

Deliverable report

COMPLEAP

Date:30.5.2019

Work package:WP3

Title: D24 Open source code for all components, including example code

Author:EDUFI

Description of the action:

EDUFI has subcontracted developers for the creation of the HTML prototype as well as the Beta prototypes for the ComLeap project. As such, two separate repositories have been created housing the open source code for each of the modules in the beta prototype as well as the full html prototype.

Outcome of the action:

The EDUFI Github repository contains source code for ComLeap Prototype components.

The HTML-prototype is available at <http://comleap-proto.testiopintopolku.fi>. The source code and documentation for the HTML-prototype is available at: <https://github.com/Opetushallitus/comleap-proto>

The Beta prototype is available at <https://poc.comleap.testiopintopolku.fi/>. All the open source code for the PoC is available here: <https://github.com/Opetushallitus/comleap>

CompLeap Beta Prototype Technologies and Development Tools

The CompLeap beta prototype, proof of concept is available at

<https://poc.compleap.testiopintopolku.fi/>. The source code and technical documentation for the HTML-prototype is available at: <https://github.com/Opetushallitus/compleap>

This source code acts as an example implementation of the prototyped CompLeap service. The prototype application demonstrates a learner profile that makes use of information about learner's skills, competences, and interests. With the aid of this information, learning opportunity recommendations are offered to help the learner realise new areas where the competences and interests could be combined.

In terms of source code, the prototype application is divided into two main components:

- Client-side application (model directory)
- Recommendations application (src directory)

Technologies and Development Tools

- JavaScript with ECMAScript 2015 features (programming language for the client app)
- R (programming language for the recommendations app)

The client app also makes use of the following core technologies and development tools:

- Node.js & NPM (<https://nodejs.org/en/>) (dependency management)
- React (<https://reactjs.org/>) (views)
- webpack (<https://webpack.js.org/>) and Babel (<https://babeljs.io/>) (build)
- styled-components (<https://www.styled-components.com/>) (styles)
- JavaScript Standard Style (<https://standardjs.com/>) and ESLint (<https://eslint.org/>) (code style)

In addition, following tools are used to ease the development:

- Docker (<https://www.docker.com/>) and Docker Compose (<https://docs.docker.com/compose/>) (development environment)

Installing dependencies

Ensure that you are using a correct version of Node.js and NPM:

```
node --version
# v10.15.3
```

```
npm --version
# 6.4.1
```

Install dependencies:

```
npm i
```

Create `.env` file for environment variables by copying it from the example file:

```
cp .env.example .env
```

Running client app locally

```
npm run start:dev
```

This starts up the development server (webpack-dev-server) and serves the app at <http://localhost:8080/>.

Alternatively, you can create a development build (`npm run build:dev`) and serve the files from the `dist` directory with some other HTTP server application. The production build can be created similarly by running `npm run build:prod`.

For using the client with the recommendations API, see [below](#).

Running recommendations API locally

To start up the recommendations API locally, run:

```
docker-compose build model-api
docker-compose up model-api
```

The API is now available at `localhost:8000`.

Connecting from the client app

First, switch from mock API to the locally served API by changing the `API_ENDPOINT` environment variable from `mock` to `/v2/match`.

To use the local recommendations API together with the client app (being served from e.g. the webpack-dev-server), both the API and an Nginx reverse proxy can be started by running:

```
docker-compose up
```

The client is now connected to the API can be accessed from `localhost:7000`.

Running tests

All tests:

```
npm run test
```

Or by type:

Puppeteer-based browser tests:

```
npm run test:ui
```

Component tests:

```
npm run test:component
```

The HTML prototype Technologies and Development Tools

The HTML prototype was developed with service provider Solita during November 2018 to January 2019. The HTML-prototype is available at <http://compleap-proto.testiopintopolku.fi>. The source code and technical documentation for the HTML-prototype is available at: <https://github.com/Opetushallitus/compleap-proto>

Setup

Install npm and Vue CLI `npm install`

Dev-environment <http://localhost:8080/>

`npm run serve`

Build prod to /dist

`npm run build`

Run unit test(s)

`npm run test:unit`

Policy for software code licensing

Software created under the CompLeap Grant Agreement is published as Open Source. The Compleap software code will be licensed under the EUPL, Version 1.1 or subsequent versions of the EUPL. A copy of the Licence is available at <https://joinup.ec.europa.eu/software/page/eupl>.

National Agency for Education will be the copyright owner and the University of Oulu will transfer the copyright of software code created under their part of WP3 to EDUFI. EDUFI will be the copyright owner as a whole. Software versions are delivered to the EDUFI Github repository.

Rights of the use of the results and of pre-existing rights by the Union has been listed in the Grant Agreement Article II.9.3.