Вариант № 1

Задание для демонстрационного экзамена по комплекту оценочной документации

|  |  |
| --- | --- |
| **Номер компетенции** | F6 |
| **Название компетенции** | Разработка мобильных приложений |
| **Номер КОД** | 1.1 |

# Описание задания.

**Описание системы**

Мобильные устройства стали неотъемлемоӗ частью жизни современного человека. Многие люди практически не выпускают смартфон из рук, используя его во всех сферах жизни.

Сегодня умный дом упрощает управление любыми бытовыми приборами. Управление позволяет с помощью мобильного устройства создавать сценарии, диммировать и контролировать безопасность дома или дачи в дали от них.

Данная система может управлять отоплением, вентиляцией, включением света, а также выполнять функции сигнализации о несанкционированном проникновении в дом или утечке воды на пол. Все эти функции реализуются при помощи специальных датчиков, подключенных к серверу. Сервер обрабатывает сигналы от датчиков и отправляет команды устройствам исполнителям.

В рамках экзамена необходимо разработать мобильное Smart House.

**Задание**

Задание состоит из 3 модулей, в рамках которых необходимо разработать приложения для трех целевых устройств: смартфон, smart-часы, а также представить результаты своей работы на презентации.

Перед началом участник получит описание функционала для реализации, дизайн-макет, реализованный с помощью AdobeXD и описание протокола API в онлайн-сервисе Swagger.io.

**Результат**

Конечным результатом является набор мобильных приложений для целевого набора устройств.

# Описание модуля 1 Смартфоны:

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: Для входа используйте учетную запись вида userX, пароль wsruserX, где X – это номер участника.

**По завершении сессии необходимо сохранить модуль в ветке “Session-X”, где Х – это номер модуля.**

Необходимо корректно обрабатывать запросы к серверу. В случае получения ошибки от сервера или отсутствия соединения с сетью Интернет необходимо отобразить соответствующий текст ошибки в диалоговом окне, которое должно закрываться только пользователем.

Необходимо строго следовать предложенному дизайну. Макеты приложения доступны в файлах Adobe XD**.**

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

**https://app.swaggerhub.com/apis-docs/k5422/smart/1.0.0-oas3#/**

Проект приложения должен быть структурирован в соответствии с выбранным паттерном проектирования приложения, то есть исходные файлы должны быть в соответствующей папке, общие классы необходимо поместить в папку **common**.

**Например, при использовании в проекте MVC, файлы проекта должны быть распределены по следующим папкам: view, model, controller, 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. Зарегистрируйте Ваше приложение используя API
3. При запуске приложения на телефоне оно должно быть зарегистрировано с использованием UUID телефона. Далее Вам будет выдан ключ, который будет актуален, только в течении 5 минут.
4. Реализуйте экран Launch Screen согласно макету. Логотип приложения должен быть расположен по центру экрана. Реализуйте анимацию как на макете (поворот логотипа по часовой стрелке)
5. Реализуйте экран SignIn Screen согласно макету:
   * При нажатии на кнопку "Enter Your House" необходимо проверять поля для ввода на пустоту, а также email на корректность (соответствие паттерну "name@domenname.ru", где имя и доменное имя может состоять только из маленьких букв и цифр). При некорректном заполнении необходимо отобразить ошибку. При корректном заполнении формы необходимо отправить на сервер соответствующий запрос.
   * При нажатии на кнопку "New Resident" необходимо осуществлять переход на SignUp Screen.
   * Только при успешной авторизации необходимо осуществлять переход на экран Main Screen. При получении ошибки от сервера необходимо отобразить ее.
6. Реализуйте экран SignUp Screen согласно макету:
   * При нажатии на кнопку "Зарегистрироваться" необходимо проверять поля для ввода на пустоту, а также email на корректность (соответствие паттерну "name@domenname.ru", где имя и доменное имя может состоять только из маленьких букв и цифр). При некорректном заполнении необходимо отобразить ошибку.
   * При корректном заполнении формы необходимо отправлять запрос регистрации на сервер. При получении ошибки от сервера ее необходимо отобразить. Только при успешной регистрации нужно перейти на экран SignIn Screen.
   * При нажатии на кнопку "Enter Your House" необходимо осуществлять переход на SignIn Screen.
7. Реализуйте экран Main Screen согласно макету:
   * Реализуйте отображение комнат согласно макету. Комнаты должны скроллиться по вертикали. Для получения списка комнат используйте соответствующий GET-запрос. Иконки комнат должны быть отображены в зависимости от типа комнаты.
   * Реализуйте нижнее меню. При активном экране соответствующая кнопка должна быть выделена цветом, как на макете. В этой сессии Вы реализуете только экран **Home**. На остальные экраны вставьте текст “Under reconstruction”.
   * Реализуйте добавление комнат. Должен осуществляться переход на экран “Add room”
8. Реализуйте экран **Add room** согласно макету:
   * Реализуйте возможность ввести название комнаты. Если пользователь начал вводить с маленькой буквы, Вам необходимо преобразовать первое слово с заглавной буквы.
   * Реализуйте возможность выбрать тип комнаты. Если список не помещается на экран должен скролится по вертикали, при этом поле ввода названия должно оставаться на месте.
   * При нажатии на кнопку Save должен отправляться на сервер запрос. Если ответ был успешный необходимо автоматически перейти на главный экран. В случае ошибки, необходимо оставаясь на текущем экране ее отобразить.
9. Main Screen:
   * При нажатии на “Your Home” должна открываться карта
   * Реализуйте возможность выбрать на карте точку и отобразите адрес под надписью “Your Home”, как на макете
   * Отправьте на сервер координаты дома
   * При нажатии на комнату должен осуществляться переход на экран с устройствами, конкретной комнаты
10. Device Screen
    * Реализуйте пустой экране как на макете
    * При нажатии на кнопку “+” должно отобразится диалоговое окно с возможность выбрать из двух вариантов устройств “Light”, “Thermostat”. После выбора реализуйте запрос к серверу с соответствующими параметрами.
    * Все добавленные в комнату устройства должны отображать, как на макете. Элементы должны скролится по горизонтали.
    * Реализуйте возможность выключения и выключения. На термостате необходимо использовать стандартный элемент, используя цветовую схему макета.
    * Light:
      + - Реализуйте возможность прибавления и убавления яркости света используя жесты swipe up и swipe down
        - Данные должны отправляться на сервер по завершению движения пальца и его подъема
        - Thermostat:

Реализуйте возможность изменения температуры, как на макете, при помощи жеста.

Реализуйте возможность изменения скорости вентилятора, как на макете, при помощи жеста.

В зависимости от выбора “Heating” или “Cool” отображение температуры будет положительное или отрицательное

Данные должны отправляться на сервер по завершению движения пальца и его подъема на всех управляющих элементах.

1. Settings Screen:
   * Реализуйте отображение иконки, и данных пользователя.
   * Реализуйте выход из приложения. Данные должны отправляться на сервер и  только при успешном ответе пользователь переходит на экран **Enter Your House**
   * Реализуйте редактирования профиля:
     + - При нажатии на соответствующее поле должна быть возможность редактирования
       - При нажатии на кнопку камеры, должна быть возможность выбрать фото из галереи
       - При нажатии на “Save” все данные и фото отправляются на сервер, а также в случае успеха перед на экран **Settings**

# Описание модуля 2 Умные часы:

Необходимо разработать мобильное приложение ДЛЯ ЧАСОВ.

Приложение должно поддерживать следующие версии ОС:

* Android 9.0 и новее
* WatchOS 7.0 и новее

В работе необходимо использовать систему контроля версий Git: Для входа используйте учетную запись вида userX, пароль wsruserX, где X – это номер участника.

**По завершении сессии необходимо сохранить модуль в ветке “Session-X”, где Х – это номер модуля.**

Необходимо корректно обрабатывать запросы к серверу. В случае получения ошибки от сервера или отсутствия соединения с сетью Интернет необходимо отобразить соответствующий текст ошибки в диалоговом окне, которое должно закрываться только пользователем.

Необходимо строго следовать предложенному дизайну. Макеты приложения доступны в файлах Adobe XD**.**

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

**https://app.swaggerhub.com/apis-docs/k5422/smart/eqfwgrhtjyfkuydwqewrtehsjyrdkutlfog**

Проект приложения должен быть структурирован в соответствии с выбранным паттерном проектирования приложения, то есть исходные файлы должны быть в соответствующей папке, общие классы необходимо поместить в папку **common**.

**Например, при использовании в проекте MVC, файлы проекта должны быть распределены по следующим папкам: view, model, controller, common.**

Необходимо реализовать следующий функционал:

1. Создайте проект приложения для smart-часов. Настройте иконку приложения согласно макету. Следует учесть разницу в отображении иконок на различных версиях операционной системы.
2. Зарегистрируйте Ваше приложение используя API
3. При запуске приложения на smart-часах оно должно быть зарегистрировано с использованием UUID часов. Далее Вам будет выдан ключ, который будет актуален только в течении 5 минут.
4. Реализуйте экран Launch Screen согласно макету. Реализуйте анимацию как на макете (уменьшение в 0 px )
5. Реализуйте экран SignIn Screen согласно макету:

* При нажатии на кнопку "Enter Your House" необходимо проверять поля для ввода на пустоту, а также email на корректность (соответствие паттерну "name@domenname.ru", где имя и доменное имя может состоять только из маленьких букв и цифр). При некорректном заполнении необходимо отобразить ошибку. При корректном заполнении формы необходимо отправить на сервер соответствующий запрос.
* При успешной авторизации необходимо осуществлять переход на экран Main Screen. При получении ошибки от сервера необходимо отобразить ее.

1. Реализуйте экран Main Screen согласно макету:

* Реализуйте отображение комнат с сервера
* При выборе комнаты должен открываться экран с устройствами

1. Реализуйте экран Device Screen согласно макету:

- Реализуйте экран отображения устройств комнат

- Реализуйте экран **Light**:

- При не выбрано плюс или минус кнопка выключения при длинном нажатии должна выключать свет,  данные должны отправляться на сервер.

- При выбраном плюсе или минусе нажатие на кнопку выключения должна прибавлять или убавлять яркость света,  данные должны отправляться на сервер при каждом нажатии.

- Не реализовывайте экран **Thermostat**

# Описание модуля 3 Защита проекта:

Подготовьте презентацию с отчетом о разработанных в рамках Экзамена приложениях. Сравните требования к приложениям и итоговый результат.

Целевая аудитория выступления - менеджеры проекта. Продолжительность выступления - 5 минут.

# Необходимые приложения

Файл с макетом Adobe XD: Smart\_House\_Module\_1. zip

Файл с макетом Adobe XD: Smart\_House\_Module\_2. zip