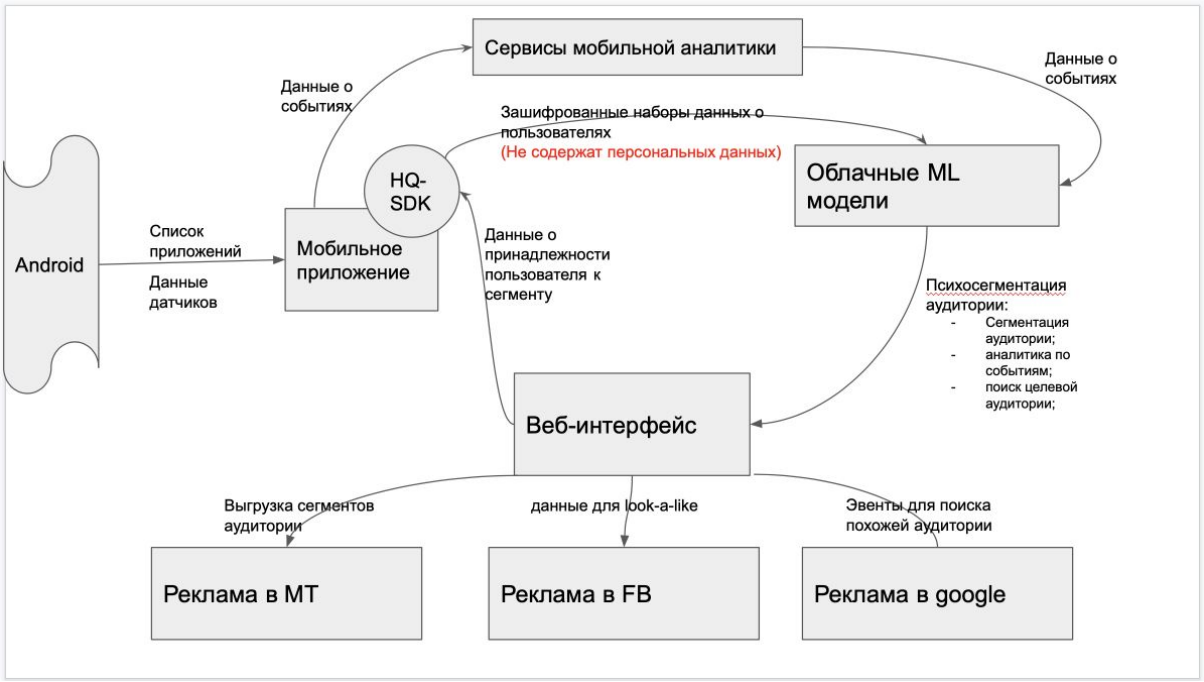
HUMANTEQ.

Инструкция по интеграции HQ SDK в мобильное приложение

Общая схема работы продукта

HQ-Monitor состоит из следующих частей:

- SDK, которое встраивается в мобильное приложение и отвечает за сбор и первичную обработку данных.
- ML моделей, которые обрабатывают полученную от SDK информацию в нашем облаке, опираясь на данные, собранные нашей научной командой.
- Веб-интерфейса, в котором отображается информация и который позволяет настраивать взаимодействие с различными внешними сервисами.



Порядок работ для интеграции SDK в мобильное приложение

Для интеграции системы аналитики пользователей с использованием психографики необходимо провести следующие действия:

1) Добавить в код мобильного приложения на android код нашего SDK
2) Настроить эвенты для работы для дальнейшей сегментации пользователей

Интеграция SDK в мобильное приложения

Подготовка к работе

1. Мигрировать на androidx
2. Добавить в project level build.gradle:

```
allprojects {
    repositories {
        google()
        jcenter()
        maven {
            url 'https://dl.bintray.com/humanteqdev/SWSdk/'
        }
    }
}
```

3. Добавить в application level build.gradle в dependencies:

```
implementation 'ru.livli:swsdk-core:0.0.1-alpha17'
```

В стартовую активити или в Application добавить

```
        SWSdk.init(this,
                "api-key",
                false,
                new SWSdk.Callback() {

             @Override
                    public void onError(@NotNull Throwable exception) {
                        Log.e("onError", "failed " +
exception.getLocalizedMessage());
                    }

                    @Override
                    public void onSuccess(@NotNull SWImpl instance) {
                        Log.e("onSuccess", "ok");
                    }
                });


SWSdk.init(this, "api-key").invokeOnCompletion {
            if(it != null)
               Log.e("onError", "failed " + it.getLocalizedMessage());
            else
               Log.e("onSuccess", "ok");
        }
```

Инициализация

Модули

Сбор статистики по приложениям

1. Подключить зависимость **implementation 'ru.livli:swsdk-apps:0.0.1-alpha17'**
2. Начать сбор:

```
instance.start(new InstalledApplicationsCollector(context)); // сбор установленных приложений
instance.start(new ApplicationsStatisticsCollector(context)); // сбор статистики использования
```

Дополнительные методы

- `SWSdk.setGender(String gender); // Задать пол клиента (опционально, повышает точность работы SDK)`
- `SWSdk.setBirthyear(int birthyear ); // Задать год рождения (опционально, повышает точность работы SDK)`
- `SWSdk.setId(String opaqueId); // передает внутренний ID пользователя, необходим для работы с API и сопоставления пользователя в HQ SDK с внутренними системами.`

Работа с событиями

Вариант 1

Подключить приложение к Firebase на тарифе “Blaze” , фиксировать в нем интересные события.
Подключить BiqQuery. (Аналитика будет проходить в ручном режиме).

Вариант 2

Фиксировать интересующие события с помощью метода `SWSdk.logEvent` Hq Sdk.

```
Map<String, Object> map = new HashMap<>();
        map.put("property1", "test1");
        map.put("property2", "test2");
SWSdk.logEvent("event_name", map)
```