**TASK 1**

**introduction to workflow and sass (Part 1)**

**7.5.2022**

The course accomplishment started by checking out the descriptions presented on the General Course Information page; according to this page, it was tried to be generally familiarized with the other course assignments as a preliminary overall course review. On the Environment Setup page, instructions regarding the installation of Version Control (Git) and code editors were skipped entirely. Since Git commands are sometimes challenging to use, especially whenever there is a conflict between commits or pushed tasks, another Git course from Tampere University was added to the course tasks as an external reference. The main programming languages used during the course were well-known *HTML*, *CSS*, and *Javascript* languages, but the Sass package and other code editor extensions were used for the first time. Therefore, related packages were added to the VS code, preparing to begin coding tasks.

**8.5.2022**

The project's local repository was made via the command *mkdir* , and VS was called to add two folders by the names of *dist* and *scss;* then, an *HTML* file named *index.html* was added to the *dist* folder and a *CSS* file named *main. scss* to the *scss* folder as well. During the opening of the VS, an error message related to *node.js* and *sass* was encountered, and the problem was solved thorugh reinstalling the *node.js*accordingly. *Node-sass* was appropriately acting to compile *SCSS* files to*CSS***.** Once all essential tools were prepared, the *node-sass* package was used to convert the *main.scss* file to the *main.css* file containing styles to be added to the *index.html.*

**9.5.2022**

Version control course (Tampere University) contains four modules as the introduction, basics, intermediate, advanced, and GitLab, respectively. Tha primary purpose of this Git course accomplishment is to review related contexts and practice more. The introduction of the Git course has been studied, and according to this module, the Git structure could be depicted in the following figure 1, showing the main commands and data flow paths between repositories.

Diagram

Description automatically generated

Figure 1:Git version control and main commands

**11.5.2022**

During this day, GitHub and Gitlab remote repositories were being checked out to be ready to save the course example project and other tasks such as the final project. Git basic commands used to move course files between local and remote repositories were reviewed, the example project was initialized, and some files and changes were added and commited to push the first version of the example project to the remote repository.

**TASK 2**

**Homepage and Core Sass/CSS (Part 2)**

**16.5.2022**

After primary project configurations, the main coding task was started by developing the <body> tag of the index.html file, the first page of the project, by adding a <header> tag containing <nav> and <div> tags. The new project version, including the last modifications, was committed and pushed to the remote repository.

**17.5.2022**

In the **<body>** tag we added a **<main** id=” home”**>** tag which contains **<h1** class=” lag-heading”**>** and**<h2** class=”sm-heading”**>** tags and a **<div** class=” icons”**>** containing <**a**>tags and related **<i**> tags as well.In this project used the Awesome icons which would be include into the Html coding files via a <**script**> tag written into the end of **<body>**tags containing the awesome website’s link as following Html code:

<script src="https://kit.fontawesome.com/a9f8805286.js" crossorigin="anonymous"></script>

Another <**script**> tag containing the directory of the project JavaScript file (main.js) has been added to the end of the **<body**> tag as well.