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 students. Enroll for the groups at `https://sam.cs.lth.se/LabsSelectSession?occasionId=743`. You can enroll on your own, the system will form groups automatically. If two or more students enroll at the same time, you will end up in the same group (the system will not split groups, but will add more students to existing groups)

- 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.

- Use the angular framework, `https://angular.io`.

- The application should be slightly larger than the salad bar app from the labs. At least one angular reactive or template form and 4-6 components depending on their complexity.

- 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.

- Fetch data using http requests and store the data in an angular service. You write an app that gather data from one or several external 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://www.dataportal.se`, `http://www.omdbapi.com`, `https://openlibrary.org/dev/docs/api/books`, `https://www.openstreetmap.org`

- The data must be passed from your service to the components using streams (rxjs observable).

Timeframe and deliverables

- 25/2 — hand-in of a project proposal. The hand-in is done in canvas. Only one in the group needs to hand in the proposal. 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).

- 11/3 — deadline for final report. Hand-in is done in canvas. 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/git`. 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)

- A short reflection on which knowledges and skills you acquired in other courses that was most useful during your project.

- 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`