1 Project set-up

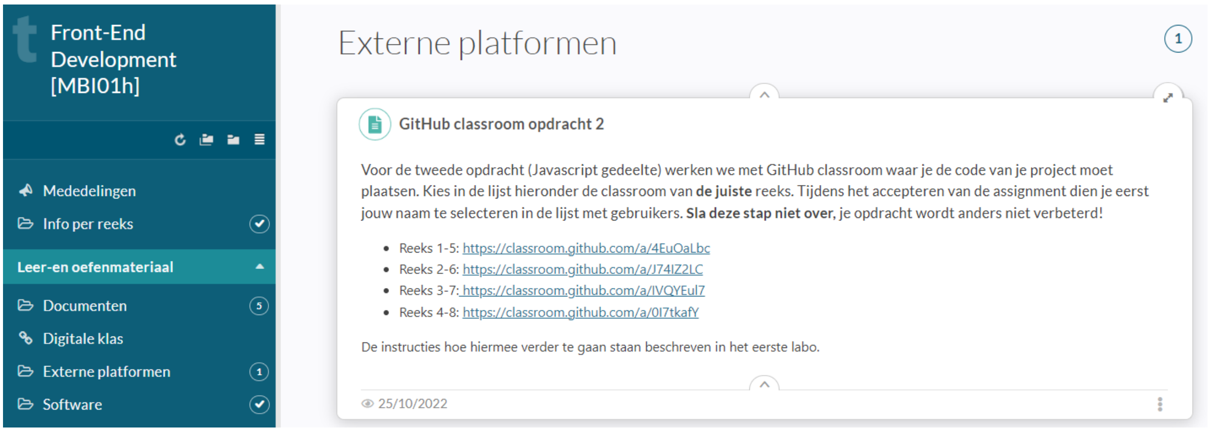
1.1 The Games Library

During this series of lab exercises, you are going to create a front-end for an existing back-end application. The application allows you to administer your games library: add and remove games from the library, add ratings for games and mark them as favourite.

Up until now, you have learned to create static websites with HTML and CSS. These skills are still necessary to build the foundations of your front-end, but from now on you are going to leverage **JavaScript** to make you website dynamic. Step by step, in each lesson, you will make a front-end app that consumes, processes and displays data from a back-end application.

1.2 Accept GitHub classroom assignment

For this project, we will use GitHub classroom to host your source code. In the first step, you will have to accept the assignment we have created for you. Go to **Toledo > Externe platformen** and click on the correct link for **your class group**:



GitHub classroom opdracht 2

Voor de tweede opdracht (Javascript gedeelte) werken we met GitHub classroom waar je de code van je project moet plaatsen. Kies in de lijst hieronder de classroom van **de juiste** reeks. Tijdens het accepteren van de assignment dien je eerst jouw naam te selecteren in de lijst met gebruikers. **Sla deze stap niet over**, je opdracht wordt anders niet verbeterd!

- Reeks 1-5: https://classroom.github.com/a/4EuOaLbc
- Reeks 2-6: https://classroom.github.com/a/J74IZ2LC
- Reeks 3-7: https://classroom.github.com/a/IVQYEul7
- Reeks 4-8: https://classroom.github.com/a/0I7tkafY

De instructies hoe hiermee verder te gaan staan beschreven in het eerste labo.

25/10/2022

Do not skip the step where you must select your name in the list. Otherwise, we cannot correct your assignment!

1.3 Clone repository

After accepting the assignment, an individual repository within the classroom will be created. An URL to this repository will be presented (you might need to refresh the page). Copy this URL.

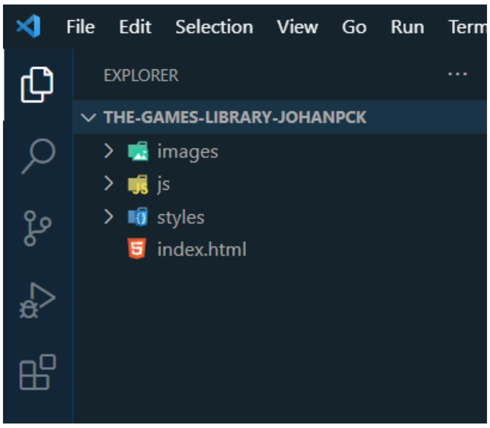
Open a terminal window (e.g. Git Bash) in a local folder (e.g. C:\front-end) and execute the following command:

```
git clone <url_of_your_repo>
```

The URL of your repo looks like:

```
https://github.com/UCLL-Frontend/the-games-library-johanpck
```

Open this folder in VSCode, a basic skeleton should already be available:



1.4 Periodically commit your work to GitHub

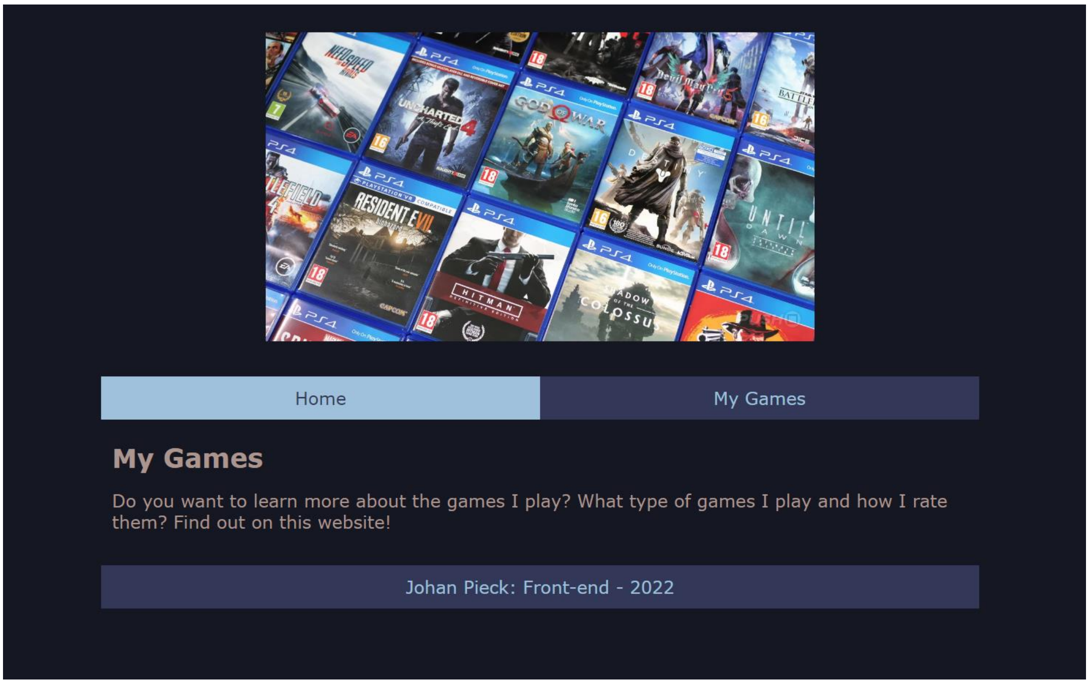
When finishing a lesson, or preferably multiple times during a lesson, commit your work to GitHub so you have a backup and version history of your work. To do this, execute the following commands from your working directory (e.g. c:\projects\the-games-library-johanpck):

```
git add *
git commit -m “Your commit message”
git push -u origin master
```

1.5 Create a static home page

The first step is finishing the static welcome page of our website for which we have already provided a basic html structure. Preview the index.html in the browser: this page still has a very basic layout. Open index.html in VSCode and look at the code. We have included some comments (between the tags <!-- -->) which will give you a hint of what to do.

Use your creativity to create a nice and shiny homepage. Here is some inspiration:



My Games

Do you want to learn more about the games I play? What type of games I play and how I rate them? Find out on this website!

Johan Pieck: Front-end - 2022

Evaluation criteria: Index page

- ❑ index.html is styled using SASS
- ❑ A logo is placed in the header
- ❑ Your name is displayed in the footer

1.6 Overview games

Open the overview.html page: this is a page that we will use to display the resulting data from our JavaScript code. On that page you will need to create an empty element with a specific ID, so we can select it in JavaScript to display data.

First you will need to include the JavaScript files, so they are loaded when opening the webpage. This can be achieved by adding a script tag in the header of your HTML:

```
<script src="js/dom.js" defer></script>
```

The *defer* attribute makes sure that JavaScript files are loaded **after** the main html. This is necessary because some html elements need to be loaded before we can target them with JavaScript.

You are going to write all your code in “games.js”, which you can find in the “js” folder. The “dom.js” file already contains some helper functions that you will need to use in games.js. Therefore, it is important to load dom.js **before** games.js.

Evaluation criteria: Overview page

- ❑ overview.html is styled using SASS
- ❑ dom.js is included in the head
- ❑ games.js is included in the head, after dom.js
- ❑ A logo is placed in the header
- ❑ Main element contains an empty div with id “status”
- ❑ Your name is displayed in the footer