



Digital BlackSmiths: Thutong LMS

Coding Standards Document

Group Members:

Fiwa Lekhulani

Daniel Rocha

Lebogang Ntatleng

Lesego Mabe

Tlou Lebelo

Contents

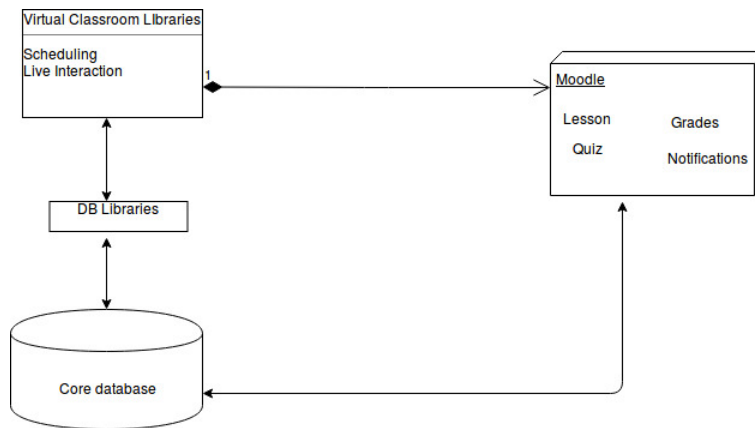
This document contains the coding conventions created by the Digital Blacksmiths team, based on the various languages used to create the Thutong Online Learning System.

The team makes use of Moodle, JavaScript and *** for the creation of the system.

Moodle is purely coded in PHP and coding standards are defined at https://docs.moodle.org/dev/Coding_style

The rest of this documentation will describe the coding conventions used for the remainder of the languages used.

Detailed System Design



Coding Conventions

To ensure that the code is flexible, readable, reliable and efficient the following aspects are considered and given the stipulated rules:

Naming Conventions

- Classes, functions and variables are to be named to be descriptive as to what their purpose is.
- e.g

Layout Rules

The structure of each file must adhere to the following layout:

- Each file must have a file header wherein which the location, version, author and update history
When a change is made to the file, the developer is required to change the date for last updated
- Only one class is allowed within a file, which may have infinite internal functions
- A class must have a description for its overall functionality and its internal functions
- No more than one blank line is prohibited between lines of code, but with that said functions must be separated with a blank line for clarity and readability
- Every class definition, function and loop must be indented. The beginning brace must be placed in the same line as the definition of the loop or function but the ending brace must be in its own line followed an empty line.
- Code must be neat and self-explanatory, comments should be used when an action is fairly complex or needs additional instruction to another programmer.

Commenting Practices

- File header are composed of multi-line comments
- class or function definitions are made preceding the function using single line comments; these comments must be brief and solely give the core purpose of the class or function.
- Single line comments are to be used to give an explanation of what a specific line does; these should be only used when necessary. Avoid redundant commenting that can be easily deciphered from code naming conventions, methods and functionality.

File Structure

Code Review Process

Each members' code uploaded to GitHub is reviewed ****(after it is committed/weekly/monthly)**** to ensure that it complies with all above-mentioned standards and so that all group members understand the purpose of newly uploaded code for overall understanding of the system and