Тестовое задание

Общая информация

* Названия экранов в техническом задании соответствуют названию экранов в Figma
* Все шрифты, цвета и иконки можно брать из Figma.
* Для проверки задания нужно прислать видео работающего приложения, всех его экранов, а также ссылку на GIT
* На выполнение ТЗ дается 1 неделя с момента получения этого файла
* В гите не указывайте название компании, чтобы другие участники не смогли скопировать вашу работу
* Желаем вам успехов!

Описание экранов

**ВАЖНО:** при открытии первым открывается экран Главная

По данной[**ссылке**](https://www.figma.com/file/fSVhgQTluvoqkAa6ZnpQQO/%D0%A2%D0%B5%D1%81%D1%82%D0%BE%D0%B2%D0%BE%D0%B5-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5_ios?type=design&node-id=0%3A1&mode=design&t=i8rS3cLLnEqkBIgg-1) можно перейти на макет в Figma. Ссылка на API дана в файле после изображений экранов.

Экран 1 – Главная

Изображение выглядит как текст, еда, снимок экрана

Автоматически созданное описание

[**API**](https://run.mocky.io/v3/058729bd-1402-4578-88de-265481fd7d54) **для категорий**

Список элементов:

- Город, где находится пользователь, дата и время.

- Фотография пользователя – Должна быть просто фотография пользователя (добавить), кнопка не кликабельна

- Блоки с обозначением кухни - являются кнопками, при нажатии на которые происходит переход на экран Категория. При нажатии на каждую категорию блюд открывается список одних и тех же блюд, что и в экране Категория.

- Нижняя навигационная панель, по которому происходит переход в разделы приложения.

Экран 2 – Категория

Изображение выглядит как текст, овощи, меню, снимок экрана

Автоматически созданное описание

[**API**](https://run.mocky.io/v3/aba7ecaa-0a70-453b-b62d-0e326c859b3b) **для блюд**

Список элементов:

- Нижняя навигационная панель – должна оставаться там при переходе в любые разделы приложения.

- Кнопка Назад – чтобы выйти на главную, со списком категорий блюд.

- Название категории – обязательно должно быть.

- Тэги – сортируют список товаров, по указанным в них тэгам. У каждого блюда есть тэг, который ему соответствует, при нажатии остаются только те блюда, имеющие соответствующий тег в описании.

- Фотография пользователя – всегда занимает верхний правый угол.

- Список блюд. По три блюда в ряд по горизонтали. Название блюд равняются по левому краю. При нажатии происходит открытие экрана Продукт (всплывающее окно).

Экран 3 – Продукт

Изображение выглядит как текст, Мобильный телефон, снимок экрана, еда

Автоматически созданное описание

Список элементов:

Вся карточка один сплошной элемент.

- Фон – задний фон должен быть в той позиции, на которой находился пользователь в момент выбора блюда.

- Фотография блюда и название – та же что и в списке блюд.

- Цена блюда и вес – стоимость за одну порцию и вес порции.

- Описание блюда – краткое описание блюда.

- В избранное – При нажатии на кнопку ничего не происходит.

- Закрыть – кнопка, чтобы закрыть карточку с ценой и описанием блюда.

- Добавить в корзину – блюдо добавляется в корзину покупателя.

Экран 4 -Корзина

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мобильный телефон

Автоматически созданное описание

Список элементов:

- Верхняя панель – та же, что и на Главной, с указанием адреса, даты и времени.

- Список блюд в корзине – указывается фотография, название, стоимость, и вес блюда.

- Количество штук в корзине – можно отсюда же увеличить или уменьшить количество блюд в корзине.

- При нажатии на “-” должно происходить уменьшение количества блюд. Если нажать минус когда блюда всего 1 штука в корзине, то оно исчезает оттуда.

- Кнопка Оплатить – указывается стоимость всей корзины. При изменении количества блюд, итоговая стоимость в кнопке меняется.

Сценарий – Заказ блюда

Актер – пользователь.

Главный успешный сценарий:

1. Пользователь видит список категорий блюд.
2. Пользователь нажимает на понравившуюся категорию и переходит в список блюд в этой категории.
3. Если пользователь нашел нужное блюда, то нажимает на нее и видит ее описание и цену.
4. Пользователь добавляет блюдо в корзину.
5. При желании пользователь может увеличить количество блюда в корзине.
6. Пользователь оплачивает покупку.

Альтернативный сценарий:

3. Если пользователь нашел нужное блюда, то нажимает на нее и видит ее описание и цену.

3а. Если пользователю не понравилось описание, стоимость, размер порции, он закрывает карточку и ищет дальше.

     3б.  Пользователь при желании может отфильтровать список блюд по указанным тэгам.

Сценарий - Удаление блюда из корзины

Актер - пользователь

Предусловие - Пользователь n количество блюд в заказ

Успешный сценарий:

1. Пользователь видит в корзине количество блюд, которые он добавил
2. Пользователь хочет уменьшить количество блюд  в корзине.
3. При нажатии на “-” у него уменьшается на одно
4. Если блюда в корзине 0, то блюдо исчезает из корзины.

      Альтернативный сценарий:

     2. Пользователь хочет уменьшить количество блюд  в корзине.

          2.а Пользователь при желании увеличивает количество блюда прямо из корзины, нажимая на “+”.

Рекомендуемый стек технологий:

Android:

* Kotlin
* Корутины (но если на RX сможете, тоже отлично)
* Flow или LiveData или RX
* Dagger или Koin
* MVVM
* AdapterDelegates
* Верстка обычная на XML
* Clean Architecture
* Многомодульность (обязательно)

iOS:

* Swift
* MVVM + Coordinator
* SwiftUI
* Combine

Flutter:

* Dart
* Block (Или любой стейт менеджер)
* Clean Architecture
* Retrofit

Критерии оценки

1. Количество сделанных требований
2. Насколько визуал соответствует Figma
3. Насколько чисто и расширяемо написан код, и соответствует SOLID
4. Какое архитектурное решение реализовано
5. Умение пользоваться основными библиотеками
6. Наличие многомогомодульности (только у Android)