**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ**

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**

**«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук

Образовательная программа бакалавриата «Программная инженерия»

|  |  |
| --- | --- |
| СОГЛАСОВАНО  Научный руководитель,  приглашённый преподаватель ОП «Программная инженерия» департамента программной инженерии факультета компьютерных наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Г.М. Сосновский  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г. | УТВЕРЖДАЮ  Академический руководитель  ОП «Программная инженерия»  профессор департамента программной инженерии, канд. техн. наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В. В. Шилов  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г. |

|  |  |
| --- | --- |
| Подп. и дата |  |
| Инв. № дубл. |  |
| Взам. Инв. № |  |
| Подп. и дата |  |
| Инв. № подл. | **RU.17701729.04.05-01 81 01-1-ЛУ** |

**«HSE APPLE» – ОБРАЗОВАТЕЛЬНАЯ СОЦИАЛЬНАЯ СЕТЬ**

**ДЛЯ НИСа ПО iOS РАЗРАБОТКЕ**

**Пояснительная записка**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.04.05-01 81 01-1-ЛУ**

Исполнитель

Изображение выглядит как текст

Автоматически созданное описаниестудент группы БПИ204

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Туракулов И. У. /

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022г.

УТВЕРЖДЕН

RU.17701729.04.05-01 81 01-1-ЛУ

|  |  |
| --- | --- |
| Подп. и дата |  |
| Инв. № дубл. |  |
| Взам. Инв. № |  |
| Подп. и дата |  |
| Инв. № подл. | **RU.17701729.04.05-01 81 01-1-ЛУ** |

**«HSE APPLE» – ОБРАЗОВАТЕЛЬНАЯ СОЦИАЛЬНАЯ СЕТЬ**

**ДЛЯ НИСа ПО iOS РАЗРАБОТКЕ**

**Пояснительная записка**

**RU.17701729.04.05-01 81 01–1**

**Листов 54**

АННОТАЦИЯ

В данном программном документе приведена пояснительная записка к программе ««HSE Apple» - образовательная социальная сеть для НИСа по iOS разработке»

В разделе «Введение» указано наименование программы, краткое наименование программы и документы, на основании которых ведется разработка.

В разделе «Назначение и область применения» указано функциональное назначение программы, эксплуатационное назначение программы и краткая характеристика области применения программы.

В разделе «Технические характеристики» содержатся следующие подразделы:

1. Постановка задачи на разработку программы;
2. Описание функционирования программы с обоснованием выбора схемы алгоритма решения задачи и возможные взаимодействия программы с другими программами
3. Описание и обоснование выбора метода организации входных и выходных данных;
4. Описание и обоснование выбора состава технических и программных средств.

В разделе «Ожидаемые технико-экономические показатели» указана предполагаемая потребность и экономические преимущества разработки по сравнению с отечественными и зарубежными образцами и аналогами.

Настоящий документ разработан в соответствии с требованиями:

1. ГОСТ 19.101–77 Виды программ и программных документов [1];
2. ГОСТ 19.102–77 Стадии разработки [2];
3. ГОСТ 19.103–77 Обозначения программ и программных документов [3];
4. ГОСТ 19.104–78 Основные надписи [4];
5. ГОСТ 19.105–78 Общие требования к программным документам [5];
6. ГОСТ 19.106–78 Требования к программным документам, выполненным печатным способом [6];
7. ГОСТ 19.404–79. Пояснительная записка. Требования к содержанию и оформлению.

Изменения к Пояснительной записке оформляются согласно ГОСТ 19.603–78 [8], ГОСТ 19.604–78-[9].

**СОДЕРЖАНИЕ**

[**1. ВВЕДЕНИЕ** 5](#_Toc102914852)

[**1.1. Наименование программы** 5](#_Toc102914853)

[**1.2. Краткая характеристика и область назначения** 5](#_Toc102914854)

[**2. НАЗНАЧЕНИЕ РАЗРАБОТКИ** 6](#_Toc102914855)

[**2.1. Функциональное назначение** 6](#_Toc102914856)

[**2.2. Эксплуатационное назначение** 7](#_Toc102914857)

[**3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ** 8](#_Toc102914858)

[**3.1. Постановка задачи на разработку программы** 8](#_Toc102914859)

[**3.2. Описание алгоритма и функционирования программы** 8](#_Toc102914860)

[**3.3. Описание и обоснование выбора состава технических и программных средств** 38](#_Toc102914861)

[**4. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ** 39](#_Toc102914862)

[**4.1. Ориентировочная экономическая эффективность** 39](#_Toc102914863)

[**4.2. Предполагаемая потребность** 39](#_Toc102914864)

[**4.3. Экономические преимущества разработки по сравнению с отечественными и зарубежными аналогами** 39](#_Toc102914865)

[**ПРИЛОЖЕНИЕ 1** 41](#_Toc102914866)

[**ПРИЛОЖЕНИЕ 2** 43](#_Toc102914867)

[**ПРИЛОЖЕНИЕ 3** 44](#_Toc102914868)

[**ПРИЛОЖЕНИЕ 4** 46](#_Toc102914869)

# **ВВЕДЕНИЕ**

## **Наименование программы**

Наименование программы – ««HSE Apple» – образовательная социальная сеть для НИСа по iOS разработке».

Наименование программы на английском языке – ««HSE Apple» – educational social network for a research seminar on iOS development».

Наименование программы для пользователя – «HSE Apple».

## **Краткая характеристика и область назначения**

««HSE Apple» образовательная социальная сеть для НИСа по iOS разработке» - мобильное приложение на операционной системе Google Android предоставляется для курса НИСа для студентов НИУ ВШЭ.

# **НАЗНАЧЕНИЕ РАЗРАБОТКИ**

## **Функциональное назначение**

Функциональным назначением приложения «HSE Apple» предоставление возможности пользователям:

Обмен/Просмотр учебными материалами

Прохождение тестов

Просмотр событий и ведомости с оценками.

Выполнение заданий

В сервисе присутствует разделение на три роли пользователей:

|  |  |  |  |
| --- | --- | --- | --- |
| Тип ресурса | Роль пользователя | | |
| Студент | Ассистент | Преподаватель |
| Чаты | чтение | создание  чтение  изменение  удаление | создание  чтение  изменение  удаление |
| Событие | чтение | чтение  изменение  удаление | создание  чтение  изменение  удаление |
| Ведомости успеваемости | чтение | чтение  изменение | создание  чтение  изменение  удаление |
| Права доступа заданиям | чтение | чтение  изменение | создание  чтение  изменение  удаление |
| Права доступа к системе приложения | - | чтение  изменение  удаление | создание  чтение  изменение  удаление |

## **Эксплуатационное назначение**

Данное приложение предназначено для студентов и преподавателей «НИУ ВШЭ», которые проходят тесты, выполняют лабораторные работы, общаются с пользователями, просматривают события и ведомость с оценками. Также предоставляются две локализованные версии: версия на русском языке и на английском языке для мобильных устройств на базе операционных систем Android.

# **ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ**

## **Постановка задачи на разработку программы**

Задачей является разработка клиентское приложение для операционной системы Google Android

* + 1. Задачи работы (Android клиент)

1. Реализовать авторизацию через ЕЛК;
2. Реализовать получение, хранение и обновление JWT (Javascript Web Tokens);
3. Разработать чат для пользователей курса;
4. Реализовать создание заданий для пользователей курса;
5. Реализовать создание новостей для участников с ролями “Преподаватель” и “Ассистент”
6. Реализовать взаимодействие с серверной частью через HTTP запросы;
7. Реализовать возможность ввода или изменения текстовых данных с валидацией;
8. Реализовать загрузку или изменение изображения через POST и PUT запросы;
9. Реализовать новостную ленту для пользователей курса;
10. Реализовать прохождение заданий для пользователей курса;
11. Реализовать выбор языковой локализации приложения;

## **Описание алгоритма и функционирования программы**

* + 1. Постановка задачи на разработку программы

#### **Структура мобильного приложения**

1. Загрузочная страниц



Рисунок Стартовая страница (Альбомная ориентация)



Рисунок 2 Стартовая страница (Портретная ориентация)

При запуске приложения, пользователю отображается загрузочная страница и в это время проверяется наличие refresh-токена, который был взят с локального хранилища устройства. Если токен присутствует пользователь автоматически переходит на страницу “Выбор курсов” иначе на страницу «Авторизация».

1. Авторизация

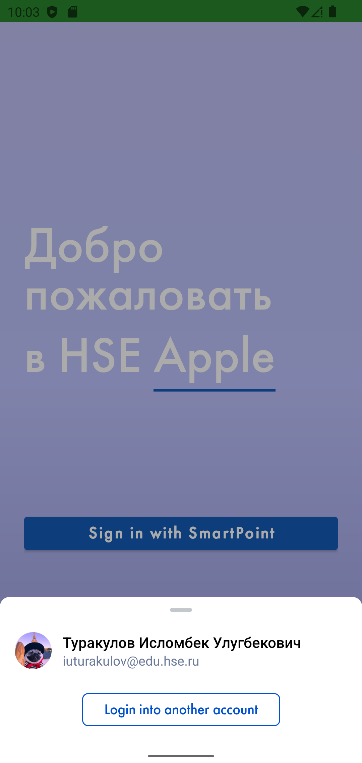


Рисунок 3 Авторизация

При переходе на страницу «Авторизация» Рис. 3 пользователю предлагается авторизация через ЕЛК (Smart Point). При нажатии на кнопку выводятся компоненты Bottom Sheet (<https://developer.android.com/reference/com/google/android/material/bottomsheet/BottomSheetBehavior>) (Рис. 3) панель для авторизации через аккаунт. Для авторизации используется уникальный client\_ID c redirect\_URI от HSE Digital и библиотеки для работы с ними: HSE Auth (<https://github.com/hseapp/HSEAuth-Android>) и HSE Core (<https://github.com/hseapp/HSECore-Android>).

AuthHelper.login(activity, requestcode)

…..…

override fun onActivityResult (requestCode: Int, resultCode: Int, data: Intent?) {  
 super.onActivityResult(requestCode, resultCode, data)  
 if (requestCode == *REQUEST\_LOGIN*) {  
 if (resultCode != *RESULT\_OK* || data == null) {  
 return  
 }  
 *ACCESS\_TOKEN* = data.getStringExtra(AuthConstants.KEY\_ACCESS\_TOKEN)!!  
 *REFRESH\_TOKEN* = data.getStringExtra(AuthConstants.KEY\_REFRESH\_TOKEN)!!  
 Timber.d("Token got, success")  
 Toast.makeText(this, "Success", Toast.*LENGTH\_LONG*).show()  
 initializeToken()  
 }  
}

Листинг 1. Авторизация пользователя через ЕЛК.

При успешной авторизация приложение получает access и refresh токен, которые нужны для регистрации на сервер.

1. Выбор курса

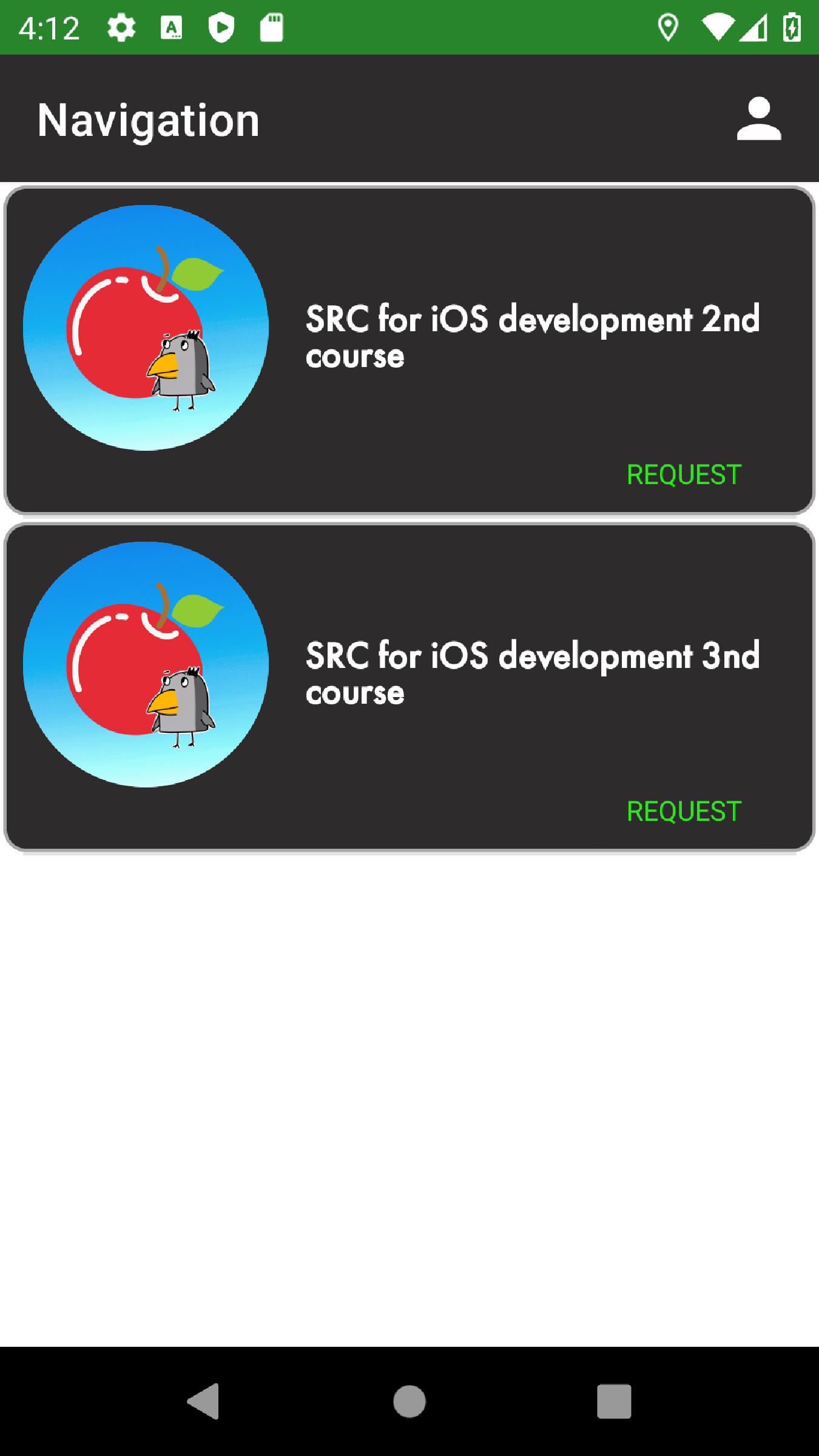


Рисунок 4 Выбор курса

При успешной авторизации пользователь переходит на страницу «Выбор курса» Рис. 4. При нажатии на кнопку “Request”, происходит подача заявки на курс или переход на страницу “Главная страница”. Пользователь ожидает подтверждения заявки от Преподавателя.

1. Главная страница приложения

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 5 Чаты

В главной странице имеется меню выбора: Чаты, Тесты, Новости и Профиль. По-умолчанию выбирается страница «Чаты» Рис.5. В жизненном цикле фрагмента (<https://developer.android.com/guide/fragments/lifecycle>) была подключена библиотека CometChatUIKit (<https://github.com/cometchat-pro/android-kotlin-chat-ui-kit>) для корректной работы функционала. Данная библиотека позволяет создать готовые UI компоненты для приложения и классы с готовым функционалом, например вывести историю сообщений, пользователей, группы.

Пользователь может ввести в поисковой панели название сообщения, группы или документа. Вводимые данные валидируются и показывается результат поиска.

При нажатии на кнопку, которая находится в правом верхнем углу, открывается новая страница просмотра всех доступных участников и групп.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок Выбор групп

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 7 Просмотр участников

Страница содержит View Pager (<https://developer.android.com/reference/androidx/viewpager/widget/ViewPager>) в котором находятся две страницы-фрагмента: «Просмотр участников» и «Выбор доступных групп». Загрузочной страницей по умолчанию является «Просмотр участников» Рис. 7. При вводе в поисковой панели текста, происходит запрос на получение похожего по значению слова.

val request =  
 UsersRequestBuilder().setSearchKeyword(s).setLimit(100).build()  
request.fetchNext(object : CallbackListener<List<User?>?>() {  
 override fun onSuccess(users: List<User?>?) = if (role.*isNotEmpty*()) {  
 val filteredItems = users?.*filter* **{ it**?.*role* != role **}** recyclerViewUserList!!.searchUserList(filteredItems)  
 } else {  
 recyclerViewUserList!!.searchUserList(users)  
 }  
  
 override fun onError(e: CometChatException) {  
 ErrorMsgUtils.showChatErrorMessage(*context*, e.*code*)  
 }  
})

Листинг 2. Авторизация пользователя через ЕЛК.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 8 Групповой чат

Для отображения сообщений используется библиотека Shimmer Facebook RecyclerView (<https://github.com/facebook/shimmer-android>). При отключении интернета список сообщений становится недоступным и показывается загрузочная компонента сообщений. Данная библиотека даёт хорошую оптимизацию при большом потоке сообщений, например при 1000 сообщений простой список не сможет справиться в рендеринге, теряются огромное количество кадров.

При долгом нажатии на кнопку, которая расположена в левом нижнем углу экрана появляется всплывающее меню со следующими возможностями:

1. Сфотографировать и отправить
2. Получить фото или видео с галереи
3. Отправить файл(-ы) или документ(-ы)
4. Поделиться геопозицией
5. Создать опросник

Изображение выглядит как текст

Автоматически созданное описание

Рисунок Отправка данных. BottomSheet (Альбомный режим)

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 10 Отправка данных. BottomSheet (Портретный режим)

После нажатия на описанные кнопки Рис. 9-10 происходит действие между самой системой Android. При завершение системных окон Android происходит закрепление действий.

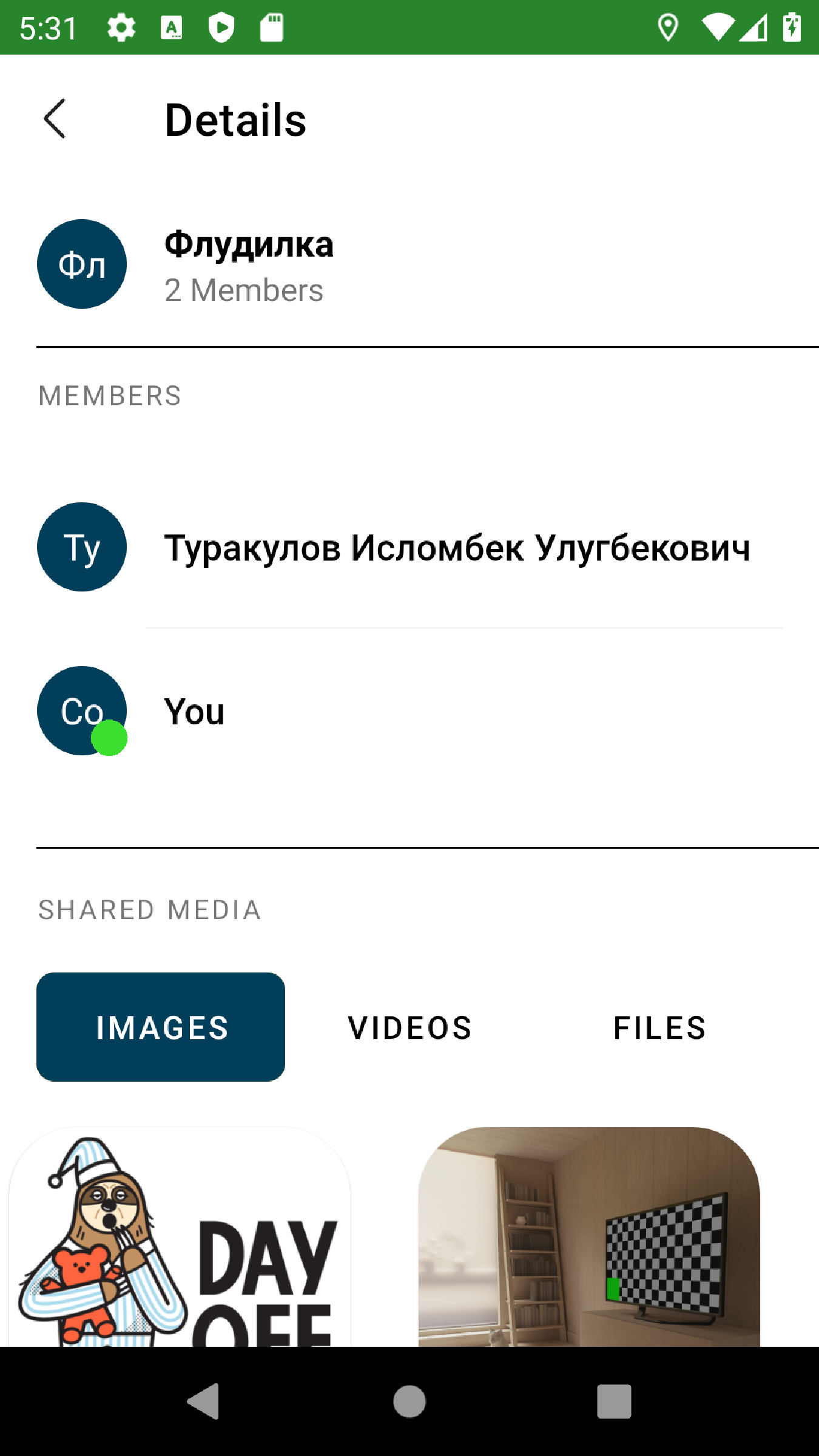


Рисунок Информация о группе (вкладка изображение)

При нажатии на кнопку, которая расположена в правом верхнем углу (как информация) происходит переход на новую страницу с информацией о группе Рис. 11-12. В меню, которая расположена после информации об участников “Shared Media” меняется вкладка при нажатии на выбранную и соответственно при смене вкладок показываются разные данные.

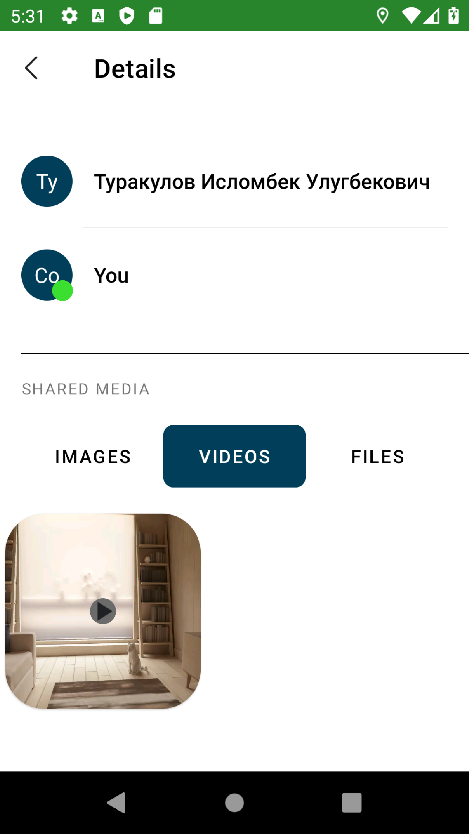


Рисунок Информация о группе (вкладка видео)

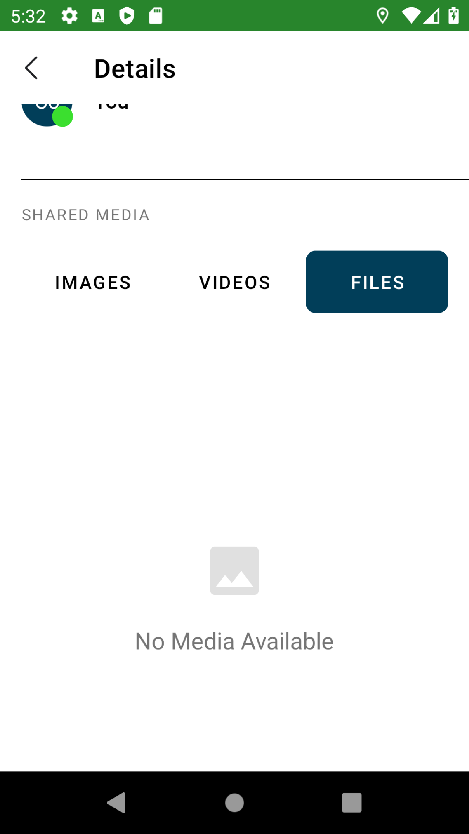


Рисунок 13 Информация о группе (вкладка файлы)

Кнопкой возвращения в чат, служит кнопка, которая расположена в левом верхнем углу. Пользователь во время общения с другими участниками группы, может отправить эмодзи, гиф анимации и стикеры. Для этого используются официальная библиотека EmojiCompat от (<https://developer.android.com/jetpack/androidx/releases/emoji>).

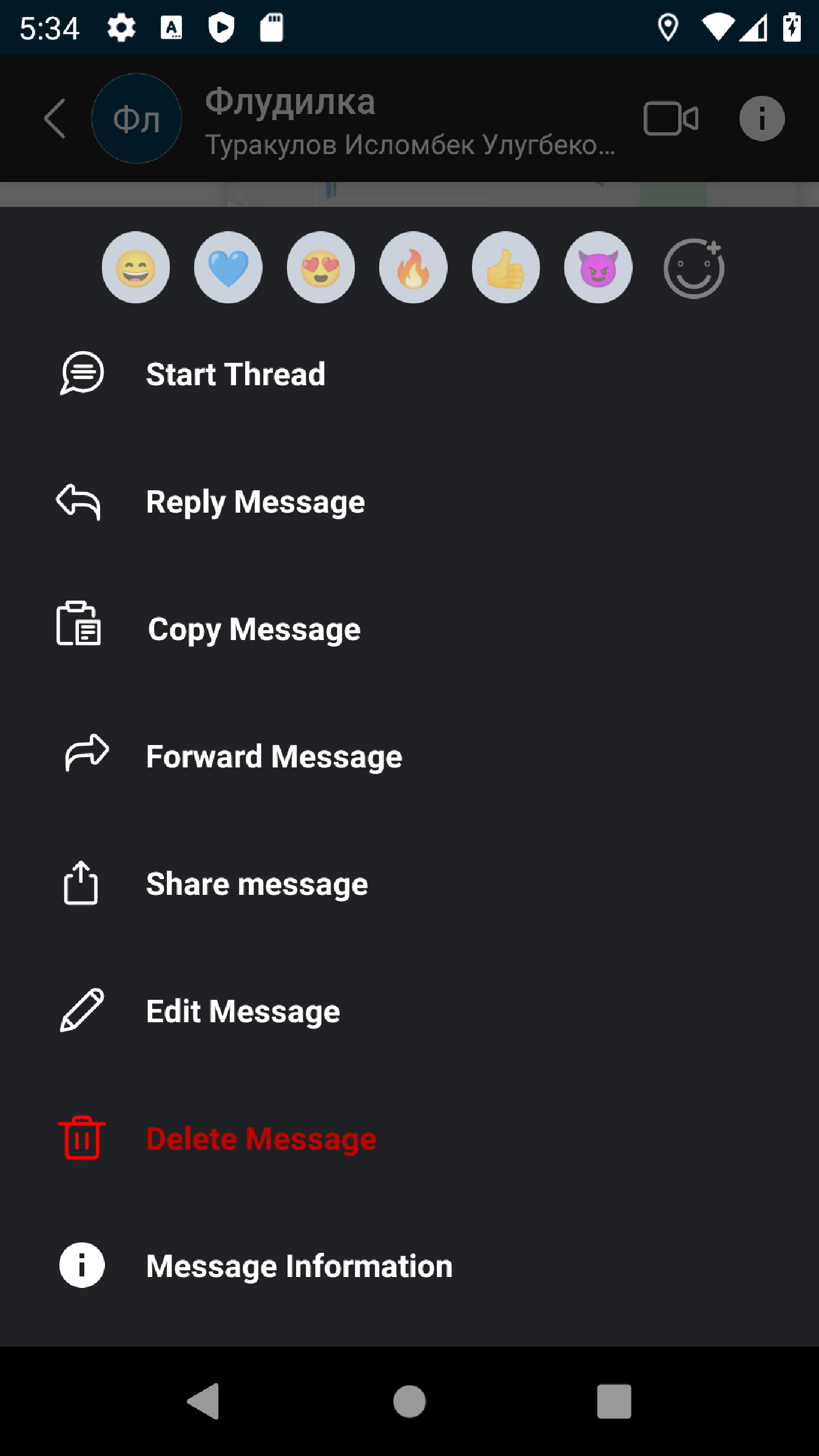


Рисунок 14 Операции над сообщением

При долгом нажатии на сообщение появляется всплывающее меню Рис. 16 выбора дополнительных действий:

1. Реакция на сообщение смайликами Рис. 15 или Рис. 17. Реакции (смайлики) на сообщения их огромное количество благодаря использованию библиотеки EmojiCompat. Выбор реакции смайликов можно расширить, показывается всплывающее окно, которое содержит список смайликов, которые доступны в установленной операционной системе приложения.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок Реакция на сообщение

1. Начать дискуссию – при нажатии на данное действие, открывается страница для дополнительного ответа на сообщение, которое было нажато пользователем. Пользователь может поделиться сообщением, оставить реакцию на сообщение Рис. 16. Дискуссию может начать каждый пользователь. В окне дискуссии можно отправить данные, которые были описаны в Рис. 10. В сообщениях дискуссии аналогично можно выполнить операции над сообщениями, которые были описаны в Рис. 14.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 16 Начать дискуссию

1. Ответить на сообщение – при нажатии на данное действие, пользователь может ответить на сообщение создавая цепочку ответов Рис. 17. При ответе на сообщение, пользователям приходит уведомление о новом сообщении или если уведомления были выключены, показываются в странице «Чаты» как непрочитанное.

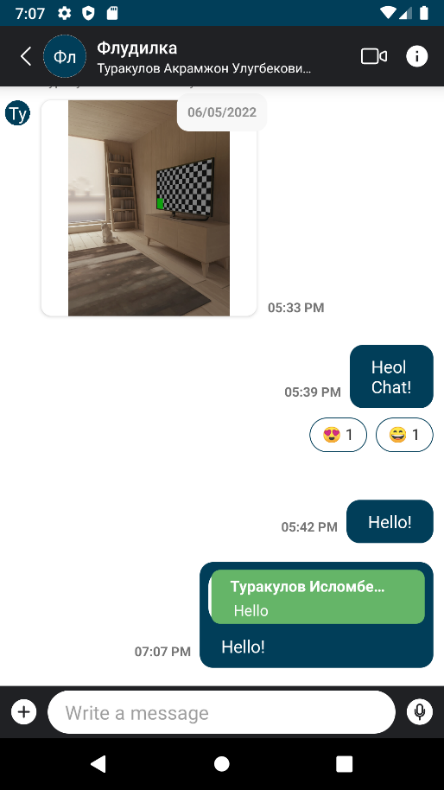


Рисунок 17 Ответить на сообщение

1. Скопировать сообщение – при нажатии на данное действие сообщение отправляется в буфер обмена. Сообщение копируется для любых данных, например при копии изображения в буфере обмена остаётся изображение с информацией.
2. Переслать сообщение – при нажатии на данное действие открывается страница выбора чатов Рис. 18. При пересылке сообщений учитывается сообщение сообщения, является ли данное сообщение другого собеседника.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 18 Переслать сообщение

При нажатии на кнопку “Send”, отправляется сообщение на все выбранные чаты Рис. 23.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 19 Пересланное сообщение

1. Поделиться сообщением - при нажатии на данное действие открывается страница выбора отправки Рис. 19. При выборе приложения “HSE Apple”, показывается список чатов, описанное выше Рис. 18. Действие происходит системой Android, при завершении действия приложение переходит под фоновый режим, но остаётся работающим.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 20 Поделиться сообщением

1. Изменить сообщение - при нажатии на данное действие показывается маленькое окошка с подсказкой о том, что данный текст изменяется Рис. 20. При изменении сообщения пользователям не уведомляется данное действие, вместо этого показывается статус “Изменено” под сообщением.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 21 Изменение текста

1. Удалить сообщение – при нажатии на данное действие сообщение удаляется из чата Рис. 21. При выполнении данного действия вместо сообщения с данными показывается уведомление о том, что сообщение было удалено. Над удалённым сообщением нельзя выполнить операции, которые выполнены в Рис. 14.

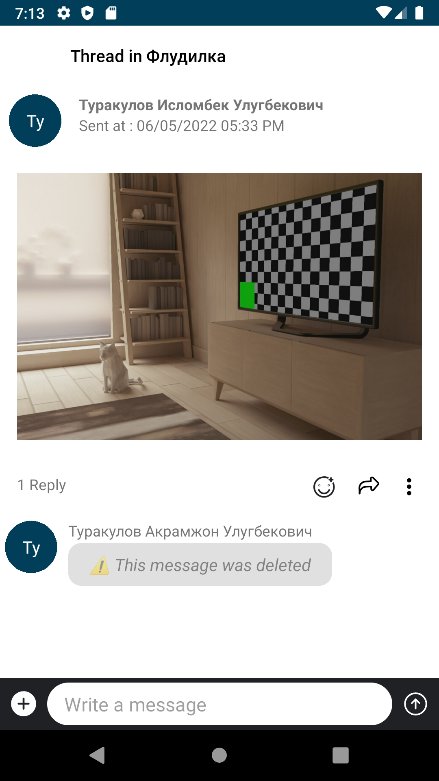


Рисунок 22 Сообщение удалено

1. Информация о сообщении - при нажатии на данное действие показывается информация отправителя сообщения и дата время отправки Рис. 22. Информация о сообщении включается в себя ФИО пользователя, статус и время отправки сообщения. При отправке изображения, видео или файла, показывается информация отправителя и название изображения.

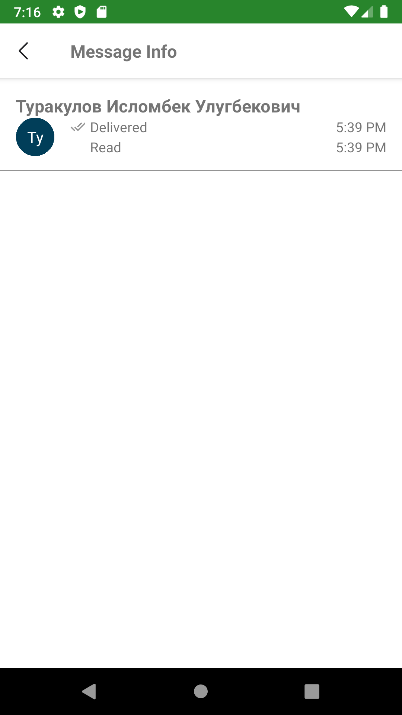


Рисунок 23 Информация о сообщении

Кнопкой возвращения в главную страницу, является кнопка, которая расположена в левом верхнем углу экрана.

1. Страница «Тесты»

На данной странице показываются текущие и завершённые тесты пользователя. Если текущие задания отсутствуют, тогда показывается подсказка с изображением и текстом, аналогично с завершёнными заданиями. Отображение данных подсказок изображена на Рис. 24-25. Задания со статусом “Завершённое” показывают оценку за задание пользователя. Тесты уникальные для каждого курса, то есть если тест доступен для второго курса, значит студенты третьего курса не смогут получить доступ к тестам второго курса.

Изображение выглядит как текст, снимок экрана, монитор, экран

Автоматически созданное описание

Рисунок Образец завершённых и текущих тестов

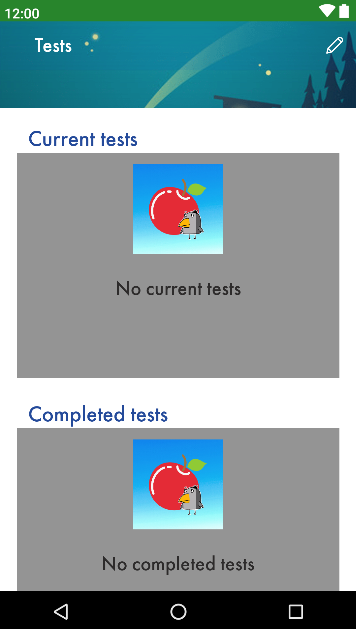


Рисунок 25 Отсутствуют тесты

Кнопка, которая расположена в правом верхнем углу доступна только пользователю с ролью “Преподаватель”. При нажатии на данную кнопку происходит переход на страницу «Создать задание» Рис. 26.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 26 Создание задания

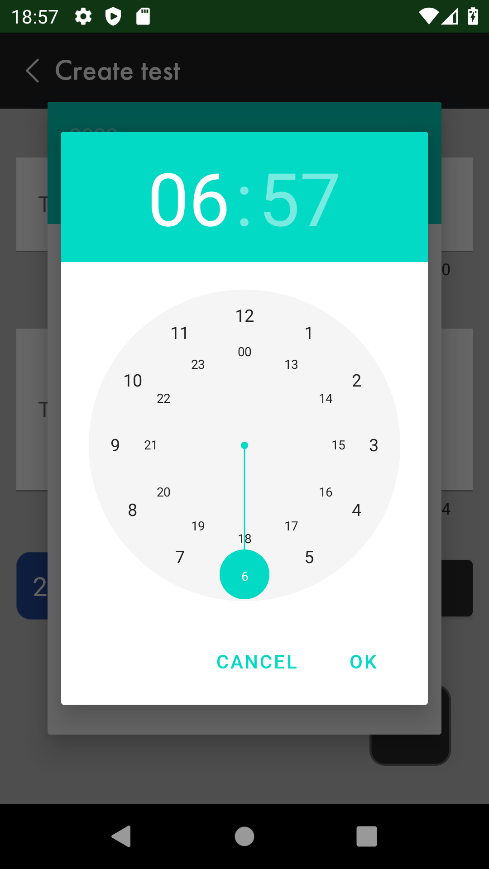


Рисунок 27 Выбор даты и времени дедлайна задания

При нажатии на кнопку “Change Deadline”, происходит системный вызов выбора даты и времени дедлайна для задания. Для корректной работы выбора используется библиотека Material Date Time Picker (<https://github.com/wdullaer/MaterialDateTimePicker>) Рис. 27.

Вводимые данные пользователем проверяются на корректность и дедлайн даты и времени.

fun TextInputEditText.validateDescription(): Boolean {  
 if (!this.*text*.*isNullOrEmpty*()) {  
 this.setText(this.*text*?.*trim*())  
 return this.*text*!!.length in 0..256  
 }  
 return false  
}  
  
fun TextInputEditText.validateTitle(): Boolean {  
 if (!this.*text*.*isNullOrEmpty*()) {  
 this.setText(this.*text*?.*trim*())  
 return this.*text*!!.length in 0..50  
 }  
 return false  
}

Листинг 3. Проверка данных на ввод.

Кнопкой сохранения данных является кнопка, которая расположена в правом нижнем углу. После нажатия на данную кнопку показывается Toast об информации отправки и закрывается страница. Задание сохраняется как GitHub ссылка на репозиторий или на любые другие файловые хранлища

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 28 Информация о задаче (режим чтения)

При нажатии на тест, выбранный из списка доступных Рис. 24, происходит отправка объекта TaskEntity на страницу «Просмотр задания» Рис. 28.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 29 Информация о задаче (режим редактирования)

Кнопки, которые расположены в компоненте Toolbar (<https://developer.android.com/reference/android/widget/Toolbar>) , а именно функции “Изменение” и “Удаление” доступны для пользователей с ролью “Ассистент” и “Преподаватель” и во время просмотра теста после создания. При нажатии на кнопку “Изменение”, открывается доступ к редактированию текстовых полей и даты времени Рис. 29. При сохранении данных является кнопка, которая расположена в нижнем центре экрана.

Изображение выглядит как текст, снимок экрана, монитор, электроника

Автоматически созданное описание

Рисунок 30 Образец отображения события

При нажатии на данную кнопку выводится Toast с информацией об ошибке. При нажатии на кнопку “Удалить” происходит удаление поста с последующим закрытием страницы.

1. Страница «Новости»

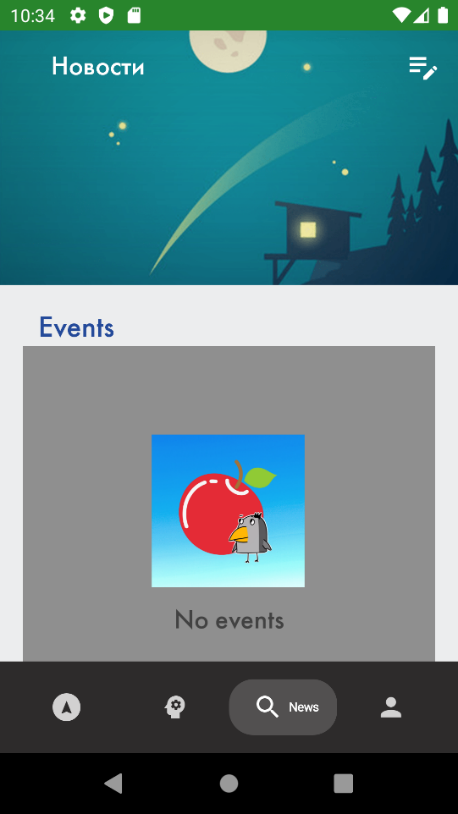


Рисунок 31 Отсутствуют новости

При открытии страницы показывается все доступные новости для определенного курса. При нажатии на новостной компонент со списка новостей, происходит открытие страницы «Информация о событии». Где показывается заголовок, описание и изображение новостного события Рис. 32.

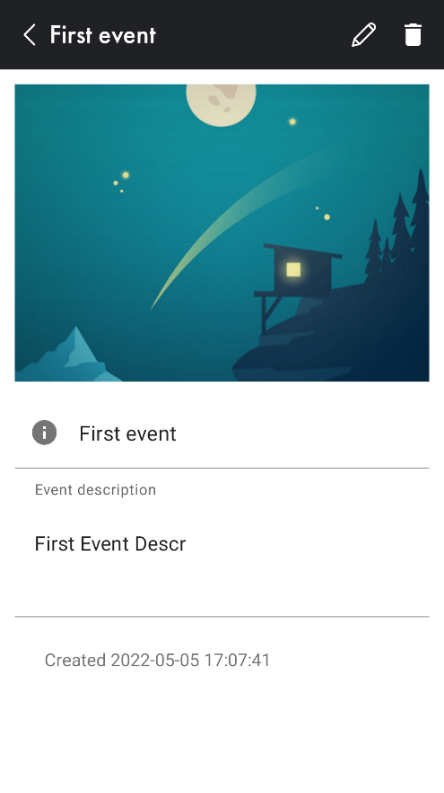


Рисунок 32 Информация новостного события

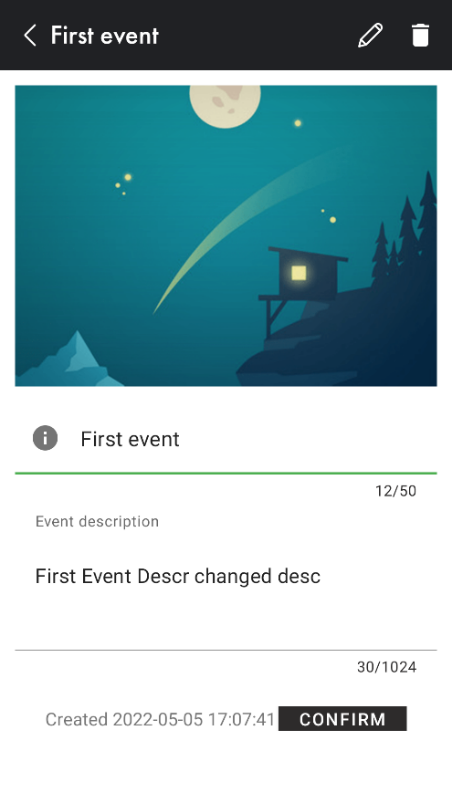


Рисунок 33 Редактирование события

Для пользователя с ролью “Ассистент” или “Преподаватель” доступны следующие функции:

1. Изменение - приложение даёт доступ к редактированию заголовка и описания новостного события Рис 33. Открывается доступ к изменению заголовка и описания. Пользователь при сохранении изменений нажимает на кнопку “Confirm”, иначе сбрасываются изменения, внесенные пользователем.
2. Удаление – показывается всплывающая панель с дополнительным выбором удаления. При подтверждении удаляется пост и закрывается страница Рис. 33.

private fun fieldOptionsInitializer(flag: Boolean)

editMode = flag  
newsTitleItem.isClickable = flag  
newsDescriptionItem.isClickable = flag  
newsTitleItem.isFocusable = flag  
newsDescriptionItem.isFocusable = flag  
newsTitleItem.isFocusableInTouchMode = flag  
newsDescriptionItem.isFocusableInTouchMode = flag  
if (!flag) {  
 newsTitleItem.tag = newsTitleItem.keyListener  
 newsDescriptionItem.tag = newsDescriptionItem.keyListener  
}  
textInputLayoutDesc.isCounterEnabled = flag  
textNewsTitleCreate.isCounterEnabled = flag  
confirmButtonEdit.visibility = if (flag) View.VISIBLE else View.GONE

Листинг 4. Проверка пользователя на роль

Кнопка, которая расположена в правом верхнем углу Рис. 32, видна только пользователю с ролью “Преподаватель”. При нажатии на данную кнопку происходит переход на новую страницу «Создать событие» Рис. 34. При вводе некорректного заголовка или описания задания, показывается информация (подсказка) об ошибке. При выборе изображения его можно изменять, трансформировать размер, масштабировать и обрезать.

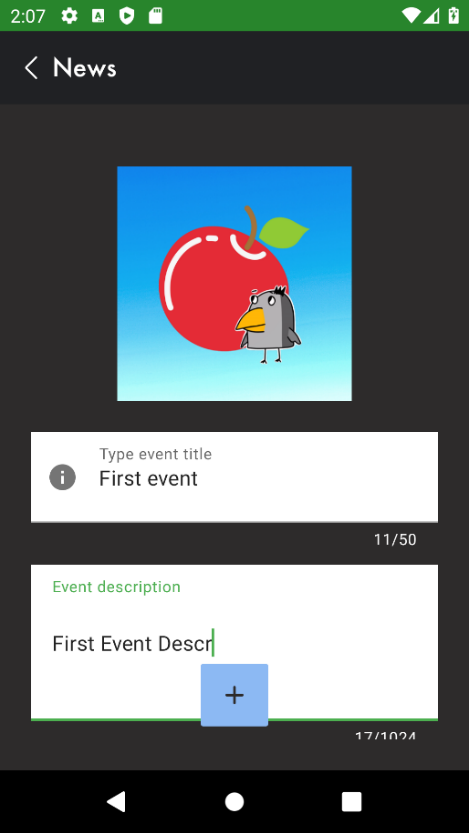


Рисунок 34 Создание события

Изображение выглядит как текст, снимок экрана, монитор

Автоматически созданное описание

Рисунок 35 Выбор изображения

При выборе изображения показывается DialogWindow с выбором изображения Рис. 35. Для данного выбора используется библиотека Image Picker (<https://github.com/Dhaval2404/ImagePicker>). При вводе данных и сохранения на кнопку, которая расположена в нижнем центре экрана, проверяет вводимый текст Листинг 3. При нажатии на кнопку (плюсик) пост сохраняется и закрывается страница.

1. **Страница «Профиль»**

При переходе на данную страницу происходит инициализация профиля и выдача информации, которая была заранее взята во время авторизации пользователя.

Изображение выглядит как текст, электроника, снимок экрана

Автоматически созданное описание

Рисунок 36 Вид от студента/ассистента

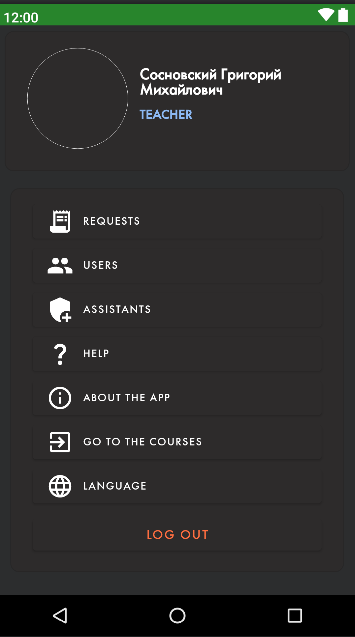


Рисунок 37 Вид от преподавателя

Для пользователя с ролью “Преподаватель” Рис. 37 отображаются расширенные возможности приложения такие как: «Заявки», «Пользователи», «Ассистенты». Для пользователя с ролью “Студент” или “Ассистент”, скрываются расширенные настройки Рис. 36.

При нажатии на кнопку “Заявки”, показывается страница «Список заявок на курс», где списком отображаются доступные заявки на курс.

При действии swipe Right (<https://material.io/design/interaction/gestures.html>) происходит событие отклонения заявки на курс, пользователю высвечивается подсказка о том, что можно отменить данное действие, аналогично с действием swipe Left(<https://material.io/design/interaction/gestures.html>) при выполнении данного события подтверждается заявка пользователя и для этого используется библиотека Recycler View Swipe Decorator (<https://github.com/xabaras/RecyclerViewSwipeDecorator>). Аналогично при нажатии на кнопку “Ассистенты”, показывается страница «Список ассистентов».

При нажатии на кнопку “Help”, происходит переход на почту тех. поддержки приложения, а именно в [letter.hseapple@gmail.com](mailto:letter.hseapple@gmail.com) .

При нажатии на кнопку “Go to the courses”, происходит переход на страницу «Выбор курса».

При нажатии на кнопку “Log out”, происходит переход на страницу «Авторизация» и зануляется Access-токен.

#### **Шифрование данных**

Чат с сообщениями находятся под End-to-End шифрованием. Библиотека CometUIKit позволяет использовать технологию Virgil E3Kit (<https://github.com/VirgilSecurity/demo-e3kit-android>).

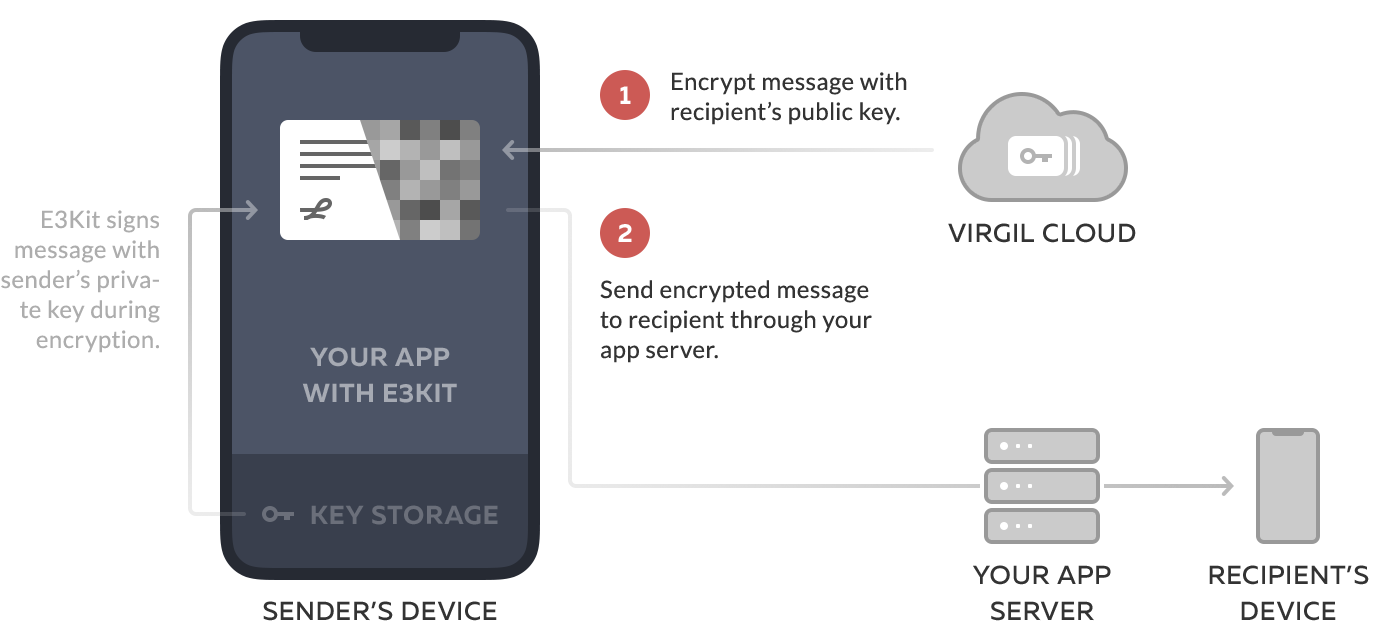


Рисунок Авторизация и кодирование информации

Данная технология использует концепцию криптографии с асимметричным ключом для достижения сквозного шифрования сообщений. Процесс и код для шифрования и дешифрования платформы можно узнать в документации Virgil (<https://developer.virgilsecurity.com/docs/e3kit/end-to-end-encryption/default/#sign-and-encrypt-data>). Пользователи и группы рассматриваются как идентификаторы на Virgil.

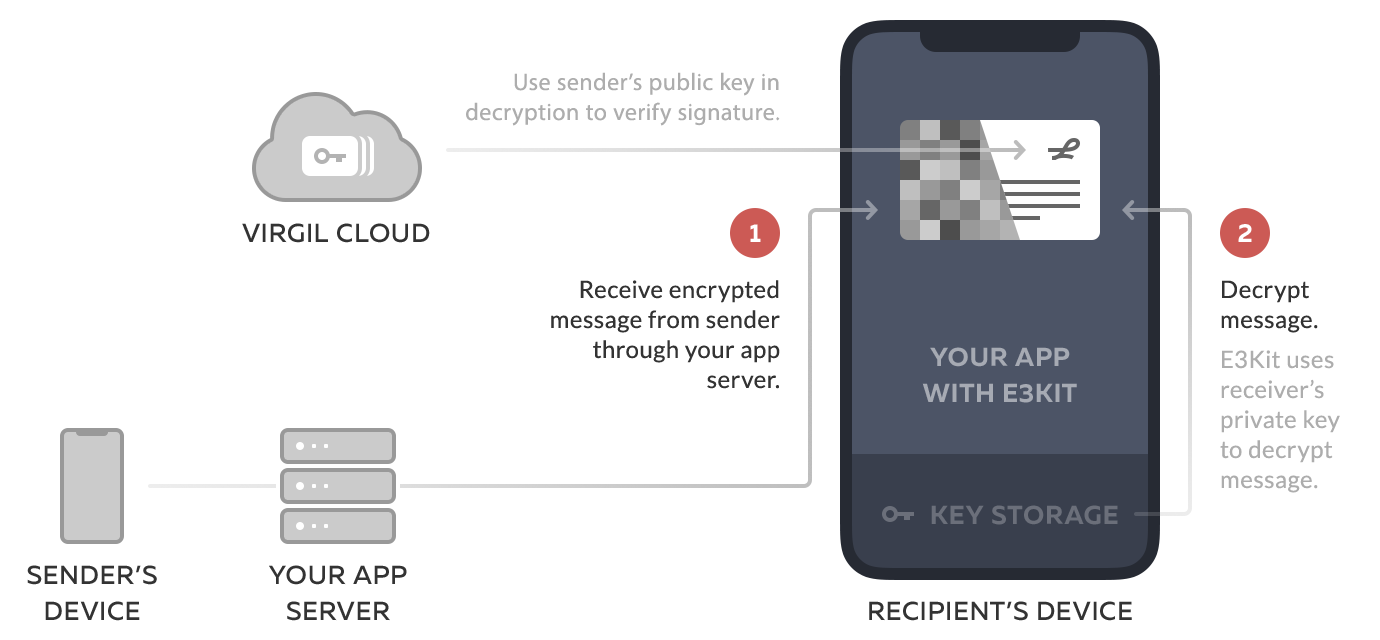


Рисунок Декодирование и отправка данных в приложение

Иная и простая концепция работы данной технологии

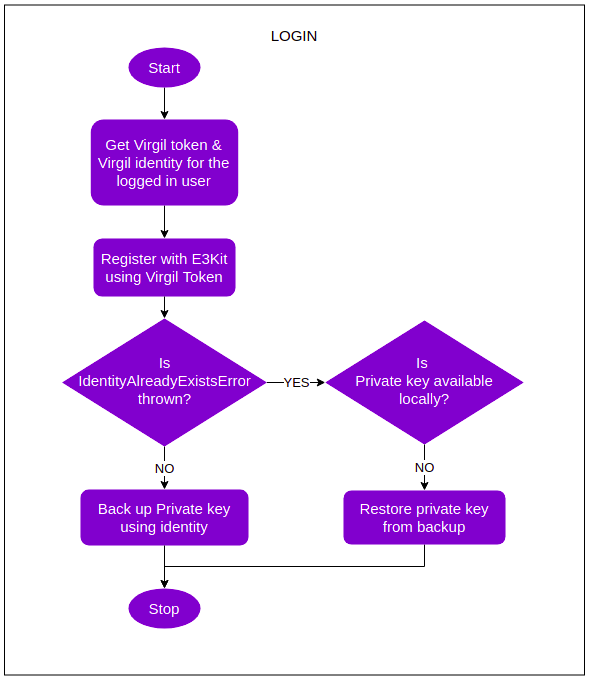


Рисунок Иной график событий с функциями

fun encrypt(text: String, users: FindUsersResult): String {

var result = ""

…..

try {

val times = if (benchmarking) 100 else 1

for (i in 1.. times) {

time += measureTimeMillis {

result = instance().authEncrypt(text, users)

}

}

Timber.w("Encrypted and signed: '$encryptedText'")

} catch (e: Throwable) {

Timber.w("Failed encrypting and signing: $e")

}

return result

}

fun decrypt(text: String, sender: Card): String {

val result: String? = null

…….

try {

val times = if (benchmarking) 100 else 1

for (i in 1.. times) {

time += measureTimeMillis {

result = instance ().authDecrypt(text, sender)

}

}

Timber.w("Decrypted and verified: $result")

} catch (e: Throwable) {

Timber.d("Failed decrypting and verifying: $e")

}

return result

}

Листинг 5. Пример кодирования и декодирования

#### **Взаимодействие и обновление токенов**

Чтобы взаимодействовать с защищённым локальным хранилищем токенов, используется библиотека HSEAuth. Значения хранятся в формате ключ-значение, где в классе библиотеки AccountManager происходит краткое обновление refresh\_token и access\_token.

#### **Получение данных из локального хранилища**

Если приложение находится в оффлайн режиме, подгружаются все ранее скачанные данные. Для кодирования и декодирования изображений используется формат base64.

## **Описание и обоснование выбора состава технических и программных средств**

### **Состав технических и программных средств**

Мобильное приложение с операционной системой Android 9.0 или выше.

### **Обоснование выбора технических и программных средств**

Приложение разрабатывалось под оптимальную рекомендованную версию операционной системы Android 9.0 «Pie» API 28. Так как, исходя из официального сайта API Levels (на 01.05.2022) процент пользователей, кто использует версию 9.0 и выше составляют 81.2%.

# **ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ**

## **Ориентировочная экономическая эффективность**

В рамках данной работы расчёт экономической эффективности не предусмотрен.

## **Предполагаемая потребность**

С данным мобильным приложением удобно совмещать чаты, тесты и новостную ленту в единое целое. Предоставить доступ к материалам курса, которая является ключевой особенностью удобства использования для НИСа.

## **Экономические преимущества разработки по сравнению с отечественными и зарубежными аналогами**

В рамках анализа конкурентов приложения были выделены следующие основные характеристики для сравнения:

* + 1. Функциональность - разные ключевые функции приложения.
    2. Русская локализация
    3. Кроссплатформенность - приложение портируется на разные платформы.
    4. Удобство использования - целевой аудитории удобно взаимодействовать с данным приложением.
    5. Безопасность данных - защищённость персональных данных.
    6. Пользовательский интерфейс (UI) - интуитивно понятный интерфейс для целевой аудитории
    7. Объем используемой базы данных - количество зарегистрированных бенефициаров.

Указанные выше характеристики ранжированы. Это означает, что «Функциональность» имеет наивысший приоритет – 8, «Объем используемой базы» - наименьший, 1.

Процесс поиска аналогов в сети Интернет позволил выявить следующих прямых конкурентов:

* ВКонтакте (международное название — VK) — Российская социальная сеть [10]
* Campus Group - Платформа управления опытом кампуса "все в одном" [11]
* Society App – Приложение для управления событиями университета и организация задач. [12]
* LearningApps - платформа для поддержки обучения и преподавания с помощью небольших общедоступных интерактивных модулей. [13]
* Online Test Pad – платформа для обучения студентов и поддержки интерактивных упражнений. [14]
* ClassMarker – платформа для прохождения тестирований. [15]
* Quizziz – платформа для прохождения головоломок, тестов и задач и общение с преподавателями. [16]
* Coursera - платформа массового онлайн-образования, который основан профессорами информатики Стэнфордского университета [17]

Далее приведена таблица анализа указанных конкурентов по описанным характеристикам. По каждому критерию выставляется оценка по пятибалльной шкале.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Критерии / Название приложения | ВКонтакте [10] | Campus Group [11] | Society App [12] | LearningApps [13] | Online Test Pad [14] | ClassMarker [15] | Quizizz [16] | Coursera [17] | HSE Apple |
| Функциональность | 5 | 3 | 3 | 4 | 4 | 2 | 4 | 5 | 4 |
| Русская локализация | 5 | 0 | 0 | 0 | 3 | 2 | 0 | 5 | 5 |
| Кроссплатформенность | 5 | 4 | 4 | 2 | 2 | 3 | 3 | 5 | 5 |
| Удобство использования | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 5 | 5 |
| Безопасность данных | 4 | 4 | 4 | 3 | 3 | 3 | 4 | 5 | 4 |
| UI | 5 | 4 | 3 | 4 | 4 | 2 | 4 | 5 | 5 |
| Доступность | 5 | 2 | 2 | 3 | 3 | 3 | 4 | 5 | 4 |
| Объем базы данных | 4 | 3 | 2 | 3 | 3 | 1 | 2 | 4 | 3 |
| Оценка: | **4,63** | **3,00** | **2,75** | **2,87** | **3,25** | **2,50** | **3,00** | **4,88** | **4,38** |

# **ПРИЛОЖЕНИЕ 1**

1. **СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ**
2. ГОСТ 19.101–77 Виды программ и программных документов. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
3. ГОСТ 19.102–77 Стадии разработки. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
4. ГОСТ 19.103–77 Обозначения программ и программных документов. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
5. ГОСТ 19.104–78 Основные надписи. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
6. ГОСТ 19.105–78 Общие требования к программным документам. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
7. ГОСТ 19.106–78 Требования к программным документам, выполненным печатным способом. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
8. ГОСТ 19.404–79 Пояснительная записка. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
9. ГОСТ 19.603–78 Общие правила внесения изменений. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
10. ГОСТ 19.604–78 Правила внесения изменений в программные документы, выполненные печатным способом. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
11. ВКонтакте [Электронный ресурс]. Режим доступа: <https://vk.com/> (Дата обращения 01.01.2022)
12. The all-in-one campus experience management platform // Campus Group [Электронный ресурс]. Режим доступа: <https://www.campusgroups.com/> (Дата обращения 01.01.2022).
13. Skyrocket your society // Society App [Электронный ресурс]. Режим доступа: <https://www.createyoursociety.com/> (Дата обращения 01.01.2022).
14. LearningApps [Электронный ресурс]. Режим доступа: <https://learningapps.org/>

(Дата обращения 01.01.2022)

1. Online test pads [Электронный ресурс]. Режим доступа: <https://onlinetestpad.com/> (Дата обращения 01.01.2022)
2. ClassMarker [Электронный ресурс]. Режим доступа: <https://www.classmarker.com/online-testing/faq/> (Дата обращения 01.01.2022)
3. Quizziz [Электронный ресурс]. Режим доступа:<https://quizizz.com>(Дата обращения 01.01.2022)
4. Coursera [Электронный ресурс]. Режим доступа: <https://www.coursera.org/> (Дата обращения 02.02.2022)
5. Системные требования Java [Электронный ресурс] // URL <https://www.java.com/ru/download/help/sysreq.html> (Дата обращения: 01.02.2022, режим доступа: свободный).
6. API levels and Android versions [Электронный ресурс]// URL: <https://apilevels.com/> (Дата обращения 01.05.2022, режим доступа свободный)

# **ПРИЛОЖЕНИЕ** **2**

**ТЕРМИНОЛОГИЯ**

1. **Access token** – кратковременный токен для идентификации подлинности пользователя
2. **Activity** – является отдельным экраном в Android.
3. **Android SDK** – средство разработки моб. приложений.
4. **Base64** – метод кодирования информации.
5. **HMAC** (англ. *hash-based message authentication code* – «код аутентификации») – путь проверки информации на целостность
6. **Fragment** – является частью модуля Activity.
7. **Response** – ответ в виде текста JSON формата.
8. **JSON** (англ. *JavaScript Object Notation*) – текстовый формат обмена данными.
9. **JWT** (англ. *JSON Web Token*) – открытый стандарт (RFC 7519) чтобы инициализировать или создать токен, который представляет собой текст в формате JSON.
10. **Refresh** **Token** – долговременный токен, который нужен для получения новых access и refresh токенов.
11. **Авторизация** — это проверка прав пользователя на доступ к определенным ресурсам.
12. **База** **данных** – совокупность данных, которая хранится в соответствии со схемой данных и манипулирования которыми выполняются в соответствии с правилами средств моделирования данных

**ПРИЛОЖЕНИЕ 3**

**ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ КЛАССОВ**

Таблица 3.1

Описание и функциональное назначение классов мобильной части

| Класс | Назначение |
| --- | --- |
| ActivityUtitilities | Класс расширения для методов и полей |
| APIComet | Класс константы, который хранит ключ-значения сервера для общения между участниками |
| App | Класс, который предоставляет для инициализации авторизации |
| AppComponent | Класс, который предоставляет возможность инициализации между компонентами авторизации. |
| AssistantsAdapter | Класс адаптер для работы с инициализацией данных списка ассистентов |
| AvailableCoursesActivity | Класс Activity предоставляет возможность отображения данных о доступных курсах |
| ChatEntity | Класс модель чата для работы с чатом |
| ChatsFragment | Класс фрагмент для работы с отображением групповых чатов и списков |
| CompletedTestsAdapter | Класс адаптер для завершённых тестов для работы с инициализации данных списка тестов |
| ConstantEntities | Класс модель констант для инициализации данных |
| Constants | Класс констант |
| CourseEntity | Класс модель курса для работы с курсом |
| Courses | Класс модель курса для работы с курсом |
| CoursesAdapter | Класс адаптер курса для работы с инициализацией данных списка курсов |
| CreateGroupChatActivity | Класс Activity для группового чата для отображения создания чатов. |
| CreateNewsActivity | Класс Activity для работы с отображений создания новостей |
| CreateTestsActivity | Класс Activity для заданий |
| CurrentTestsAdapter | Класс адаптер для текущих тестов |
| ListOfAssistantsActivity | Класс Activity для отображений списка ассистентов |
| ListOfUsersActivity | Класс Activity для отображений списка участников |
| LoginAuthActivity | Класс Activity для авторизации через ЕЛК |
| MainActivity | Класс Activity для отображения меню страниц |
| MessageEntity | Класс модель сообщений |
| NewsAdapter | Класс адаптер для инициализации новостной ленты |
| NewsFragment | Класс Fragment для инициализации Layout компонент |
| NewsInfoActivity | Класс Activity для инициализации информации о новостях |
| PostEntity | Класс модель для событий |
| PreferenceManager | Класс Manager для контролирования данными |
| ProfileFragment | Класс Fragment для инициализации профиля пользователя |
| RequestEntity | Класс модель для заявок и дальнейшей работы с ним. |
| RequestsActivity | Класс Activity для инициализации списка заявок |
| RequestsAdapter | Класс адаптер для инициализации заявок курса |
| RoleOfUsers | Enum класс для обработки пользователей |
| SplashActivity | Класс Activity для отображения загрузочного экрана и инициализации предварительных данных |
| TaskEntity | Класс модель для выполнения/создания заданий |
| TaskInfoActivity | Класс Activity для отображения информаций заданий |
| TestsFragment | Класс Fragment для инициализации текущих и завершённых заданий |
| UserEntity | Класс модель для инициализации пользователей |
| UserTaskEntity | Класс модель для ответа заданий |

# **ПРИЛОЖЕНИЕ 4**

**ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ ПОЛЕЙ И ФУНКЦИЙ**

Таблица 4.1

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Методы | | | | | | | | | |
| Класс | ActivityUtilities.kt | | | | | | | |
| Имя | Модификатор доступа | Тип | | Аргументы | | Назначение | |
| Activity.toast | internal | Расширение | | message: CharSequence | | Метод для показа сообщения или подсказок | |
| showToast | public | Метод | | message: String | | Метод для показа сообщения в определенной странице | |
| String.asDateTime | public | Расширение | | Объект String | | Метод, который переводит дата и время в строку | |
| TextInputEditText.validateTitle | public | Расширение | | Объект TextInputEditText | | Метод для проверки ввода на корректность | |
| TextInputEditText.validateDescription | public | Расширение | | Объект TextInputEditText | | Метод для проверки ввода на корректность | |
| TextInputEditText.validateNews | public | Расширение | | Объект TextInputEditText | | Метод для проверки ввода на корректность | |
| Calendar.formatDateTime | internal | Расширение | | Объект Calendar | | Метод, который переводит дата и время в строку | |
| Свойства | | | | | | | |
| Имя | Модификатор доступа | Тип | | Доступ | | Назначение | |
| APP\_ACTIVITY | lateinit public | Activity | | get, set | | Назначение главного Activity | |
| toast | private | Toast | | get, set | | Объект для работы с Toast информациями. | |
| Свойства |  |  | |  | |  | |
| Класс | APIComet.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Аргументы | | Назначение | |
| APP\_ID | public const | String | | get | | ID Приложения | |
| API\_KEY | public const | String | | get | | ID API | |
| AUTH\_KEY | public const | String | | get | | ID Авторизации | |
| API\_COUNTRY | public const | String | | get | | ID Страны | |
| Методы |  |  | |  | |  | |
| Класс | App.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Аргументы | | Назначение | |
| cometInitialization | private | Метод | | - | | Метод для выставления настроек пользователя | |
| createNotificationChannel | private | Метод | | - | | Метод для инициализации уведомления | |
| provideAuthComponent | private | Метод | | - | | Метод для обработки авторизации компонент | |
| Свойства |  | | | | | | |
| Имя | Модификатор доступа | Тип | | Доступ | | Назначение | |
| TAG | private | String | | get | | Название класса | |
| Методы |  |  | |  | |  | |
| Класс | AppComponent.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Аргументы | | Назначение | |
| inject | public | Метод | | app: App или activity: LoginAuthActivity | | Закрепление объекта в компоненту | |
| loginComponent | public | Метод | | Отсутствуют | | Получить компоненту авторизации | |
| Свойства |  |  | |  | |  | |
| Имя | Модификатор доступа | Тип | | Доступ | | Назначение | |
| Builder | public | Интерфейс | | get | | Билдер инициализатор | |
| Методы |  | | | | | | |
| Класс | AssistantsAdapter.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Аргументы | | Назначение | |
| reloadItems | private | Метод | | - | | Инициализация ассистентов | |
| addItem | private | Метод | | item: User, position: Int | | Добавление пользователей в качестве ассистента | |
| removeItem | private | Метод | | position: Int | | Удаление пользователей со списка | |
| getAllItems | private | Метод | |  | | Получение всех ассистентов со списка | |
| Свойства |  |  | |  | |  | |
| Имя | Модификатор доступа | Тип | | Доступ | | Назначение | |
| context | private | Context | | get, set | | Контекст | |
| mItems | private | ArrayList<User> | | get, set | | Список ассистентов | |
| Методы |  |  | |  | |  | |
| Класс | AvailableCoursesActivity.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Аргументы | | Назначение | |
| updateItems | private | Метод | | list: ArrayList<Courses> | | Обновление данных курса | |
| Методы |  | | | | | | |
| Класс | ChatEntity.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Аргументы | | Назначение | |
| ChatEntity | public | Конструктор | | val id: Long = null,  var title: String? = null,  var description: String? = null, var courseID: Long? = null,var updatedBy: Long? = null, var group\_avatar: String? = null, var createdAt: Timestamp? =null, var updatedAt: Timestamp? = null | | Конструктор | |
| Свойства |  |  | |  | |  | |
| Имя | Модификатор доступа | Тип | | Доступ | | Назначение | |
| id | private | Long | | get, set | | ID чата | |
| title | private | String? | | get, set | | Заголовок чата | |
| description | private | String? | | get, set | | Описание чата | |
| courseId | private | Long? | | get, set | | ID курса | |
| updatedBy | private | Long? | | get, set | | Кем был обновлен чат | |
| group\_avatar | private | String? | | get, set | | Base64 изображение | |
| createdAt | private | LocalDateTime? | | get, set | | Дата и время создания | |
| updatedAt | private | LocalDateTime? | | get, set | | Дата и время обновления | |
| Методы |  | | | | | | |
| Класс | ChatsFragment.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Аргументы | | Назначение | |
| initializeListeners | private | Метод | | - | | Метод для инициализации и объявления чата | |
| loadFragment | private | Метод | | fragment: Fragment | | Метод для загрузки новой страницы-фрагмента | |
| startGroupIntent | private | Метод | | group: Group? | | Метод для запуска новой страницы с чатом | |
| joinGroup | private | Метод | | group: Group? | | Метод для принятия подтверждения в группу | |
| Свойства |  | | | | | | |
| Имя | Модификатор доступа | Тип | | Доступ | | Назначение | |
| progressDialog | private | ProgressDialog | | get, set | | Окно загрузки | |
| groupPassword | private | String? | | get, set | | Пароль вводимой группы | |
| group | private | Group? | | get, set | | Объект группы | |
| Методы |  |  | |  | |  | |
| Класс | CompletedTestAdapter.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Аргументы | | Назначение | |
| reloadItems | private | Метод | | - | | Инициализация ассистентов | |
| getAllItems | private | Метод | |  | | Получение всех ассистентов со списка | |
| Свойства | | | | | | | |
| Имя | Модификатор доступа | Тип | | Доступ | | Назначение | |
| completedTestList | private | ArrayList<TaskEntity> | | get, set | | Список заданий | |
| Свойства | | | | | | | |
| Класс | ConstantsEntities.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Доступ | | Назначение | |
| arrayOfRequestCourses | public | ArrayList<RequestEntity> | | get, set | | Список заявок | |
| allTests | public | ArrayList<TaskEntity> | | get, set | | Список из заданий | |
| USER | public | UserEntity | | get, set | | Класс модель пользователя | |
| postInfo | public | PostEntity | | get, set | | Класс модель новостного события | |
| taskInfo | public | TaskEntity | | get, set | | Класс модель задания | |
| Свойства | | | | | | | |
| Класс | Constants.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Доступ | | Назначение | |
| ACCESS\_TOKEN | public | String | | get, set | | Access токен для авторизации пользователя | |
| REFRESH\_TOKEN | public | String | | get, set | | Refresh токен для обновления авторизации пользователя | |
| USER\_CHAT | public | User | | get, set | | Объект пользователь для работы с чатом | |
| IP\_ADDRESS | public | String | | get, set | | Адрес сервера для работы с запросами | |
| REQUEST\_LOGIN | public | Long | | get | | Код для успешной авторизации | |
| TEMP\_TOKEN | public | String | | get | | Временный токен для показа примера | |
| SELECTION | public | CourseSelection | | get, set | | Enum тип для определения выбранного курса | |
| CourseSelection | public | Enum-класс | | get, set | | Enum-класс | |
| Свойства | | | | | | | |
| Класс | CourseEntity.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Доступ | | Назначение | |
| id | public | Long? | | get, set | | ID курса | |
| title | public | String? | | get, set | | Заголовок курса | |
| description | public | String? | | get, set | | Описание курса | |
| Свойства | | | | | | | |
| Класс | Courses.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Доступ | | Назначение | |
| nameOfCourse | private | String | | get, set | | Название курса | |
| image | private | String? | | get, set | | Base64 изображения курса | |
| Методы | | | | | | | |
| Класс | CoursesAdapter.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Аргументы | | Назначение | |
| CoursesAdapter | public | Конструктор | | courses: ArrayList<Courses> | | Список курсов | |
| updateData | public | Метод | | course: ArrayList<Courses> | | Обновление курса | |
| initializePreRequests | private | Метод | | position: Courses,  holder: View | | Инициализация заявок | |
| initializeButtons | private | Метод | | courseId: Int | | Инициализация полей | |
| Методы | | | | | | | |
| Класс | CreateNewsActivity.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Аргументы | | Назначение | |
| setListeners | public | Метод | | - | | Инициализация компонент Layout | |
| createEvent | private | Метод | | - | | Создание новостного события | |
| encodeUri | private | Метод | | imageUri: Uri | | Кодировать изображение в строку | |
| checkStoragePermissions | private | Метод | | - | | Проверить доступ к хранилищу | |
| requestStoragePermissions | private | Метод | | - | | Запросить доступ к хранилищу | |
| checkCameraPermissions | private | Метод | | - | | Проверить доступ к камере | |
| requestCameraPermissions | private | Метод | | - | | Запрос доступ к камере | |
| showImagePickerDialog | private | Метод | | - | | Показ диалогового окна для выбора изображения | |
| Методы | | | | | | | |
| Класс | CreateTestsActivity.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Аргументы | | Назначение | |
| setListeners | private | Метод | | - | | Инициализация Layout компонент | |
| createTest | private | Метод | | - | | Создание задания | |
| onTimeSet | private | Метод | | - | | Переопределение выбранного времени | |
| onDataSet | private | Метод | | - | | Переопределение выбранной даты | |
| Методы | | | | | | | |
| Класс | CurrentTestsAdapter.kt | | | | | | |
| Имя | Модификатор доступа | Тип | | Аргументы | | Назначение | |
| reloadItems | private | Метод | | - | | Инициализация текущих заданий | |
| getAllItems | public | Метод | | - | | Получение списка заданий | |
| Свойства | | | | | | | |
| Имя | Модификатор доступа | | Тип | | Доступ | | Назначение |
| completedTestsList | private | | ArrayList<TaskEntity> | | get, set | | Список из заданий |
| Методы |  | |  | |  | |  |
| Класс | ListOfAssistantsActivity.kt | | | | | | |
| Имя | Модификатор доступа | | Тип | | Аргументы | | Назначение |
| initializeCallBacks | private | | Метод | | - | | Инициализация логики добавления и удаления ассистентов |
| Методы |  | |  | |  | |  |
| Класс | ListOfUsersActivity.kt | | | | | | |
| Имя | Модификатор доступа | | Тип | | Аргументы | | Назначение |
| loadFragment | private | | Метод | | fragment: Fragment? | | Открытие и переход на выбранный фрагмент |
| Методы |  | |  | |  | |  |
| Класс | LoginAuthActivity.kt | | | | | | |
| Имя | Модификатор доступа | | Тип | | Аргументы | | Назначение |
| authLoginCredentials | private | | Метод | | - | | Получение или регистрация пользователя |
| createUser | private | | Метод | | - | | Создание пользователя для чата |
| login | private | | Метод | | - | | Авторизация в приложение |
| Методы |  | |  | |  | |  |
| Класс | MainActivity.kt | | | | | | |
| Имя | Модификатор доступа | | Тип | | Аргументы | | Назначение |
| setupSmoothBar | private | | Метод | | - | | Инициализация нижнего меню |
| Методы |  | |  | |  | |  |
| Класс | NewsAdapter.kt | |  | |  | |  |
| Имя | Модификатор доступа | | Тип | | Аргументы | | Назначение |
| reloadItems | private | | Метод | | - | | Обновление и инициализация новостного события |
| getItemCount | public | | Метод | | - | | Получить количество объектов в списке |
| getAllItems | public | | Метод | | - | | Получение списка из новостного события |
| Свойства |  | |  | |  | |  |
| Имя | Модификатор доступа | | Тип | | Доступ | | Назначение |
| news | private | | ArrayList<PostEntity> | | get, set | | Работа для с новостными событиями |
| Методы |  | |  | |  | |  |
| Класс | NewsFragment.kt | |  | |  | |  |
| Имя | Модификатор доступа | | Тип | | Аргументы | | Назначение |
| initializeNewsAdapter | private | | Метод | | - | | Инициализация новостной ленты |
| initializeGradesAdapter | private | | Метод | | - | | Инициализация оценок за тесты |
| Методы |  | |  | |  | |  |
| Класс | NewsInfoActivity.kt | | | | | | |
| Имя | Модификатор доступа | | Тип | | Аргументы | | Назначение |
| initializeTopAppBar | private | | Метод | | - | | Обновление и инициализация верхнего меню |
| initializeFields | private | | Метод | | - | | Инициализация полей |
| editDataNews | private | | Метод | | - | | Запрос на изменение данных новостей |
| validateInputFields | private | | Метод | | - | | Метод для проверки данных |
| Свойства |  | |  | |  | |  |
| Класс | PostEntity.kt | | | | | | |
| Имя | Модификатор доступа | | Тип | | Доступ | | Назначение |
| id | private | | Long? | | get, set | | ID новостного события |
| courseId | private | | Long? | | get, set | | ID курса |
| createdBy | private | | Long? | | get, set | | Кем был создан |
| updatedBy | private | | Long? | | get, set | | Кем был обновлен |
| title | private | | String? | | get, set | | Заголовок новостного события |
| content | private | | String? | | get, set | | Контент задания новостного события |
| mediaLink | private | | String? | | get, set | | Base64 изображение |
| docLink | private | | String? | | get, set | | Ссылка на документ |
| createdAt | private | | String? | | get, set | | Дата и время создания |
| updatedAt | private | | String? | | get, set | | Дата и время изменения |
| Методы |  | |  | |  | |  |
| Класс | ProfileFragment.kt | | | | | | |
| Имя | Модификатор доступа | | Тип | | Аргументы | | Назначение |
| initializeDefaultFeature | public | | Метод | | - | | Инициализация для пользователя ролью “Студент” и “Ассистент” |
| initializeTeacherFeatures | public | | Метод | | - | | Инициализация для пользователя с ролью “Преподаватель” |
| Свойства |  | |  | |  | |  |
| Класс | RequestEntity.kt | | | | | | |
| Имя | Модификатор доступа | | Тип | | Доступ | | Назначение |
| id | private | | Long | | get, set | | ID заявки |
| userId | private | | Long | | get, set | | ID пользователя |
| courseId | private | | Long | | get, set | | ID курса |
| roleId | private | | Long | | get, set | | Роль пользователя |
| approved | private | | Boolean | | get, set | | Принята ли заявки или нет |
| createdAt | private | | LocalDateTime | | get, set | | Дата и время создания |
| updatedAt | private | | LocalDateTime | | get, set | | Дата и время изменения |
| Методы |  | |  | |  | |  |
| Класс | RequestEntity.kt | | | | | | |
| Имя | Модификатор доступа | | Тип | | Аргументы | | Назначение |
| getAllItems | public | | Метод | | - | | Получение всех объектов со списка |
| addItem | public | | Метод | | item: RequestEntity, position: Int | | Добавление объекта в список |
| removeItem | public | | Метод | | position: Int | | Удаление объекта со списка |
| Свойства |  | |  | |  | |  |
| Имя | Модификатор доступа | | Тип | | Доступ | | Назначение |
| news | private | | ArrayList<PostEntity> | | get, set | | Работа со списком объектов |
| Свойства |  | |  | |  | |  |
| Класс | RoleOfUsers.kt | | | | | | |
| Имя | Модификатор доступа | | Тип | | Доступ | | Назначение |
| STUDENT |  | | int | | get | | Роль пользователя |
| ASSISTANT |  | | int | | get | | Роль пользователя |
| TEACHER |  | | int | | get | | Роль пользователя |
| Методы |  | |  | |  | |  |
| Класс | SplashActivity.kt | | | | | | |
| Имя | Модификатор доступа | | Тип | | Аргументы | | Назначение |
| isLoggedIn | private | | Метод | | - | | Проверка на авторизацию |
| Свойства |  | |  | |  | |  |
| Класс | TaskEntity.kt | |  | |  | |  |
| Имя | Модификатор доступа | | Тип | | Доступ | | Назначение |
| id | private | | Long? | | get, set | | ID задания |
| courseId | private | | Long? | | get, set | | ID курса |
| form | private | | String? | | get, set | | Тип задания |
| title | private | | String? | | get, set | | Заголовок задания |
| description | private | | String? | | get, set | | Описание задания |
| task\_content | private | | String? | | get, set | | Информация о задании |
| deadline | private | | String? | | get, set | | Дедлайн задания |
| status | private | | Boolean? | | Get, set | | Статус задания “Завершённое” или “Текущее” |
| createdBy | private | | Long? | | get, set | | Кем был создан |
| updatedBy | private | | Long? | | get, set | | Кем был изменен |
| createdAt | private | | String? | | get, set | | Дата и время создания |
| updatedAt | private | | String? | | get, set | | Дата и время изменения |
| Методы |  | |  | |  | |  |
| Класс | TaskInfoActivity.kt | | | | | | |
| Имя | Модификатор доступа | | Тип | | Аргументы | | Назначение |
| setListeners | private | | Метод | | - | | Инициализация компонент Layout |
| validateInputFields | private | | Метод | | - | | Проверка на корректность данных |
| initializeTopAppBar | private | | Метод | | - | | Инициализация верхнего меню |
| fieldOptionsInitializer | private | | Метод | | - | | Инициализация свойств |
| Методы |  | |  | |  | |  |
| Класс | TestsFragment.kt | |  | |  | |  |
| Имя | Модификатор доступа | | Тип | | Аргументы | | Назначение |
| initializeCompletedTests | private | | Метод | | - | | Инициализация завершённых тестов |
| initializeCurrentTests | private | | Метод | | - | | Инициализация текущих тестов |
| Свойства |  | |  | |  | |  |
| Класс | UserEntity.kt | |  | |  | |  |
| Имя | Модификатор доступа | | Тип | | Доступ | | Назначение |
| id | private | | Long? | | get, set | | ID пользователя |
| fullName | private | | String? | | get, set | | ФИО пользователя |
| email | private | | String? | | get, set | | Почта пользователя |
| createdAt | private | | LocalDateTime? | | get, set | | Дата и время создания аккаунта пользователя |
| Свойства |  | |  | |  | |  |
| Класс | UserTaskEntity.kt | |  | |  | |  |
| Имя | Модификатор доступа | | Тип | | Доступ | | Назначение |
| id | private | | Long? | | get, set | | ID пользователя |
| answer | private | | String? | | get, set | | Ответ пользователя |
| score | private | | Int? | | get, set | | Оценка пользователя |
| status | private | | Boolean? | | get, set | | Статус задания “Завершённое” и “Текущее” |
| taskId | private | | Long? | | get, set | | ID задания |
| createdBy | private | | Long? | | get, set | | Кем был создано задание |
| updatedBy | private | | Long? | | get, set | | Кем было обновлено задание |
| createdAt | private | | LocalDateTime? | | get, set | | Дата и время создания задания |
| updatedAt | private | | LocalDateTime? | | get, set | | Дата и время изменения задания |

**ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | |
| Номера листов (страниц) | | | | | Всего листов (страниц в докум.) | № документа | Входящий № сопроводительного докум. и дата | Подп. | Дата |
| Изм. | Измененных | Замененных | Новых | Аннулированных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |