

Academic year: 2022-2023
Education Bachelor of Applied Computer
Science Phase: 1
Examiner: G. Jongen, J. Pieck, J. Van Hee, A.
Vranken OPO: MBI01H - Front-end dev
Resources: Laptop, limited wifi network 'xtoledo'
Date: January 19, 2023
Start time: 13
Duration: 2 hours 20' + possibly 30% additional time.



Student: No:

Question	Points	Score
Display citations on a responsive page	6	
'Long quotes' button	3	
Adding a citation using the form	4	
New functionality with games back-end	3	
Total	16	

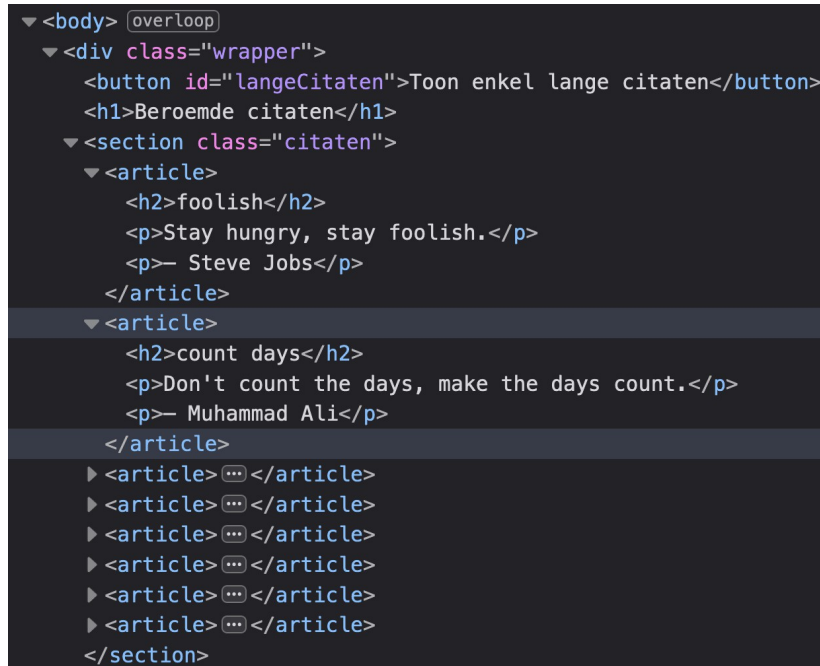
- Do not loosen the staples of this bundle.
- Connect your laptop to the restricted wifi network 'xtoledo'. The password is 'exam001'.
- No smartphone, nor smartwatch.
- At xtoledo.ucll.be you can find this task as a PDF, start code in a Zip file and two videos (one for the first three questions together and one for the fourth question)
- You hand off your HTML, CSS and JS code in one Zip file named 'FamilyNameFor-name.zip'. **Make sure this zip contains only your HTML, CSS and JS, so not the movies, the PDF of the assignment or the back-end!** Handing it off is done by uploading this zip to the xToledo assignment. For some reason, this upload is slower in Chrome than in other browsers. Please be patient if you use Chrome . . .
- Once everything is uploaded, check again in the assignment to make sure your Zip is there.
- Hand over the assignment bundle and sign the attendance list. Good luck!

This exam consists of two parts:

- Questions 1 to 3 belong together. You don't need a back-end for this. You create a small application in HTML, CSS and JS to display quotes from an array and add to this array. You get three files that all three need to be completed: quotes.html, style.css and questions123.js. Now watch the movie clip "questions123.mov. This movie without sound shows how the page should look for different screen widths (RWD), what the purpose of the top button is and how the form should work.
- Question 4 is a new problem related to the games library you created the previous weeks. However, you use a back-end *that has been modified especially for this task*. **You may not use your old back-end!** Watch movie 'question4.mov' to get an idea of what you have to make.

Question 1. Display citations on a responsive page (6 points)

In 'questions123.js' you will find an array containing a number of citation objects. As you can see, a citation has a title, the citation itself (the text), the author and the language in which it was written. For this first exercise, make sure that the quotes from the array are displayed via JS as in Figure 1, that is, with `article`, `h2` and `p` elements. Note that the language of the quote is not shown. Add the required code to the file 'questions123.js'. This part of the question is at 4 out of 6 points.



```
<body>
  <div class="wrapper">
    <button id="langeCitaten">Toon enkel lange citaten</button>
    <h1>Beroemde citaten</h1>
    <section class="citaten">
      <article>
        <h2>foolish</h2>
        <p>Stay hungry, stay foolish.</p>
        <p>- Steve Jobs</p>
      </article>
      <article>
        <h2>count days</h2>
        <p>Don't count the days, make the days count.</p>
        <p>- Muhammad Ali</p>
      </article>
      <article>...</article>
      <article>...</article>
      <article>...</article>
      <article>...</article>
      <article>...</article>
    </section>
  </div>
</body>
```

Figure 1 All quotes are read from the array and added to the DOM. The figure shows the view in the inspector of the developer tools.

Add CSS code to the 'style.css' file to show the quotes as separate boxes (see movie and Figure 2 on the next page). You don't need to use SCSS. After all, it's not about that many lines of CSS so it's not worth using Sass. In addition, you make sure the page has basic responsiveness. On large screens, three quotes are displayed side by side. As the page gets smaller (test in the 'Responsive Design View' of your developer tools), the number of quotes next to each other decreases to two and finally even one. The attached video shows what you should *approximately* achieve (it does not have to be pixel-precise . . .). On very large screens, the width of this page is reduced. This CSS is on 2 out of 6 points.

Question 2. 'Long quotes' button (3 points).

For this query, make sure that a click on the 'Show only long quotes' button shows only those quotes *whose text exceeds 50 characters* (Figure 2). In addition, change the content of the h1 title to 'Famous quotes (more than 50 characters)'. Add the necessary code to the 'questions123.js' file.

Toon enkel lange citaten

Beroemde citaten (meer dan 50 karakters)

schatbewaarder
De Vlamingen zijn de
schatbewaarders van de
Nederlandse taal
— Jan de Hartog

vandaag
Wat je vandaag moet doen,
moet je doen zoals je morgen
denkt dat je het had moeten
doen.
— Toon Hermans

aimer
Le verbe aimer est difficile à
conjuguer : son passé n'est
pas simple, son présent n'est
qu'indicatif, et son futur est
toujours conditionnel.
— Jean Cocteau

équipe
Les performances
individuelles, ce n'est pas le
plus important. On gagne et
on perd en équipe.
— Zinedine Zidane

Titel

Citaat

Auteur

Taal

Voeg citaat toe

Figure 2 After pressing the button, only quotes with text longer than 50 characters are displayed

Question 3. Adding a citation using the form (4 points)

For this last question in the quotes theme, create a form with four input fields in the HTML file:

- title;
- quote;
- author;
- language code.

You ensure (via code in 'questions123.js') that when the user clicks the 'add citation' button, the input is validated first. You check the following items:

- Is every field filled in?
- Is the text of the quote field at least 10 characters long?

If *both* conditions are met, the new quote is added to the array 'citations'. If not, an appropriate error message is displayed in the div with id="feedback" as you can see in the movie 'questions123.mov'. The two possible error messages are:

- "Error: not all fields filled in!"
- "Error: quote must be at least 10 letters long!"

You don't need to write CSS code for this question. Everything you need is already in the CSS file.

Question 4. New functionality with games back-end (3 points).

For this question, first start up the *custom* gameslibrary back-end. You can do this as follows:

- Download and extract ("unzip") the new back-end from xToledo.
- Go to this directory and start the server with **npm start** (in a terminal, e.g. VS Code itself). If all goes well you will get the message "Server is running on port 3000". Test that out by going to **http://localhost:3000** in your browser so you can see the documentation.
- For this question, you need two files:
 1. 'gamesbib.html': contains HTML and CSS. You do not change anything in this file;
 2. 'question4.js': this is where you write all the required JS code.

For this task, make both buttons work. If you click on either button, the detailed information of the game with that name will be displayed in an **ul** list. In addition, you will see in a counter (in the **h2** element) how many times *both buttons* have been clicked *together*. The movie clip 'question4.mov' briefly shows what is intended. Figure 3 shows which GET request you should use to request the information of a game via its full name to the server.

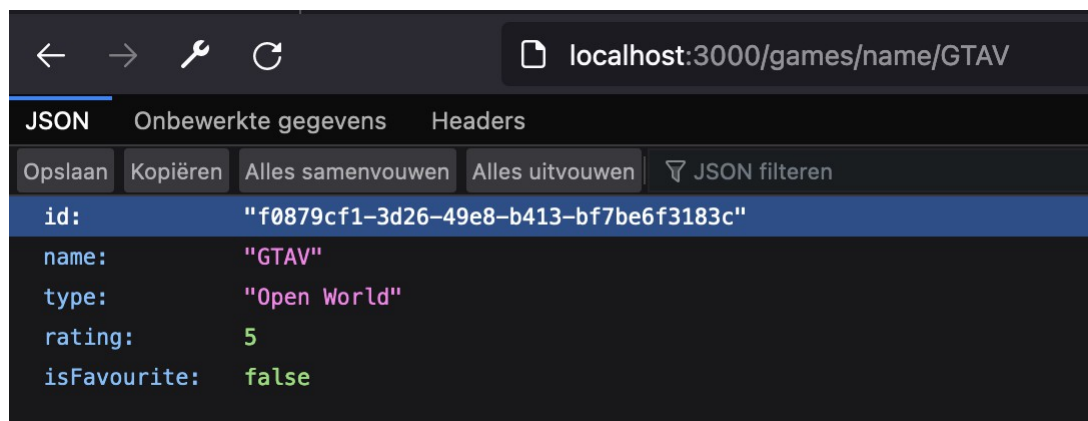


Figure 3 The result of **http://localhost:3000/games/name/GTAV**

You don't change the HTML file (which already contains all the necessary CSS in the **style** element). Just write code in the 'question4.js' file.