

## General information

- You are to report the time spent on the project, so remember to keep a log of working hours.
- The project is to be carried out in groups of 4-6 students. No exceptions.
- Looking for a group, or the group is too small? Write a post at piazza.
- There are no scheduled tutor meetings. Please use piazza, or make an appointment with `per.andersson@cs.lth.se` if you get stuck.
- Divide the work into smaller parts and let the individual group members take responsibility of the different parts. Work in small increments and use continuous integration. I encourage pair programming and continuous discussions among the group members. Having responsibility of a component does not mean you have to do all the work. Help your group members, and they help you.

## Project description

The project is mandatory for the web programming course. The goal is for you to get hands on experience with a selected part of the course that has not been covered in the labs.

Mandatory parts for all projects:

- You are to develop a single page web application.
- The application should be slightly larger than the salad bar app from the labs. Probably 4-6 pages depending on their complexity.
- Use the angular framework, <https://angular.io>.
- Styling must be done using an angular package, for example angular material <https://material.angular.io>, or ng-bootstrap <https://ng-bootstrap.github.io/>. This is different compared to the labs, where css classes were used to style the app. In the project, you are to get hands on experience with using angular components and directives for styling.

In addition to the requirements above, you are to select one topic from the course content to build your application from. Here are some suggestions. Pick one, do not be too ambitious. It is better to do one well, than handing in a half working solution of an ambitious project you did not have time to finish.

- Use redux for state management. The standard package for this is `@ngrx/store`. Be aware, you will need to write a fair amount of boiler plate code to set up the actions and reducers.
- Http requests and merging data. You write an app that gather related data from different sources and present them to the user. Use the angular http object for fetching data. There are plenty of open data sources, for example: <https://opnnadata.se>, <http://www.omdbapi.com>, <https://openlibrary.org/dev/docs/api/books>,
- Explore reactive programming using RxJS. Angular is built upon RxJS, for example reactive forms, `EventEmitters` (passing data to parent components) and the http interface. The application should use both RxJS operators, as well as higher order observables.
- Your own suggestion.

## Timeframe and deliverables

- 21/2 — hand-in of a project proposal. The hand-in is done here: <https://sam.cs.lth.se/portal/>. When you hand-in the proposal, you also form the project group, remember 4-6 students in each group. Describe the functionality of your app and what packages you plan to use. The proposal can be half an A4 page, and definitely less than 2 A4 pages. I will give you feedback on the coverage of your proposal (to little/much work, to simple/complex task).
- 13/3 — deadline for final report. Hand-in is done on the same web page as the proposal. I believe in running code. I prefer codeSandbox, but a git repo also works. Hint, if you use github.com, you can create a codeSandbox that is synced with your repo, see <https://codesandbox.io/docs/importing#import-from-github>. In addition to running code, you are to hand-in a written report, see bellow for details.

## The final report

The report should cover the following topics:

- Success ratio and lessons learned. Describe how much of your initial idea was actually implemented. You do not need to be detailed about what you got running, I see this from the running code. Rather, the focus should be on the parts that did not work out as intended.
- Main obstacles during the project. This can be anything, from group dynamics to getting a piece of code to run. (primarily so I can adjust the topics for coming years)
- Individual statement of contributions. Here I expect that several group members have been involved in most parts of the project. Please also report the amount of time spent on the project. The reported time can be anonymous, i.e. student 1, student 2..., and will only be used to adjust the topics for coming years.
- The report is probably 1-5 A4 pages.

*Editor:* Per Andersson

*Contributors* in alphabetical order:

Alfred Åkesson

Oscar Ammkjaer

Per Andersson

*Home:* <https://cs.lth.se/edaf90>

*Repo:* <https://github.com/lunduniversity/webprog>

This compendium is on-going work.

**Contributions are welcome!**

*Contact:* [per.andersson@cs.lth.se](mailto:per.andersson@cs.lth.se)

You can use this work if you respect this *LICENCE*: CC BY-SA 4.0

<http://creativecommons.org/licenses/by-sa/4.0/>

Please do *not* distribute your solutions to lab assignments and projects.

Copyright © 2015-2020.

Dept. of Computer Science, LTH, Lund University. Lund. Sweden.