The solution in terms of content and concept:

My main aim was to learn Angular combined with NGRX. NGRX — is a state management tool that gives more power with Angular to maintain the state for an application.

For the Front-end technologies decision was taken easy according to my aim.

As for Backend technologies, I found an article about a blogging platform working in the style of SPA, on the base of the existing backend, in which man cannot make changes.

It was very fascinating that the same Medium.com clone (called Conduit) could be built using different frontends and backends. And man can mix and match them because they all adhere to the same API. This means may be in the future I could expand this project with other technologies.

I have decided to make a blogging platform for students.

RealWorld project provides a ready-made backend with rest API supporting all necessary functionality.

RealWorld provides also a custom theme for Bootstrap 4.

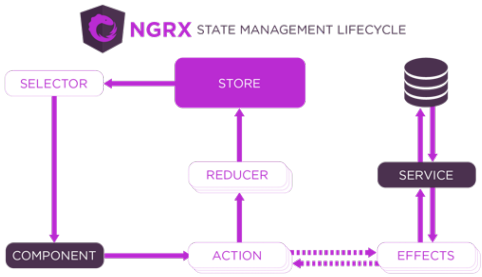
I took advantage of this but made the necessary changes to the styles.scss file according to my wishes

Here is an example of an expected final project:

http://angular.realworld.io/

It was helpful to see how my final implemented features should like, as well as it was helpful to compare my result for debugging.

As Angular and Ngrx look quite complicated for the beginner:



According to the given diagram, I have decided to use the following structure:

Each page has its module (in one directory I have collected component, action, reducer, store, selector effect, and service), then in the root file named app.module.ts, I added all these modules.

Besides that, I have shared modules, that can be reused on multiple pages. For instance, Backend errors can appear on the registration page, log-in page, and creating article page.

One more approach was interesting for me to try – an optimistic update. I used it the favoriting the articled and to follow the users. The advantage is that the view changes immediately, and the user gets a fast response. It would not be a good approach to use it for online payment, but for my tasks, I found it useful.

This was implemented differently. First, I have created a component without service and store (as I did with other modules) and implemented this feature without API.

It worked but after refreshing the page – the result disappeared.

Then I combined this feature with the backend creating service, action types, action, and effects. I didn't create reducers or selectors as I didn't save any data, but only dispatch action to the backend.

Learned lessons:

1) I have learned Angular.

- understanding of how to create Angular applications
- fundamentals about modules, components
- outing and handling navigation
- how to use TypeScript to write Angular applications

2) I have learned NGRX - a framework for building reactive applications in Angular. And I have understood the NGRX state management life cycle in practice

3) I have learned Optimistic updates in Angular applications.

4) I have learned to use Store Devtools - Instrumentation ngrx that enables visual tracking of state and time-travel debugging.

5) Comparing the previous phases, I have improved my knowledge of how to provide an architecture diagram.

In conclusion, I can sum up that I have achieved my goals: a blogging platform works with all implemented features scheduled in the previous phases, as well as I learned aimed tool to implement the blogging platform.

BIBLIOGRAPHY

Tour of Heroes application and tutorial, (n.d). https://angular.io/tutorial

NgRx Team, (n.d). https://ngrx.io

Siminons, E. (2017): Introducing RealWorld (URL: https://medium.com/@ericsimons/introducing-realworld-6016654d36b5)

Kapoor, N. (2018): Real Time Example of NGRX + Angular (URL: https://medium.com/@nikhilkapoor17/real-time-example-of-ngrx-angular-2a298098b034).

Kapoor, N. (2018): 3 W's with NGRX (URL: https://medium.com/@nikhilkapoor17/3-ws-with-ngrx-14a0de27357).

Pranskunas, V. (2019): Optimistic updates in Angular applications (URL: https://medium.com/@vpranskunas/optimistic-updates-in-angular-applications-a78ae2adc746).