Anita Budhathoki

Profess Freeman Lo

ITEC 225

9/22/2021

Mini-Project 2- Morsecode (SDLC)

The objective of this mini project was to bring together much of what we have discover so far in developing in this language called JavaScript by integrating HTML/CSS. Some of the precise requirements for this simple project are utilizing Opensource Background Patterns, Google Fonts, Opensource Icons, Quotes API, Pure CSS, Favicon Generator, and a page Loader etc. When talking about feasibility study, everything that was needed for this simple project was provided or was available for free. This project is to grasp some understanding of how to build a simple app by utilizing Opensource Background Patterns, Google Fonts, Opensource Icons, Quotes API , Purecss, Favicon Generator. By building a simple morsecode application and a password generation JS application. Students will be able to gain SDLC skills and fundamentals of JS programming on how to solve mathematical and real software engineering tasks.   
The project has met the basic requirements, professor’s expectations and current software can support the software.

First, I created the basic HTML file for my page. I used Purecss form to stack and added CSS in the html file so it can handle the styling for us. Language is set to English and meta condense are inserted as well. Files should be placed to rucs.radford.edu and github In my html file, I inserted all the source code along title, viewport for the visual portion of the screen. I have separate CSS file where I have to used it to make amazing back image. Moreover, I also have used site like font-awesome, hero patterns, favicon generator to make my page design better and its layout. Before I started writing the Java-script for this project, I checkout the links that you provided, and they really helped me explore and know more about how I can design my page. For this project I have of four files and two images for favicon, two of them are strict to being Java-Script file, one is HTML and one for CSS styling.

Secondly, I wrote some functions to convert the English text to Morsecode and Morsecode to English text which was not difficult text as I did lot of research and the professor you helped me figure out. I created some validation functions to make sure whatever users enters is going to be transform into morsecode because in morsecode not all characters are included so that means it checks for the valid input status for morsecode making sure it is not characters that cannot be transformed, null and if error if found then it will give alert message if not it will convert to text in uppercase. I also have some functions which handles the characters other than roman scripts that are used in morsecode example

Graphical user interface, text, application

Description automatically generated

I have tagged the js, css files to html as required and it works perfectly fine. , I used script strategy to add JavaScript in this HTML file. It is very important to declare the java-script or css whenever using in the html.

Text

Description automatically generated

Additionally, I have function which will clear the input and output box when the user hits the clear so the user can try new inputs.

Text

Description automatically generated

In my passwordscript java-script file, I have a function which converts morsecode to text and these are roman script and numbers used for morsecode plus some validation functions. I used [Ready to check - Nu Html Checker (w3.org)](https://validator.w3.org/nu/#file) to test html and fix the error. Testing is very important, and it continues until software is working according to the expectations, is bug-free, and stable for that system. To maintain the system over period, it might need fixing, upgrade and enhancement. The main concentration here is SDLC should be met continuously, and system continuously perform as the way user want.