2 Сессия

**Общее**

Необходимо разработать мобильное приложение для смартфона, удовлетворяющее следующим требованиям:

Минимальная версия ОС, поддерживаемая приложением, должна быть: Android - 13.0, iOS - 14.0.

**В работе необходимо использовать систему контроля версий Git, которую предоставляет организатор.**

Все проекты проверяются с Git репозиториев. Необходимо загрузить каждую сессию в отдельную ветку с именем «Session\_Х», где Х – это номер сессии. По завершению каждой сессии необходимо делать средствами GitHub Merge слияние с основной веткой, которая должна называться «Main», при этом ветка удаляться не должна.

В качестве бэкенда будет использован Supabase. Авторизация в нем будет происходить через выданный аккаунт GitHub.

**Необходимо строго следовать предложенному дизайну.**

**Вся верстка должна быть адаптивной (следует учитывать разные размеры экранов). Необходимо:**

* **Избегать появления большого пустого пространства;**
* **Следить за отсутствием искажения элементов;**
* **Все элементы должны полностью находится в границах и на месте, указанном в макете;**
* **Учитывать расстояние между элементами;**
* **Используйте шрифты согласно макету.**

**Дизайн предложен в Figma:**

<https://www.figma.com/file/MluNlGZBfxE45DUUlJEEdx/OECH-APP?type=design&node-id=14-7747&mode=design&t=mEbo48F3pMKRk1S0-0>

Необходимо осуществлять комментирование кода в созданных классах. Обязательны следующие комментарии:

* Описание назначения класса
* Дата создания
* Автор создания
* Описание назначения вложенных элементов программного кода

При разработке проекта приложения вам необходимо использовать архитектуру ([см.файл с описанием архитектуры](https://docs.google.com/document/d/1DHGJzhbW0XN6D1P-USjFZkrPoljXKINbpMohblCme8I/edit?usp=sharing)), в которой будут разделены слои бизнес-логики, представлений и домена. Изменение бизнес-логики и/или представления одного из экранов не должно повлечь за собой изменение других экранов и нарушение работоспособности приложения, за исключением переходов. Допускается использование Supabase.

Файлы проекта распределены по папкам в соответствии с архитектурой. Допустимо использование папки Common для общих файлов.

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. В процессе обмена данными с сервером должна осуществляться стандартная индикация.

Все медиа ресурсы должны кэшироваться.

**Сессия 2**

Во время сессии предстоит разработать админ-панель и профиль пользователя. Админ-панель предназначена для обновления статуса доставок и техподдержки пользователей. Данные для авторизации админа:

Log In admin - admin@mail.ru

Password admin - admin123

1. Создайте экран «Forgot password», как на макете:

* При нажатии на кнопку «Send OTP» при наличии в поле ввода корректного e-mail, осуществляется переход на экран «OTP Verification».
* Реализуйте возможность перехода на экран «Log in» при нажатии на «Sign in».
* Реализуйте отправку запроса на сервер для получения кода.

2. Создайте экран «OTP Verification», как на макете:

* Реализуйте возможность повторного запроса кода по истечению таймера 00:10.
* Если код-пароль введен верно, то все квадраты окрашиваются в синий цвет, если код-пароль введен не верно, то все квадраты окрашиваются в красный цвет.
* При корректном коде при нажатии на кнопку «Set New Password» осуществляется переход на экран «New Password».
* Реализуйте отправку кода на сервер для верификации.
* Реализуйте повторную отправку кода на сервер для верификации.

3. Создайте экран «New Password», как на макете:

* При нажатии на кнопку «Log In», если пароли совпадают осуществляется изменение пароля и переход на экран «Home».
* Реализуйте отправку запроса на сервер для изменения пароля.

4. Создайте экран «Admin», как на макете:

* Реализуйте возможность выхода из панели админа, возвращение к экрану «Log in».
* При нажатии на кнопку «Orders» осуществить переход на экран «Admin Orders».
* При нажатии на блок меню, появляется выделение, соответствующее макету.

5. Создайте экран «Admin Orders», как на макете:

* Реализована возможность возврата на экран «Admin» и при нажатии на заказ реализован переход на экран «Admin Order Info». Выведите с сервера пролистываемый список активных заказов.
* Если статус не минимальный, при нажатии на кнопку «Reduce Status» статус возвращается к предыдущему этапу и отправляется запрос на изменение статуса на сервер.
* Если статус не максимальный, при нажатии на кнопку «Increase Status» статус переходит к следующему этапу и отправляется запрос на изменение статуса на сервер.

6. Создайте экран «Admin Order Info», как на макете:

* Реализуйте возможность вернуться на экран «Admin Orders».
* Данные о заказе берутся с сервера.