

**Конкурсное задание**

**по компетенции**

**«Разработка мобильных приложений»**

**Сессия 3**

Все проекты проверяются с Git репозиториев. Адрес http://62.113.111.138/. Для входа используйте учетную запись, которая была вам предоставлена. Необходимо загрузить каждую сессию в отдельную ветку с именем “Session-X”, где Х – это номер сессии.

Необходимо строго следовать предложенному дизайну. Макеты приложения доступны по ссылке: https://www.figma.com/file/91FXFW2tZz6bB8NZ5ifGk6/VOS-Session-3

Вы будете работать с REST API, доступ к которому у Вас будет на протяжении всех сессий.

Документация к API: https://app.swaggerhub.com/apis-docs/serk87/APIfood/1.0.0

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.

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 для часов:

1. Настройте иконку приложения согласно макету.
2. Реализуйте экран SplashScreen согласно макету.
3. Реализуйте экран SignInScreen согласно макету:
   * При нажатии на кнопку «Sign In» необходимо проверять поля для ввода на пустоту, а также email на корректность (email должен соответствовать паттерну "name@domenname.ru", где имя и домен второго уровня могут состоять только из маленьких букв и цифр, домен верхнего уровня - только из маленьких букв. Длина домена верхнего уровня - не более 3х символов). При некорректном заполнении необходимо отобразить ошибку с помощью диалогового окна. При корректном заполнении формы необходимо получить токен с помощью соответствующего запроса REST API.

При успешной авторизации необходимо осуществлять переход на экран OrderScreen. При получении ошибки от сервера необходимо отобразить ее с помощью диалогового окна.

1. Реализуйте экран OrderScreen согласно макету:
   * Необходимо отобразить текущий заказ курьера, данные берутся с сервера. При получении ошибки от сервера необходимо отобразить ее с помощью диалогового окна.
   * Реализован возврат на предыдущий экран любым способом. Token обнуляется.
   * При нажатии на адрес должен открываться экран MapScreen.
   * Список блюд скролится по вертикали, при необходимости.
2. Реализуйте экран MapScreen согласно макету:
   * Проложите маршрут до адреса указанного на экране OrderScreen, от текущей геолокации курьера.
3. Реализуйте один Unit и один UI тест на экране , опишите их сценарий работы в readme.