# 

**CSE300: Software Engineering**

**Monsoon 2022**

**Group 7**

**Group Members**

| **Name** | **Enrolment Number** | **Email-id** |
| --- | --- | --- |
| Sakshi Shah | AU1940213 | sakshi.s5@ahduni.edu.in |
| Astha Patel | AU1940312 | astha.p@ahduni.edu.in |
| Kareena Matwani | AU1940314 | kareena.m@ahduni.edu.in |
| Vraj Parikh | AU1940185 | vraj.p1@ahduni.edu.in |
| Rushil Borad | AU1940179 | rushil.b@ahduni.edu.in |

# **Coding Standards Documents**

[**Coding Standards Documents**](#_of0u52nbvvix) **1**

[1. Coding Style](#_ra5in11kl1kt) **2**

[1.1 Website Structure](#_yhuj32bqstf8) 2

[1.2 Template Style](#_yiru5mfx91yf) 2

[2. Indentation](#_w2o5vek9dt8m) **3**

[2.1 Spaces, Tabs and Indentation](#_df8ea3s0sbs4) 3

[2.2 Formatting of Lines for Source File and Modules](#_pulrhgchjts3) 3

[3. Comments](#_wccmvueslyii) **3**

[3.1 Block Comments](#_uh2cxck214u3) 3

[3.2 Inline Comments](#_x1qfllpa7rxt) 3

[4. Import and Configuration](#_dks9ta8f6wf) **3**

[**4.1 Error Handling**](#_t82n44unhsqa) **4**

[**4.2 CSS and Javascript**](#_xo33aomv9gfd) **4**

[4.2.1 Style Rules](#_ivk1mswoz7jx) 4

[4.2.2 Formatting Rules](#_hhbpo3t2scoz) 5

**PHP and MySQL**

# **Coding Style**

## **1.1 Website Structure**

The maintenance of the codes play a very important role, the uniformity of the same should be generated and should be understood by each member in the team. The readability and reusability of the code thus gets improved when the efforts are made on maintaining the uniformity across all the file and pages.

The files in the task management system are separated by their features and functionality and this creates an easy testable environment , and in such a way individual functionality testing can be done.Moreover , different folder for css , javascript and php files are made for easy navigation of all the teammates working together.

## **1.2 Template Style**

Here , In this part we tried to embed frontend files and make them look aesthetic while making them using javascript , CSS , Bootstrap . The connection at the backend was made using PHP and MySQL , at backend different databases are created through PHP retrieves information and displays it to the Admin and User utilizes different functionalities.

The common variables used across all the codes and the availability of the libraries in all the pages and codes are maintained.

# **2. Indentation**

## **2.1 Spaces, Tabs and Indentation**

* When there are more than 20-25 files the probability of having confusion is very high so proper spacing , Indentation of each loop , each function , each class or division created is a must.
* The tabs should be given considering there are loops and functions. In CSS and javascript files the spaces and tabs are well maintained so as to have a differentiation in understanding borders , buttons and backgrounds etcetera.

## **2.2 Formatting of Lines for Source File and Modules**

* Wrapping up between and among the sections are required , continuation in each section of code throughout the php file has to be organized.
* Nested block indentation and spacing is required , in nested queries while writing mySql queries , spaces and indentation are required.

# **3. Comments**

## **3.1 Block Comments**

* Synchronizability of the code while maintaining the understandability of the code , so when there is diffusion of ideas and features put on a project the project code still remains understandable.
* Multi line comments in the beginning are more suitable for the reference of the page about what features or functionalities are implemented here and related to which part of the website.

## **3.2 Inline Comments**

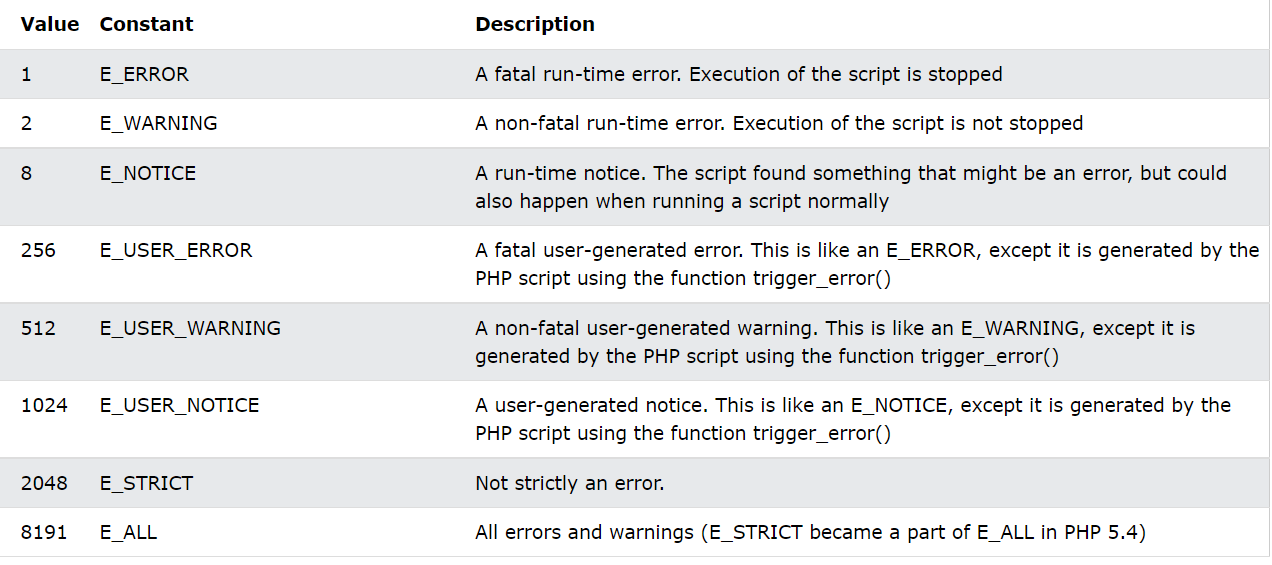
* Considering the spaces , the comments are before each section and at every line the comments are registered explaining about each function called and each if else condition maintained.

# **4. Import and Configuration**

* **Config.php file** is coded and included in every file so that the connection with the database created at phpmyadmin is maintained.
* Login session enables the user to register and authentication is done through php retrieval of database storage. If the admin adds the user it is authenticated and retrieved.

## **4.1 Error Handling**

* In PHP error checking messages are embedded so that the user does not feel stuck if the event it tried to implement could not be implemented. 404 error not found is the php file generated if the user navigates or takes actions which are not acceptable or understandable.
* die() error messages are generated as a trigger message to notify the errors to the user , also the custom errors can also be triggered to the user.
* Error report levels are also used for the error handling:



[Source](https://www.w3schools.com/php/php_error.asp)

* In MySql disable the abort on error command has to be executed so if there is an error it can be handled relatively.

## **4.2 CSS and Javascript**

### **4.2.1 Style Rules**

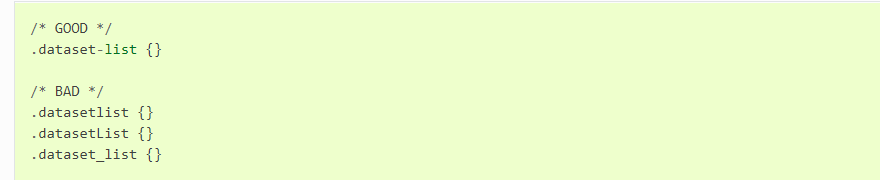
* The use of dot (.) should be done to access attributes of a variable. Although, it has a special meaning when appearing in variable sections.When the template detects a variable, like {{ variable }} which should consist of any alphanumeric characters and the underscore, it should never start with an underscore.
* The variable attributes beginning with an underscore are generally considered private variables.
* The Value , selector and property tags should be well maintained.

### **4.2.2 Formatting Rules**

* The use of two space indentations is must with the double quotes referred at every point while writing css codes.
* The coding was done considering the hex color codes more than using rgba() as in reference to all the codes.
* fall back properties should be there considering the older websites as reference.
* In CSS , we tried to not have block indentations. Sometimes these indentations may create confusion.

**Naming Conventions**

* Tried to have lowercase letters and hyphen separations between them.



[Source](https://docs.ckan.org/en/2.9/contributing/css.html)

**Commenting**

* Descriptive , Multi line comments describing the block of code.
* Definition of all child blocks is not required if the idea is explained well in the parent block like calendar.php has information regarding all declarations made so this becomes easy to understand for child blocks as well like insert.php and update.php.
* Readability becomes smoother if the comments are made crisp and considering the technical language.

**Control Structures**

* Sequential , Iterative and Selection control structures are followed considering javascript.
* Blocks inscribed inside the control structures like if else, switch case and Loops are followed with none or many statements , respective executions will be made , also if the indentations are implemented the manual checking and testing of all the functionalities would be made simple.