**Техническое задание к мобильному приложению “Food”.**

# Цель приложения.

Целью “Food” является предоставление пользователям интуитивно понятного приложения для заказа доставки еды.

# Задачи приложения.

Среди задач приложения можно выделить:

* Обеспечить удобный выбор блюд и ресторанов.
* Обеспечить оформление и оплату заказов в несколько шагов.
* Обеспечить работу системы рейтингов, отзывов и рекомендаций.
* Обеспечить административный интерфейс для ресторанов.

# Функциональные требования.

Требования при разработке серверной части:

1. Реализованная база данных должна покрывать все задачи приложения;
2. Все данные, которые используется в системе, должны храниться в базе данных, за исключением локальных и данных для тестирования;
3. Для управления доступом к данным должна быть реализована политика безопасности;
4. В базе данных должно быть реализовано хранение изменяемых файлов и изображений;
5. Аутентификация пользователя должна быть реализована по почте и паролю;
6. Восстановление пароля пользователя по почте должно быть реализовано с помощью шестизначного OTP-кода;
7. База данных должна быть заполнена достаточным количеством данных для проверки работоспособности всех функций приложения;

Минимальные требования к функционалу приложения:

1. аутентификация: неавторизованные пользователи должны быть ограничены в доступе к данным;
2. регистрация пользователя: для создания нового пользователя необходимо ввести: имя пользователя, электронную почту и пароль. Данные пользователя отправляются на сервер для регистрации;
3. авторизация пользователя: для авторизации пользователя необходимо ввести: электронную почту и пароль. Данные пользователя отправляются на сервер для авторизации.
4. пользовательские сессии:

* после успешной аутентификации данные пользователя сохраняется в системе для дальнейшего использования;
* электронная почта и пароль сохраняются в локальное хранилище, они используются для повторного входа и удаляются при выходе из системы;
* данные токена пользователя сохранятся в виде локальной переменной и очищаются при закрытии приложения.

1. валидация вводимых данных:

* проверять поля на соответствие типу данных;
* проверять поля для ввода на пустоту, на длину (короткий текст и числа: не более 20 символов, пароль: не более 8 символов);
* email на корректность (соответствие паттерну "name@domenname.ru", где имя и доменное имя может состоять только из маленьких букв и цифр).
* проверять повторный ввод пароля на соответствие исходному;
* проверять даты на корректность;

1. навигация в приложении:

* реализовать навигацию между всеми необходимыми экранами приложения
* для реализации навигации рекомендуется использовать фрагменты или библиотеки навигации;
* для перехода между основными экранами приложения (например, главный экран, профиль, корзина) рекомендуется использовать нижнюю навигационную панель

1. ПИН-код:

* после аутентификации пользователю открывается экран ПИН-кода, на котором он может ввести свой ПИН-код;
* если пользователь успешно авторизован в системе, при повторном входе в приложение открывается Экран входа по ПИН-коду, в котором пользователь может ввести ПИН-код либо выйти из приложения для повторной аутентификации.

1. восстановление пароля:

* пользователь может перейти к функции восстановления пароля либо с экрана входа, либо с экрана профиля;
* пользователь вводит электронную почту, на которую приходит шестизначный OTP-код;
* после отправки кода на почту приложение переводит пользователя на экран Ввода OTP-кода, где пользователь должен ввести корректный OTP-код;
* если OTP-код корректен, пользователь переходит на экран Изменения пароля, где вводит и подтверждает новый пароль.

1. локальное хранилище:

* необходимо хранить данные пользователя, необходимые для автоматической аутентификации, в локальном хранилище приложения.

1. профиль пользователя:

* личные данные пользователя сохраняются в его профиле и отображаются на экране Личного профиля;
* на экране Личного профиля пользователь может изменить данные о пользователе: имя пользователя, электронную почту, пароль, аватар и т.д.;
* пользователь может загрузить аватар из галереи устройства.

1. отображение множества объектов данных пользователю:

* данные должны отображаться в удобном для пользователя формате в виде горизонтального и вертикального списка;
* обновление списка объектов по жесту;
* детализация элементов списка:
  + переход на подробное описание элемента по нажатию на элемент;
  + вызов контекстного меню при долгом нажатии на элемент;
  + действия при нажатии на детали на элементе списка (например, добавление в избранное при нажатии на кнопку на элементе списка)
* экран Получение подробностей об одном объекте.

1. фильтрация данных:

* поиск по названию и/или описанию объекта данных;
* фильтрация по нескольким свойствам объекта данных (например, категориям, кол-ву товара, просмотрам и т.д.) отдельно или одновременно.

1. изменение данных о объекте данных:

* создание, редактирование и удаление объектов данных;
* система должна корректно аутентифицировать пользователя и позволять управлять только разрешенными объектами;
* изменение данных должно быть реализовано в удобном для пользователя виде.

1. работа со сложными объектами:

* отображение, фильтрацию, создание, удаление, редактирование сложных объектов (имеющих две и более связи с другими таблицами) или сложных для восприятия (неструктурированных, или связующих таблиц) в удобном человеко-читаемом формате;

1. работа с изображениями:

* все изменяемые медиа должны храниться, создаваться и добавляться из сервиса Supabase.

1. обработка ошибок:

* необходимо корректно обрабатывать запросы к базе данных;
* 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.

1. выход:

* выход из пользовательской сессии;

# Дизайн-макет в Figma.

Требования при реализации дизайна приложения:

1. дизайн приложения должен строго следовать разработанному макету;
2. вся верстка должна быть адаптивной (следует учитывать разные размеры экранов);
3. приложение должно иметь настроенную иконку в соответствии с макетом;
4. приложение должно иметь следующие экраны:

* загрузочный экран;
* экран Приветствия (Onboarding), который отображается, если пользователь не авторизован;
* экран Входа;
* экран Авторизации;
* экран Ввода ПИН-кода;
* экран Входа по ПИН-коду;
* экран Регистрации;
* экран Восстановление пароля;
* экран Ввода OTP-кода;
* экран Изменения пароля;
* экран Личного профиля;
* экран Получение множества объектов данных;
* экран Получения множества данных с фильтрацией;
* экран Получение подробностей об одном объекте;
* экраны Создание нового объекта, Изменения объекта (допускается объединение);
* экран Создание сложных объектов (например, корзина, группа пользователей)
* диалоговые окна для ошибок.