Testing MyTime app

Validation

HTML

HTML testing was done with https://validator.w3.org/

Test results:

1st run: 26 errors & 5 warnings
- The errors were remedied.
- All warnings referred to empty headings, but which content is created dynamically.

2nd run: 0 errors & 5 warnings (Empty heading)

CSS

CSS testing was performed with https://jigsaw.w3.org/css-validator/

Test results:

1st run: 102 errors & 756 Warnings
- W3C Validator cannot parse :root variables(90).
- 14 errors which were other than color parse Errors were remedied.
- Warnings were mostly unknown vendor extension warnings to do with bootstrap or font awesome (692). A part of those referred to overdriven button style features.
- Other warnings (64) were as well unknown vendor extension warnings. The extensions were automatically added by VSCode to transitions, border-radius and transformation declarations and were removed.

2nd run:
- 88 errors
- 692 warnings (mainly unknown vendor extension errors - bootstrapcdn & font-awesome)

Example of color parse error:

133	`.fa-home`	Value Error : color Parse Error `var(--clr-home)`

Example of unknown vendor extension warning:

6		`-webkit-text-size-adjust` is an unknown vendor extension

Example of Same color warning

6	`.btn-outline-warning:hover`	Same color for `background-color` and `border-color`

Some adjustments have been done to the CSS code after the test, to address issues that came up in testing.

JavaScript

JavaSript was checked first for correct writing and syntax using JSHint, which was installed as an extension to VSCode IDE.

- At the testing round it made 39 suggestions for either missing or extra semicolons, which were addressed.
- Also, 6 items were changed to dot notation.
- JSHint gives a warning for all ES6 featured functions and expressions (let, const and arrow functions).

JSHint Metrics:

Script.js Metrics

There are 8 functions in this file.
Function with the largest signature take 1 arguments, while the median is 0.
Largest function has 30 statements in it, while the median is 6.5.
The most complex function has a cyclomatic complexity value of 6 while the median is 1.

TasksScript.js Metrics

There are 22 functions in this file.
Function with the largest signature take 2 arguments, while the median is 0.
Largest function has 41 statements in it, while the median is 3.
The most complex function has a cyclomatic complexity value of 5 while the median is 1.

Record.js Metrics

There are 17 functions in this file.
Function with the largest signature take 1 arguments, while the median is 1.

Largest function has 45 statements in it, while the median is 5.
The most complex function has a cyclomatic complexity value of 7 while the median is 1.

Stop.js Metrics

There are 17 functions in this file.
Function with the largest signature take 1 arguments, while the median is 1.
Largest function has 45 statements in it, while the median is 5.
The most complex function has a cyclomatic complexity value of 7 while the median is 1.

Otherwise no automated testing was done on JavaScript files. Lots of manual testing instead.
For this instance Jasmine testing did not seem necessary.

Responsiveness and compatibility

Responsiveness

MyTime was designed for mobile devices, but can be used on any screen. On big screens the application fill the left side of the viewport and stretches always till the whole viewport height.

The application max width is limited to 430px. The overflow-x starts at 320px.

Application was tested with the browser tools in Chrome, for different size mobile devices and screens for responsiveness. In the development phase both browsers were used.

Device	index.html	recording.html	Notes
iPad	Good	Good	Side and long, both work
iPad Pro	Good	Good	Side and long, both work
iPhone X	Very good	Very good	Optimal screen size
iPhone 6/7/8 Plus	Good	Good	
iPhone 6/7/8	Good	Good	
iPhone 5/ SE	Fair	Fair	Works, but the elements look little squeezed.
Pixel 2	Good	Good	
Pixel 2 XL	Good	Good	

Galaxy S5	Fairly good	Fairly good	
Laptop (MPDI) screen	Good	Good	
Laptop with touch	Good	Good	

Browser compatibility

Chrome, Firefox, Edge, Dafri and Opera were used for testing browser compatibility:

	Intended appearance	Intended responsiveness	Notes
Chrome	Good	Good	
Firefox	Good	Good	
Edge	Fairly good	Fairly good	The scroll bars are visible on unintended fields
Safari	Good	Good	
Opera	Fairly good	Fairly good	The scroll bars are visible on unintended fields

User testing

The application was tested by three volunteers.

The interface is self explanatory. It is easy to navigate through the application and its purpose is clear. Although the design is intuitive and easy to use, the color schema is not very exciting. The test users thought the application is useful, but were expecting more engaging experience. Only the parts that are ready could be tested, but the idea of the application and how it might help the user as the design is intended to was as well clear.

Self testing

Steps.
1. Open the application through the link https://junon72.github.io/MyTimeApp/
   - The application appears on the screen
2. Give a name for a project
   - Pressing Enter without writing anything on the input field prompts a message 'Give your project a name'
   - Pressing OK, the input field becomes active.

- Giving a name with only empty spaces prompts the same message.
- After writing a name with (any) characters the Create button activates.
3. Create project.
    - Pressing create button opens new page 'Tasks'

First deviation.
- Navigating to the next page from the bottom navigation prompts a message 'would you like to give your project a name?
- Pressing OK taks back to the index page.
- Pressing Cancel gives the project a nameProject(default), but allows creating tasks

4. Creating tasks.
    - Pressing plus button prompts a message 'Give your task a name'.
    - Pressing OK activates the input field.
    - Writing a task name with (any) characters and pressing the plus button prints the name on the screen with a minus button.
    - Button with a text 'New Task' appears.
    - Pressing the button gives a new input with a plus button.
    - Pressing the minus button deletes the task.
    - Entering a name that is in the list already prompts a message 'Name already exists.'

Second deviation.
- navigating back to index page and back to tasks page maintains already given name and tasks.

5. Pressing the Start Project opens Recording page.
6. Pressing a recording button carrying a task name 'activates' the task and the time display next to the button.
    - Pressing another button deactivates the previous one.
    - Clickin Defaults tasks at the bottom of the tasks field opens the default tasks Lunch and Break.
    - Pressing one of them de activates the previous task.

The timers are not functional (yet).

7. Pressing the Pause button at the bottom of the screen prompts a 'Paused' message and deactivates all the recording buttons.
8. Pressing OK activates the buttons again. The task which was last active, is marked.
- pressing any task button removed the marking.
9. Pressing the Stop button at the bottom of the screen opens a Stop page. None of the task buttons are active.

Third deviation:
- Moving back and forth with the bottom navigation and top navigation, keeps the tasks and allows free navigation without losing the information.
10.   Pressing the Checked button opens Timesheet page.
11.   Pressing the Share button opens Share page.
- Input fields are active, you can write on them. The email input field gives red edges if it doesn't detect @ sign.
12.   Pressing the Send button opens Project Complete! Page.
13.   Pressing New MyTime Project button opens the index page.

Glitches and inconvenienced I encountered while testing, and which remain unresolved for now:

530px is the absolute minimum height the application is still somewhat workable.

The time displays of the task buttons appear to have event handlers as well and react to pressing them. This is unintentional. They should be detached from the button.

When the task list is long and the window size over 430px and the task list is long, the left side border disappears from the bottom part of the list.