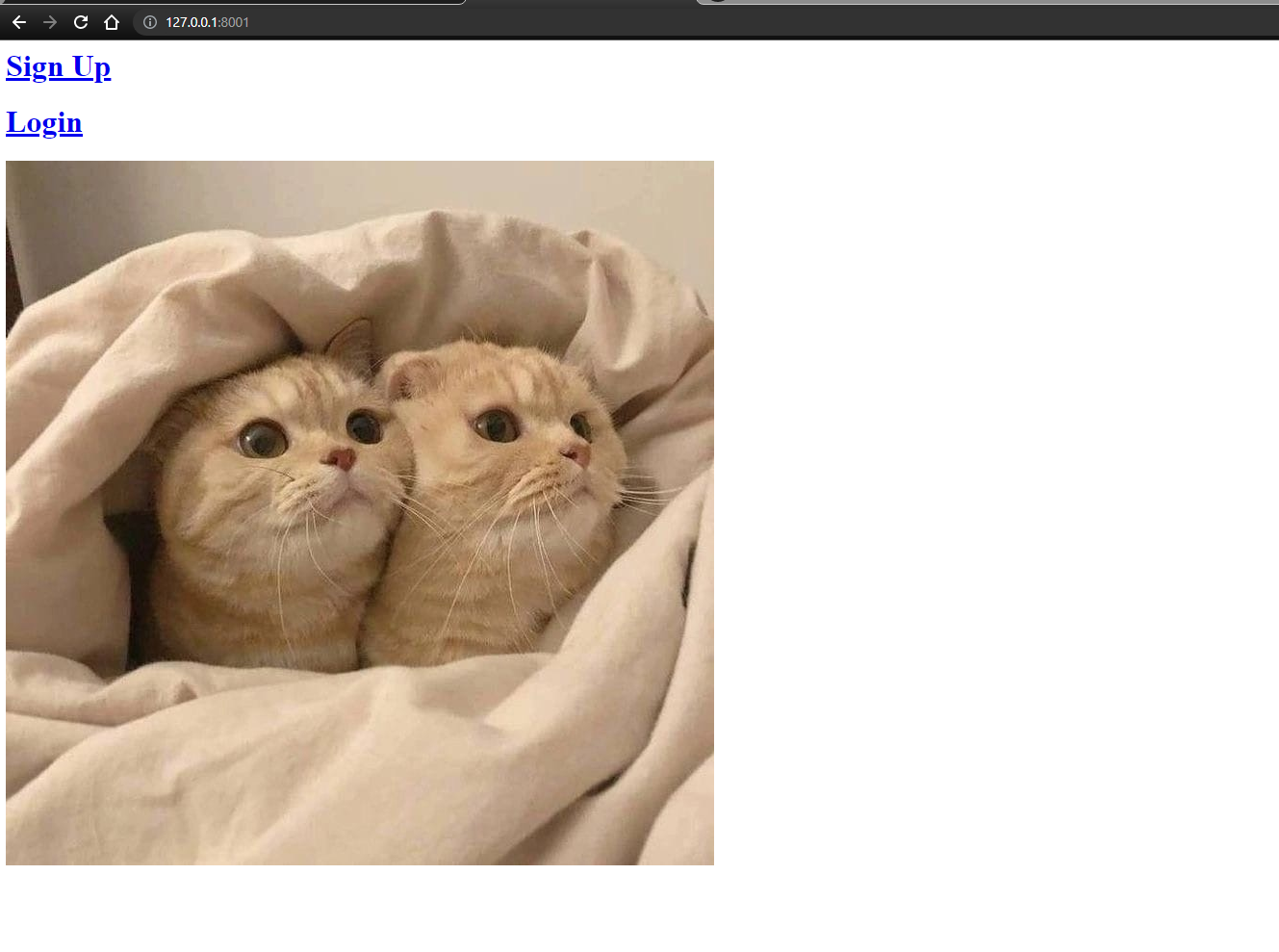
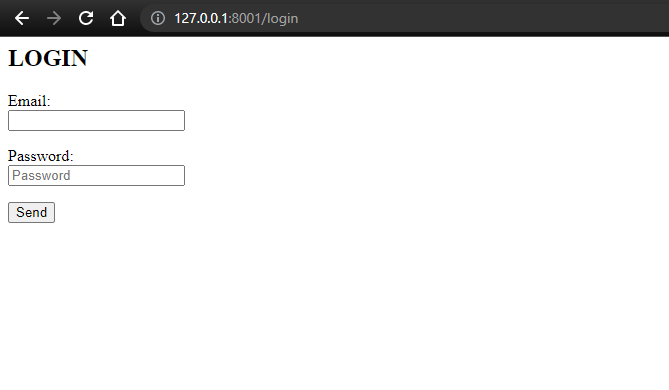
При входе в приложение должен быть какой-то фон и две кнопки: login, signup. Пример на вебке:



Рисунок

При нажатии на логин нужно ввести почту и пароль. Пример на вебке:



Рисунок

При нажатии на signup нужно ввести почту, пароль, подтверждение пароля. Пример на вебке:

Изображение выглядит как текст

Автоматически созданное описание

Рисунок

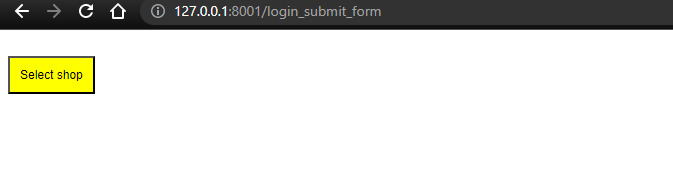
При нажатии на кнопку send (у тебя наверное будет кнопка далее), и на странице логина, и на странице сигнапа, дальше перекидывает на страницу с вводом кода подтверждения (тип двухфакторка). Пример на вебке:

Изображение выглядит как текст

Автоматически созданное описание

Рисунок

После ввода кода (он шестизначный, если это имеет значение. Состоит только из цифр) происходит переход на страницу с выбором магазинов. У нас это кнопка при наведении на которую раскрываются магазы (какие пока хз. Просто добавь туда «магазин 1», магазин 2 и тд). Пример на вебке до наведения и после:



Рисунок

Изображение выглядит как текст

Автоматически созданное описание

Рисунок

После выбора магазина тебе приходит json (пример example.json) в котором есть колонки приведенные на скрине ниже. Надо его распарсить и отрисовать. Пример на вебке:

Изображение выглядит как текст, в помещении, снимок экрана

Автоматически созданное описание

Рисунок

Далее еще не продумано. Предположительно на вебке в правом верхнем углу этого скрина будут формы ввода количества дней, количества денег на эти дни, черный список продуктов (либо нет, не помню уже, обещали мы его или нет). И потом будет происходиь магия и будет составляться рацион. Рацион по заданным ценникам и дням будет приходить тебе в том же формате, что и example.json, так что отрисовывать надо будет точно так же.

Теперь по поводу куков: после перехода на страницу ввода кода подтверждения (Рисунок 4) тебе прилетит строка, называемая JWT-токен. Его мы вставили в куки в поле value. Если для мобилки есть какая-то хрень, которая выставляет этот токен, то супер. Если нет, то искать аналог куков. Теперь по взаимодействию:

Рисунок 1 – нажатие любой кнопки на этом рисунке не приводит к взаимодейтсвию мобилки с нашим сервером.

Рисунок 2 – отправляешь json вида {email: email, password: password} по айпишнику. В ответ приходит что-то типа OK, NEOK. Скорее всего это будет сделано через http запросы, так что заранее делай сенд и ресив в виде http. Вместе с OK тебе приходит JWT-token.

Рисунок 3 – то же самое.

Рисунок 4 – отправляешь json вида {code: code}. В ответ ловишь OK/NEOK (пускать/не пускать к выбору магазина).

Рисунок 5-6 – отправляешь json вида {shop: shop}. Скорее всего в значении будет цифра, об идентификаторах магазинов потом договоримся. В ответ ловишь json вида example.json.

Рисунок 7 – Отрисовываешь полученный example.json. Делаешь заглушку на отправку json вида {days: days\_number, money: money, …}. Отправляешь. В ответ ловишь json так же вида example.json. Отрисовываешь.

Заметки:

1. Практически на каждом переходе ты отправляешь серверу запрос. Будь готов отлавливать коды респонсов. Какой-то из них у нас будет означать, что jwt-токен истек. При получении этого кода вставить заглушку на повторный вход на сайт (страница логина, далее подтверждение кода).
2. Предусмотреть кнопки назад, НО окно с подтверждением кода не может быть достижимо при нажатии кнопки назад. Все остальные окна должны быть доступны. При этом, если авторизованный пользователь с непросроченным токеном, захотел зайти на страницу логина или signup (рисунки 2,3) он не должен попадать на эти страницы. Должен кидаться на страницу выбора магазинов (рисунок 5) .