# Структура проекта

Прежде чем я начну рассказывать о самом приложении скажу пару слов о структуре проекта. И на чем собственно он написан

Апи - .NET Core Web API

3 основных раздела:

1) Базы данных, бизнес логика, контроллеры. Раздел базы данных это сущности, которые описывают таблицы, репозитории для каждой сущности, Unit of work для объединения репозиториев и более удобной работы с ними.

2) Бизнес логика это объекты ДТО и сервисы, которые преобразовывают данные с базы данных и передают в контроллеры уже данные в нужном нам виде.

3) Ну и контроллеры это точки доступа от клиента к нашему апи.

Ангулар - 3 основных раздела:

1) папка апи - работа с апи – это интерфейсы и сервисы для работы с апи

2) папка страницы – все страницы приложения, каждая со своим модулем, со своим роутинг модулем (со своим маршрутом), со своим основым компонентом и с вложенными компонентами, которые используются только на этой странице

3) и третий раздел – это общие компоненты, которые могут использоваться на разных страницах.

Все элементы максимально вынесенны в отдельные компоненты, которые максимально переиспользуются.

# Вступление

Теперь перейдем к самому приложению.

Из всех существующих приложений мое, наверно, больше всего напоминает инстаграмм, но есть некоторые отличия. Это приложение не для фоток себя любимого, там где ты сидишь в дорогом ресторане, или за рулем дорогой машины (не всегда своей), что бы показать окружающим насколько ты крут, и что бы хорошо выглядять в глазах других людей. Это приложение не будет показывать твою статусность - оно будет показывать твои интересы. ["Покажи мне не себя, а покажи мне свои интересы и я скажу кто ты" - первая заставка на слайде]

То есть фотографиями мы показываем свои интересы – это могут быть снимки природы во время каких-то путешествий, снимки каких-то краевидов, возможно животных, или предметов искусства. Может это снимок картины, которую ты написал или вышивки, которую ты вышил. То есть пользователи рассказывают о своих хобби фотографиями и обсуждают их в комментариях.

# Главная страница

1. Возможность просматривать бесконечную ленту постов

Как только мы заходим в приложение мы сразу видим главную страницу с лентой постов. Пост состоит из имени пользователя, фотографии, дальше у нас кол-во лайков, дизлайков, описание, кол-во комментариев к посту и дата. Так же здесь мы видим возможность зарегистрироваться или войти в систему.

След посл главной страницы:

Подробнее о бесконечном скролле: Здесть мы видим ту же главную страницу, но я уменьшил ширину постов, что бы нагляднее. Как только мы зашли на страницу загрузилось только 2 поста (это можно изменить, но по умолчанию размер страницы для пагинации указан как 2). Но как только мы начинаем скролить вниз – мы видим в панеле разработчиков, что появляется сразу же еще 2 поста следующих. Важно заметить, что мы только начали скролить а два следующих поста уже выгрузились и есть внизу, мы просто их не видим. Но в этом и вся фишка, пока мы скролим, мы не успеваем никогда доскролить до конца, потому что времени пока мы дойдем до конца поста уже вполне хватить что бы выгрузить следующие два поста. К тому же посты отсортированны по дате что бы вверху всегда были последние.

# Логин

2. Аутентификация на основе JWT

Для аутентификации мы будем использовать JWT токен, то есть для доступа к любой странице кроме главной мы будем в запросе слать токен, который выдается пользователю после успешного ввода логина и пароля. Этот токем мы храним в local storage.

# Регистрация

3. Загрузка файлов на сервер, валидация

Ну с регистрацией все понятно, тут все как обычно, но здесь можно немного рассказать о загрузке файлов на сервер. После того как мы нажали на кнопку «Загрузить файл» открывается окно выбора файла и там нам предлагают выбрать файл с разрешением .jpg, .png, или .jpeg. (и видны только файлы с таким разрешением) и если мы послушаем и выберем правильно – все будет хорошо, иначе мы увиди ошибку. Но если же мы все сделам правильно, то файл загрузится на сервер и от туда вернется новый путь этого файла в папке уже на сервере и вместе с остальными данными пользователя мы это все отправим в запросе на регистрацию.

Валидация реализована как на стороне клиента так и на стороне сервера, в случае если ее отключат на клиенте.

# 4. Профиль пользователя

После того, как мы удачно залогинились мы можем зайти в свой профиль. Здесь мы видим информацию о пользователе и ниже будут сразу выгруженны посты этого пользователя.

для следующих 3 пунктов я не делал отдельные слайды, здесь ничего особо интересного нет

- избранные посты пользователя это посты которые пользователь лайкнул – они отображаются точто так же как и обычные посты пользователя, с пагинацией и все такое.

- возможность изменить данные своего профиля (отредактировать информацию о себе). Обычная форма бля редактирования

- и выложить новый пост – тоже форма для добавления нового пользователя

Информация о пользователе – она сделана в виде карточки отдельным компонентом (потому что испотльзовать эту карточку мы будем не только на странице профиля пользователя). Ну и вобще практически все здесь отдельные компоненты и многие из них используются в разных местах.

# 5. Who is alpha here??

В приложении реализована система рейтинга и ее логика очень простая – твой рейтинг это количество лайков на всех твоих постах минус количество дизлайков на всех твоих постах. Это и заставит пользователей выкладывать адекватный контент и создает какой-то дух соревнования что будет подстегивать пользователя побольше проводить времени в приложении.

# 6. Поиск

6. Найти пользователя или пост?

С помощью одной формы поиска мы можем искать как пользователей по их юзернейму, так и росты по их описаниям. То есть если описание содержит в себе ключевое слово то вывести все такие посты и показать с пагинацией. Так же можно вводить и в аппер кейсе, там сравнение идет в ловере все. Для пользователей я не делал отдельный слайд – если введенно верное имя пользователя то мы увидим карточку пользователя с информацией и списой его постов (опять же с пагинацией)

# 7. Куда можно кликать?

При наведении на имя пользователя вся верхняя полоска статет оранжавой и при нажатии мы перейдем на страницу пользователя мы увидим карточку пользователя с информацией и списой его постов (точно такая же страница как при поиска пользователя – этот переход это такой же собственно поиск, только имя пользователя приходит не с текстового поля, а с роут параметра адресной строки).

Для лайков и дизлайков сужествуют проверки, при которых нельзя ставить лайк по своему посту или если ты незарегистрированный пользователь, а так же нельзя ставить лайк повторно.

Ну и при нажарии на комментарии мы переходим на странику с комментариями

# Комментарии

Здесь мы видим пост, комментарии к нему и текстовое поле в которое можем вписать текст и отправить его как новый комментарий к посту, комментарии можем так же лайкать и так же переходить на страницу пользователя при клике на имя пользователя.

Ответы к комментариям:

Если ответы к коментариям есть – мы видим количество и определенную стилизацию, которая показывает нам что туда можно кликать.

**Ответы на комментарии**

При клике мы видим все ответы и кнопка по которой мы кликнули так же служит кнопкой скрытия ответов обратно . так же мы видим кто кому ответил.

# Еще ответов мне побольше

Для комментариев и ответов на комментарии так же задействована логика погинации. Если комментариев будет больше 5 (настраивоемое значение) – видим кнопу «Показать следующие»

При нажатии на кнопку мы увидим следующие 5 (или сколько будет) ответов.

# Комментируем комментарии

Мы так же можем отвечать на комментарии после нажания на кнопку «ответить»

Прямо под комментарием или ответом на комментарий появляется текстовое поле в котором мы видим, кому мы отвечаем (текстовое поле – отдельный компонент). Это очень удобно, поскольку не нужно никуда скролить или переходить.