

Final project requirements

Intro

We have spent a whole semester together learning about the web, it's underlying technologies and useful tools.

It's time now to demonstrate this knowledge you have gathered by creating a real-world project.

Project Description

The project will build on top of the work you have done thus far with the CV project, finalizing it and turning it into a real web application with CRUD operations.

Screens

1. CV

- ★ The design of the template should resemble the original one as much as possible in terms of aesthetics.
- ★ Colors can be altered.
- ★ Pay close attention to custom fonts, icons etc. - all are required.
- ★ All user-related data on the page is dynamic and received from server upon load
- ★ Any place where a dynamic content should be loaded - show "Loading.." while it is loading.
- ★ Email address should be a link
- ★ Layout should be dynamic to contain any number of hobbies, skills, languages, skills etc.



2. Users screen

- ★ List all users in the database in a table, **example:**

First name	Last name	Email	Delete	Edit
Arik	Liber	liber.arik@gmail.com	Delete	Edit
Jhon	Doe	jhon.doe@gmail.com	Delete	Edit

- ★ Actions on user:
 - Delete user (button) - Show a message when user delete and remove him from table
 - Edit user (button) - Opens the CV Form screen
 - Show user CV (link) - Opens the actual CV of the selected user
- ★ "Add user" button which redirects to a new user form

3. Create/Update form

- ★ The form allows to edit (if user selected) or create new CV.
- ★ On successful user creation/edit - show a message

Example:



Basic info

Contact info

About me

Social networks

Experience

Pro skills

Per skills

Education

Title

Place

Education date

Description

Title

Place

Education date

Description

Hobbies

- ☐ Sport
- ☒ Reding
- ☐ Swimming
- ☐ Programming
- ☒ Sleeping
- ☒ Doing nothing

Languages

Hebrew level (number)

English level (number)

Russian level (number)

Italian level (number)

Important notes

- ★ The project will be carried out by teams of 2 developers
- ★ Code should be clean and clear!
Use indentations, comments, proper file, function and variable naming.
- ★ Make sure all runs, no console errors or server errors allowed

Grading

- ★ Every student will present the project for 10-15 minutes during which I will ask any kind of questions.
- ★ Make sure you are well familiar with the material.
- ★ The grade consists of 30% knowledge, 70% project.
- ★ During the exam you will be asked to perform some minor changes

GOOD LUCK!