

**Конкурсное задание**

**по компетенции**

**«Разработка мобильных приложений»**

Сессия 1

Необходимо разработать мобильное приложение ДЛЯ СМАРТФОНА, удовлетворяющее следующим требованиям:

Приложение должно поддерживать следующие версии ОС:

* Android 9.0 и новее
* iOS 13.0 и новее

В работе необходимо использовать систему контроля версий Git. Ссылка на Git: Для входа используйте учетную запись вида wsruserX, где X – это номер участника. Необходимо загрузить каждую сессию в отдельную ветку с именем “Session-X”, где Х – это номер сессии. Для каждого проекта необходим отдельный репозиторий.

Необходимо корректно обрабатывать запросы к серверу. В случае получения ошибки от сервера или отсутствия соединения с сетью Интернет необходимо отобразить соответствующий текст ошибки с помощью диалогового окна.

Необходимо строго следовать предложенному дизайну. Макеты приложения доступны по ссылке: https://www.figma.com/file/xGHdXPgFC8LNQUpiBON2H5/WorldCinema-Session1-%D0%9C%D0%B5%D0%B6%D0%B2%D1%83%D0%B7?node-id=0%3A1

Во время работы не будет доступа в Интернет, кроме документации и API. Описание протокола API доступно по ссылке:

https://app.swaggerhub.com/apis-docs/WorldSkills-MAD/WorldCinema/1.0.0

Проект приложения должен быть структурирован по экранам, то есть исходные файлы конкретного экрана должны быть в соответствующей папке. Общие для нескольких экранов классы необходимо поместить в папку common.

Необходимо реализовать следующий функционал:

1. Создайте проект. Настройте иконку приложения согласно макету. Следует учесть разницу в отображении иконок на различных версиях операционной системы.
2. Реализуйте экран Launch Screen согласно макету. Текст должен быть отдельным элементом. Логотип приложения должен быть расположен по центру экрана.
3. Реализуйте экран SignUp Screen согласно макету:
   * При нажатии на кнопку "Зарегистрироваться" необходимо проверять поля для ввода на пустоту, а также email на корректность (требования к email описаны в документации к API). При некорректном заполнении необходимо отобразить ошибку с помощью диалогового окна. Так же необходимо проверять равенство пароля и его повтора.
   * При корректном заполнении формы необходимо отправлять запрос регистрации на сервер. При получении ошибки от сервера ее необходимо отобразить с помощью диалогового окна. При успешной регистрации нужно автоматически осуществить авторизацию и перейти на Main Screen.
   * При нажатии на кнопку "У меня уже есть аккаунт" необходимо осуществлять переход на SignIn Screen.
   * При первом запуске приложения после Launch Screen должен отображаться SignUp Screen. При последующих - SignIn Screen.
4. Реализуйте экран SignIn Screen согласно макету:
   * При нажатии на кнопку "Войти" необходимо проверять поля для ввода на пустоту, а также email на корректность (требования к email описаны в документации к API). При некорректном заполнении необходимо отобразить ошибку с помощью диалогового окна. При корректном заполнении формы необходимо отправить на сервер соответствующий запрос.
   * При нажатии на кнопку "Регистрация" необходимо осуществлять переход на SignUp Screen.
   * При успешной авторизации необходимо осуществлять переход на экран Main Screen. При получении ошибки от сервера необходимо отобразить ее с помощью диалогового окна.
5. Реализуйте экран Main Screen согласно макету:
   * Содержимое экрана должно скроллиться по вертикали.
   * Реализуйте рекламный блок в верхней части экрана согласно макету. Для получения информации о фильме-обложке используйте соответствующий GET-запрос.
   * Изображение для обложки необходимо загружать с сервера.
   * При нажатии на кнопку "Смотреть" необходимо осуществлять переход на экран соответствующего фильма.
   * Реализуйте отображение подборок фильмов согласно макету. Подборки должны скроллиться по горизонтали. Для получения списка фильмов используйте соответствующий GET-запрос.
   * При нажатии на конкретный фильм должен осуществляться переход на Movie Screen для соответствующего фильма.
   * Реализуйте отображение последнего просмотренного видео. Для получения информации необходимо использовать запрос /usermovies с параметром filter = ”lastView”. На экране необходимо отобразить только последний просмотренный фильм.
   * При нажатии на изображение последнего просмотренного фильма должен осуществляться переход на экран соответствующего эпизода.
   * В нижней части экрана необходимо реализовать управляющий компонент согласно макету.
   * Реализуйте цветовую дифференциацию активной и неактивных вкладок согласно макету.
6. Реализуйте экран Movie Screen согласно макету:
   * Содержимое должно скроллиться по вертикали. Информацию о фильме необходимо получать с сервера.
   * Цвет возрастного ограничения должен зависит от рейтинга фильма согласно макету.
   * Список кадров должен скроллиться по горизонтали.
   * Теги должны отображаться один за другим до конца строки. При достижении конца строки следующий тег должен быть перенесен на следующую строку.
   * Реализуйте отображение списка эпизодов согласно макету. Данные необходимо запрашивать с сервера.
   * Реализуйте блок отображения видео согласно макету. При нажатии на кнопку “Запуск” необходимо начинать воспроизведение видео. В качестве видео-фрагмента используйте preview первого эпизода. При его отсутствии необходимо скрыть данный блок.