Математическая модель прыжка	0

3.4.5. Информация о пользователе	0
3.4.6. История парашютного спорта	0
3.4.7. Карта с аэродромами	0
3.4.8. Календарь прыжков	0
3.5. Тестирование и отладка приложения	0
3.6. Руководство пользователя	0
Заключение.....	0
Список используемой литературы	0
Приложение	

СПИСОК ИСПОЛЬЗУЕМЫХ СОКРАЩЕНИЙ

HTTP - англ. HyperText Transfer Protocol - «протокол передачи гипертекста» - протокол прикладного уровня передачи данных;

XML - англ. eXtensible Markup Language - расширяемый язык разметки;

СУБД - Система управления базами данных, (англ. Database Management System, сокр. DBMS) — совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных;

Nickname – псевдоним пользователя;

Dropzone (дропзона) – термин, обозначающий площадку – аэродром, на котором выполняются прыжки с парашютом;

Активность – рабочая область приложения с различным интерфейсом.

ВВЕДЕНИЕ

Разработка мобильного приложения для статистического планирования выполнения прыжков с парашютом в максимально комфортных условиях. Также приложение является неким мобильным блокнотом (дневником) парашютиста с интерфейсной возможностью отметок количества прыжков и иных данных об этом. Реализован функционал для примерного определения максимальной скорости свободного падения при прыжках с различной высоты в различном положении парашютиста и расчётом критического времени раскрытия после отделения от самолёта. Приложение должно реализовывать следующий функционал: имиджевый – привлекать новый контингент в парашютный спорт; практический – определение комфортных годовых и месячных промежутков для прыжков, удобность статистики прыжков за период, а также математическую модель потенциального прыжка; информационный – ознакомление с историей парашютного спорта и иной нишевой информацией.

Все скриншоты рабочего приложения представлены с реального телефона. Начальная отладка и тестирование проходили на дополнительно настроенном в Android Studio эмуляторе.

1. Обзорная часть

В обзорной части будет рассмотрено ...

1.1. Анализ функционала существующих мобильных приложений в сфере парашютного спорта

Boogie – Skydiving logbook – иностранное нерусифицированное приложение для отображения статистики прыжков и иной общей информации.

1.2. Обзор структуры для реализации приложения

1.3. Обзор технологий для создания мобильного приложения

1.4. Актуальность

Данное приложение является весьма узконаправленным и может охватить лишь тематическую аудиторию, однако, потенциально может быть интересно и обычным пользователям.

Текущие показатели запросов за месяц, согласно сайту <https://wordstat.yandex.ru>:

- ✓ «Парашютный спорт» - 6 916 запросов
- ✓ «Парашют» - 276 545 запросов
- ✓ «Прыжок с парашютом» - 69 667 запросов

- 2. Расчётно-конструкторская часть
 - 2.1. Требования к функционалу приложения
 - 2.2. Исследование стека используемых технологий
 - 2.3. Исследование математической модели прыжка с парашютом
 - 2.4. Разработка структуры базы данных
 - 2.5. Разработка структуры серверной части приложения
 - 2.6. Разработка структуры клиентской части приложения
 - 2.7. Разработка интерфейса пользователя
 - 2.7.1. Регистрация и авторизация
 - 2.7.2. Главная страница
 - 2.7.3. Погода
 - 2.7.4. Математическая модель прыжка
 - 2.7.5. Информация о пользователе
 - 2.7.6. История парашютного спорта
 - 2.7.7. Карта с аэродромами
 - 2.7.8. Календарь прыжков
 - 2.8. Разработка плана тестирования приложения

3. Экспериментальная часть

3.1. Структура проекта

Разработана структура проекта, приведена на рисунке.

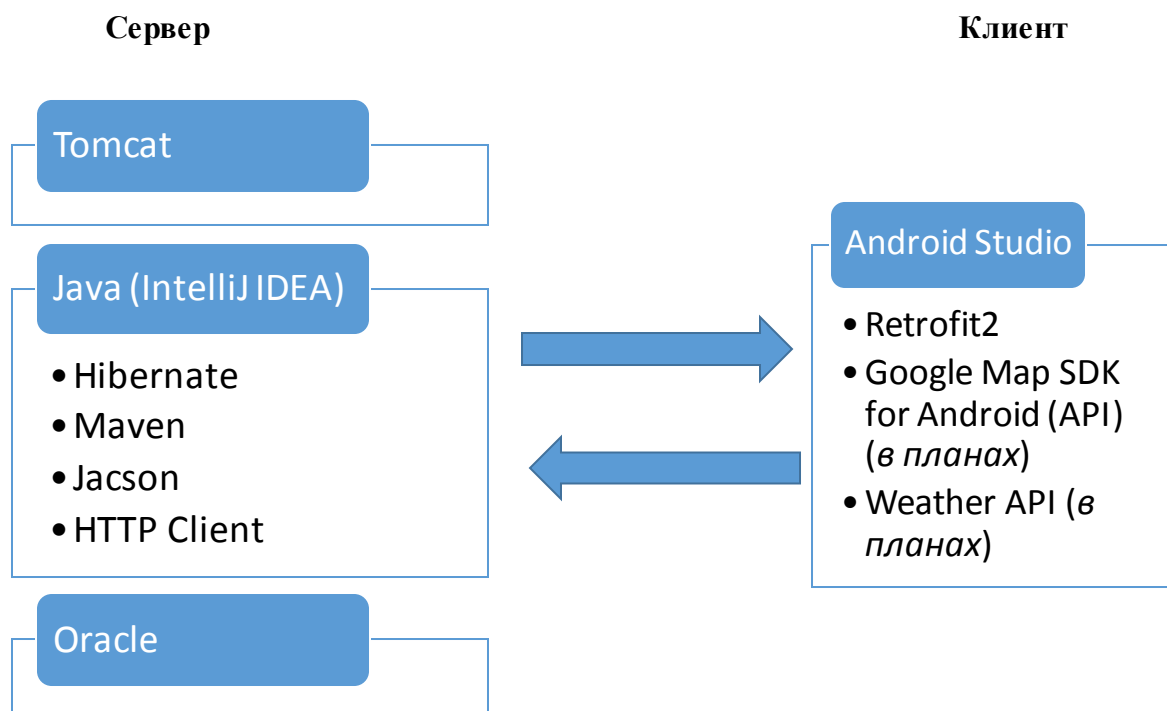


Рисунок № Структура проекта

3.2. Реализация серверной части приложения

- Tomcat - контейнер сервлетов для развёртывания приложения;
- Среда разработки Java (IntelliJ IDEA) - интегрированная среда разработки программного обеспечения для многих языков программирования, в частности Java;
- Hibernate – фреймворк для связи РСУБД и ООП;
- Maven – фреймворк для автоматизации сборки проектов на основе описания их структуры в файлах на языке POM (англ. Project Object Model), являющемся подмножеством XML;
- Jackson –сериализация/десериализация JSON;
- HTTP Client – фреймворк для работы с http запросами
- Oracle - реляционная СУБД. Хранит данные о регистрации клиента и информацию о нём.

3.3. Реализация базы данных в Oracle SQLDeveloper

База данных (далее БД) состоит на данный момент из 2-х таблиц (схема представлена на рисунке 1.4.1). В первой - содержится информация о регистрации пользователей, которая соответственно используется и в качестве подтверждения авторизации. Там имеются такие поля как id (идентифицирующий номер пользователя) – автоматически генерируется при регистрации нового пользователя, email (электронная почта), nickname (псевдоним пользователя) – при помощи триггера в БД после добавления в таблицу «Регистрация» автоматически добавляется в таблицу «Информация» и password (пароль). Стоит отметить, что в данный момент пароль никак не шифруется – это явный недостаток, который в дальнейшем будет исправляться, чтобы не было возможности получить пароль пользователя через БД. Во второй таблице содержится поле псевдонима из первой (в качестве основного ключа) и различная информация о самом пользователе. Вся информация из второй таблицы не является обязательной к заполнению, и любой пользователь сознательно или неосознанно может оставить все поля кроме nickname (заполняется автоматически при регистрации) пустыми или заполнить лишь некоторые из них.

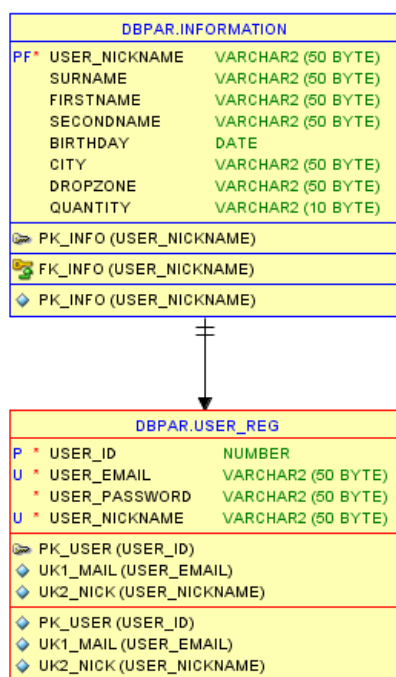


Рисунок № Схема БД

3.4. Реализация клиентской части приложения (интерфейсы и алгоритмы)

- Среда разработки Android Studio. Для создания активностей и интерфейса на них используются стандартные библиотеки и элементная база Android Studio.

Подключённая библиотека Retrofit2 используется для работы с HTTP запросами при регистрации, авторизации и сохранении данных о пользователе.

- Добавление карты по API (есть наработки, но не готовые для включения в проект) и погоды планируется в дальнейшем.

3.4.1.Регистрация и авторизация

Авторизация – это первая активность, которую видит пользователь при входе в приложение (рисунок 2.1.1).



Рис. № Интерфейс авторизации

Пользователь имеет в доступе 2 поля для ввода текста, кнопку «показать введенный пароль» справа от поля ввода пароля и 3 активных кнопки «Войти», «Регистрация» и «Отмена». В случае, если пользователь незарегистрирован при нажатии на кнопку «Регистрация» для пользователя откроется возможность зарегистрироваться (рассмотрено в пункте 2.1.2). Рассмотрим краткий алгоритм работы авторизации:

- 1) Ввод данных в активные поля
- 2) Формируется HTTP запрос со стороны клиента для отправки на сервер с целью подтверждения наличия данного пользователя в системе
- 3) Сервер проверяет сходство полученных данных с данными в БД
 - a. Если такого клиента в базе нет, выводится сообщение «Такой пользователь не зарегистрирован» (рис. 2.1.2)
 - b. Если email пользователя найден, но пароль не совпадает, то выводится ироничное сообщение «Неверный пароль, а функции поменять нет, ХА, вспоминай» - хочу сразу обратить внимание, что такой функции пока

действительно нет, но в дальнейшем это необходимый функционал. (рис. 2.1.3)

- с. Если логин (email) и пароль совпали с имеющимися в БД, то сервер возвращает положительный статус и пользователь, успешно авторизовавшись, входит на главную страницу приложения. Также из БД подцепляются данные из таблицы «Информация» и в соответствующей активности заполняются все поля.



Рис. № Нет пользователя



Рис. № Неверный пароль

При нажатии кнопки «Регистрация» становится неактивной кнопка «Войти», появляется дополнительное поле для ввода псевдонима (никнейма) пользователя появляется также новая кнопка «Зарегистрироваться», а кнопка «Регистрация» меняет своё название на «Назад к входу» (при её нажатии восстанавливается функционал авторизации (см. пункт 2.1.1). Функционалом кнопки «Зарегистрироваться» является сохранение данных о новом пользователе в БД. После нажатия этой кнопки, также возвращается принцип работы авторизации, но теперь не нужно вводить заново данные, т.к. они сохранены в активных полях, поэтому при нажатии кнопки вход пользователь перейдёт на главную страницу. Ещё раз опишу по пунктам алгоритм регистрации:

- 1) Нажатие на кнопку «Регистрация» («Вход неактивна», новое активное поле (никнейм), название «Регистрация» меняется на «Назад к входу», новая кнопка «Зарегистрироваться») (рис. 2.1.4)
- 2) Пользователь заполняет 3 активных поля (почта, никнейм и пароль);
- 3) Пользователь выбирает дальнейшее действие

- а. Кнопка «Зарегистрироваться»
 - i. Данные отправляются на сервер по HTTP;
 - ii. Сервер сохраняет нового пользователя в БД и заполняет поле никнейм в таблице «Информация» сохраняя все остальные поля пустыми (если не возникло ошибок на стороне сервера, все высвечиваются в нижней части экрана)
 - iii. От сервера приходит положительный статус регистрации
 - iv. Пользователь может выполнить авторизацию
- б. Кнопка «Отмена» или «Назад к входу»
 - i. Позволяет пользователю вернуться к авторизации



Рис. № Регистрация нового пользователя

3.4.2. Главная страница

На основной активности главной страницы (рис. №) пользователю даётся возможность выбрать одну из четырёх активностей: «WEATHER» (заглушка), «ПРЫГНУТЬ», «О СЕБЕ» и «ИСТОРИЯ». Каких-то особых алгоритмов в работе главной страницы нет, т.к. её основной задачей является переход по активностям.

Также на главной странице присутствует меню в правой верхней части экрана (Рис. №). Данное меню позволяет выйти к активности «Регистрации/авторизации», если это необходимо пользователю. И соответственно пользователь выходит из своего аккаунта и ему необходимо заново авторизоваться.



Рис. № Главная страница

3.4.3. Погода

На данный момент не реализовано находится в приложении в качестве макета будущей активности. В дальнейшем вижу данную активность как сводка прогнозов погоды по каждому аэродрому, который отмечен на активности с картой, запрос на обновление данных о погоде будет отправляться сервером N-раз в сутки и далее визуально обновляться у пользователей.

На данной активности присутствует меню с двумя вариантами, а именно «Выйти» - данный пункт перенаправляет на активность «Регистрации/авторизации» и пункт «На главную», который позволяет вернуться на «Главную страницу».



Рис. № Заглушка на активность «Погода»

3.4.4. Прыгнуть (активность математической модели прыжка)

В активности «Прыгнуть» пользователь наблюдает большое количество активных полей, в которые он может занести различные характеристики, как своих габаритов, так и желательных параметров прыжка. Всего на активности находится 6 активных полей (рисунок 2.4.1), в которых прописаны подсказки о назначении этого поля и физических величинах, в которых необходимо указывать значения. Кнопка для вычисления данных. 2 поля для вывода вычисленных данных.

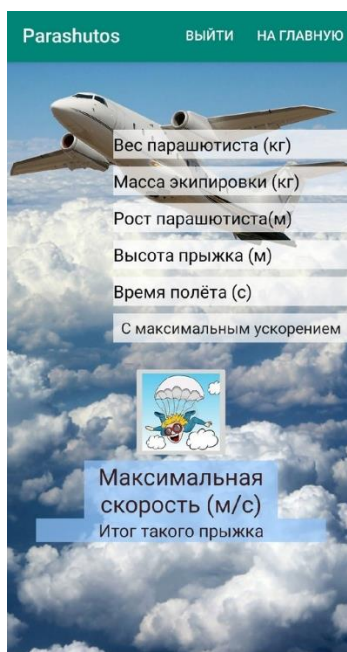


Рис. № Общий вид активности

Поля:

1. «Вес парашютиста» - вес самого спортсмена в килограммах;
2. «Масса экипировки» - подразумевается парашютная система, каска и иные принадлежности, которые будут на спортсмене;
3. «Рост парашютиста» - рост спортсмена в метрах;
4. «Высота прыжка» - соответственно высота, с которой парашютист будет отделяться от самолёта. Делится на 4 категории внутри приложения, от которых зависит условно обозначенная критическая высота раскрытия парашюта, от которой в дальнейшем по формулам высчитывается критическое время открытия (timeCritOpen) и высота, на которой будет достигнута максимальная скорость (heightMaxSpeed):
 - 1 категория – от 200 до 1000 метров: значение критической высоты открытия 200 м, набора максимальной скорости на этих высотах не будет;
 - 2 категория - от 1000 до 2000 метров: значение критической высоты открытия 300 м, примерное время набора максимальной скорости 12 с;

- 3 категория – от 2000 до 4000 метров: значение критической высоты открытия 350 м, примерное время набора максимальной скорости 12,5 с;
 - 4 категория – выше 4000 метров: значение критической высоты открытия 400 м, примерное время набора максимальной скорости 14 с;
5. «Время полёта» - здесь имеется ввиду, время, которое парашютист собирается находиться в свободном падении;
6. «Spinner» с выборкой значений. Значения в поле spinner несут крайне важное смысловое и математическое значение, т.к. от положения парашютиста в воздухе зависит его скорость полёта, ускорение и время, затраченное на весь полёт. Есть возможность выбрать из 3-х положений, каждое из положений имеет свой определённый коэффициент сопротивления воздуха:
- а. «С максимальным ускорением» - самый маленький коэффициент = 0.0975;
 - б. «Равноускоренное падение» - среднее принятое значение = 0.195;
 - в. «Плашмя, руки разведены» - большое значение сопротивления = 0.2125.

На активности есть 2 поля для вывода ответов. В первое поле выводится значение максимальной скорости в м/с (maxSpeed), а во второе критическое время открытия (timeCritOpen) и высота, на которой будет достигнута максимальная скорость (hightMaxSpeed) с сопутствующими поясняющими фразами.

Все математические вычисления происходят после заполнения всех полей корректными данными и по нажатию на кнопку с парашютистом. Ниже будут представлены различные вариации работы данной активности.

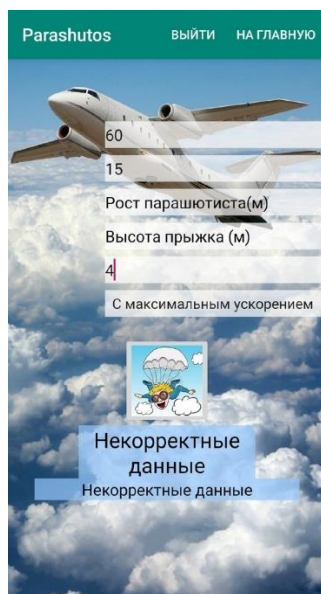


Рис. № Пустые поля для ввода

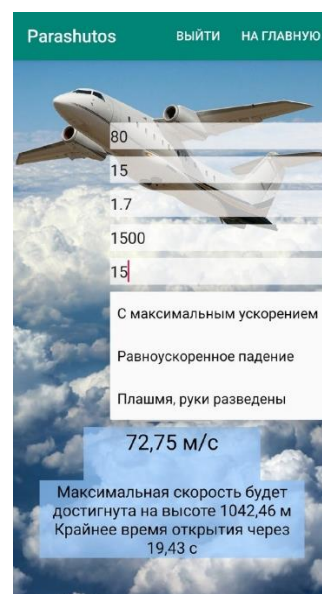


Рис. № Макс. ускорение, корректное время

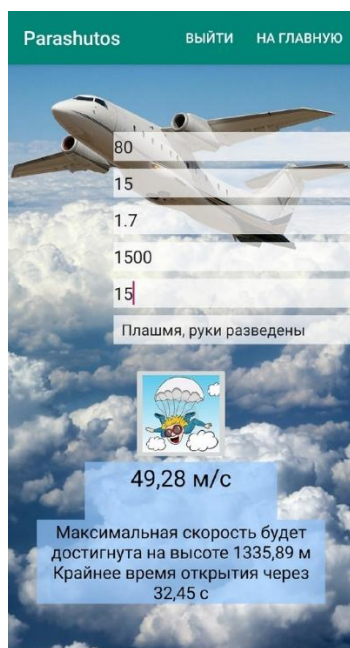


Рис. № Падение плашмя, корректное время

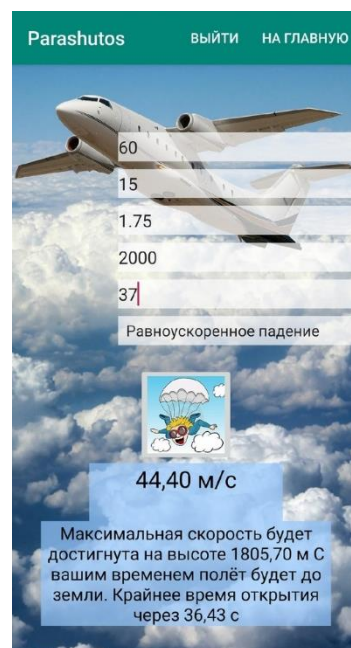


Рис. № Равноускоренное, некорректное время

Формулы, использованные для расчётов (переменные в формулах взяты из кода приложения и не являются общепринятыми):

Формула максимальной скорости парашютиста:

$$\text{maxSpeed} = \text{Math.sqrt}((2 * G) / (p * Cx * St)), (1)$$

G – сила тяжести, Cx – коэффициент сопротивления воздуха, p – плотность воздуха, St – условная площадь тела парашютиста

Формула критического времени открытия (Формула 2):

$$\text{timeCritOpen} = \text{Math.sqrt}(2 * (\text{Heightmax} - h_u) / a)$$

Heightmax – высота прыжка, h_u – установленное значение критической высоты открытия, a – ускорение парашютиста с учётом сопротивления ($a = g - g * (p * Cx * St)$)

Формула высоты набора максимальной скорости (Формула 3):

$$\text{heightMaxSpeed} = \text{Heightmax} - a * \text{Math.pow}(\text{timeHMS}, 2) / 2$$

timeHMS – установленное (для каждого промежутка высот) среднее время набора максимальной скорости

На данной активности присутствует меню с двумя вариантами, а именно «Выйти» - данный пункт перенаправляет на активность «Регистрации/авторизации» и пункт «На главную», который позволяет вернуться на «Главную страницу».

3.4.5. О себе (информация о пользователе)

В этой активности пользователь может редактировать информацию о себе. Как описывалось в пункте 2.1.1 поля уже заполнены данными о пользователе. В случае, если пользователь ни разу их не заполнял, в любом случае в верхнем поле, которое всегда является неактивным к заполнению, будет выведен никнейм пользователя, который он написал при регистрации. При входе в активность все 8 текстовых полей неактивны, при нажатии на кнопку «Редактировать», активность появляется у всех полей, кроме первого (никнейма). Поля: «Имя», «Фамилия», «Отчество», «Дата рождения», «Родной город», «Любимая дрозона», «Количество прыжков» (рисунок 2.5.1).

При заполнении поля «Дата рождения» выполняется проверка на корректность даты, необходимо, чтобы она соответствовала виду «дд.мм.гггг», если дата некорректна, выведется ошибка об этом и подсказка, какого вида должна быть дата.

При заполнении всех необходимых полей, пользователь, для того, чтобы данные записались в БД, должен нажать кнопку «Сохранить». Алгоритм её работы:

- 1) Данные готовятся для отправки по HTTP на сервер с одновременной проверкой даты на корректность (если поле пустое, то пустым оно и останется);
- 2) Данные отправляются на сервер для обновления таблицы «Информация», ожидается статус от сервера
 - а. Если статус положительный, данные сохраняются в таблицу;
 - б. Если сервер возвращает ошибку, она выводится на экран, сохранения данных не происходит.

На активности присутствует также кнопка «Отмена», которая работает как выход на «Главную страницу».

На активности присутствует меню с двумя вариантами, а именно «Выйти» - данный пункт перенаправляет на активность «Регистрации/авторизации» и пункт «На главную», который позволяет вернуться на «Главную страницу».

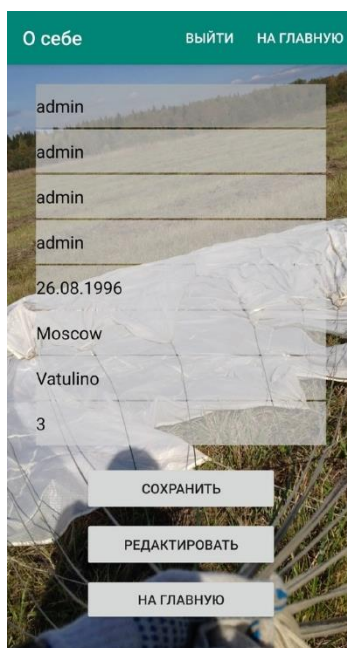


Рис. № Активность «О себе» с заполненными полями

3.4.6. История

Данная активность основывается на встроенном конструкторе в Android Studio `Scrolling_activity`. Был незначительно изменён её дизайн и соответственно текст, выведенный на экран пользователя является историей создания парашютного спорта.

Данная активность в дальнейшем будет дорабатываться и расширяться другими источниками информации. На рисунке 2.6.1. представлен вид активности «История»:

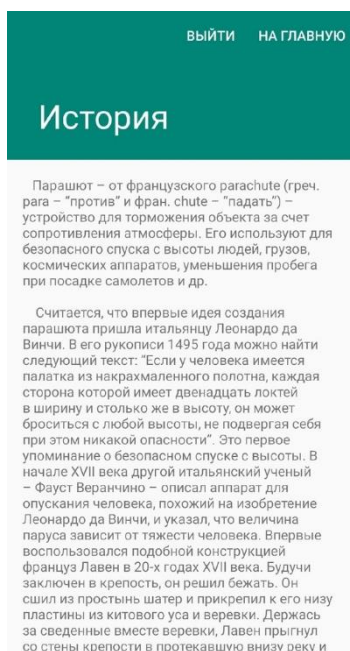


Рис. № Активность «История»

На данной активности присутствует меню с двумя вариантами, а именно «Выйти» - данный пункт перенаправляет на активность «Регистрации/авторизации» и пункт «На главную», который позволяет вернуться на «Главную страницу».

3.4.7. Карта с аэродромами

Карта с аэродромами (на данный момент не реализована в настоящем приложении никак, есть в качестве отдельного приложения для изучения методов работы с картой (реализовано в стандартном конструкторе Android Studio для работы с картами), API ключ соответственно имеется).

3.4.8. Календарь прыжков

Календарь прыжков (на данный момент не реализовано в приложении, рассматривается как потенциальная дополнительная активность).

3.5. Тестирование и отладка приложения

3.6. Руководство пользователя

Заключение

Список используемой литературы

Приложение