# To Do List Assignment

Created by: Alex Krebs

Functional Requirements

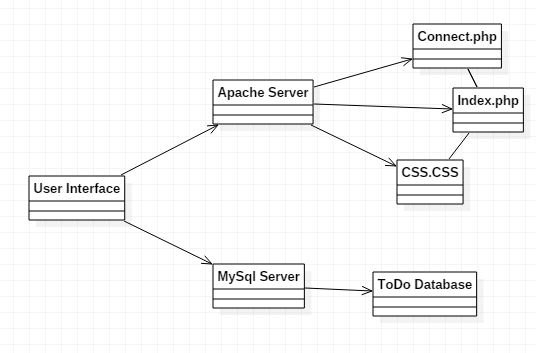
The functional requirements for this assignment are as follows:

1. The ToDo list must be connected through a front end HTML interface, using a PHP and MySql backend for data storage.
2. It would be desired to have an intellectual and visually appealing front end, so the user can navigate through easily.
3. The data must be stored on the Apache server for access using the local host.

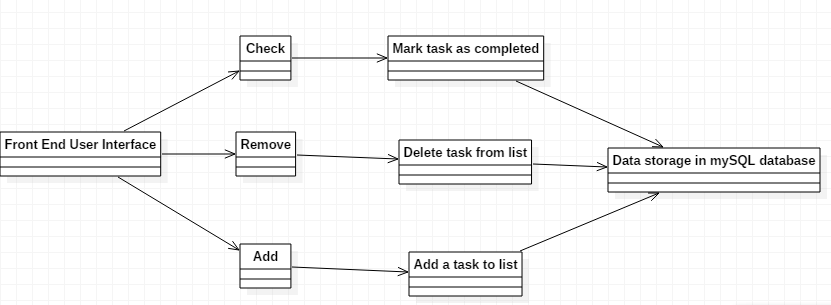
Non-Functional Requirements

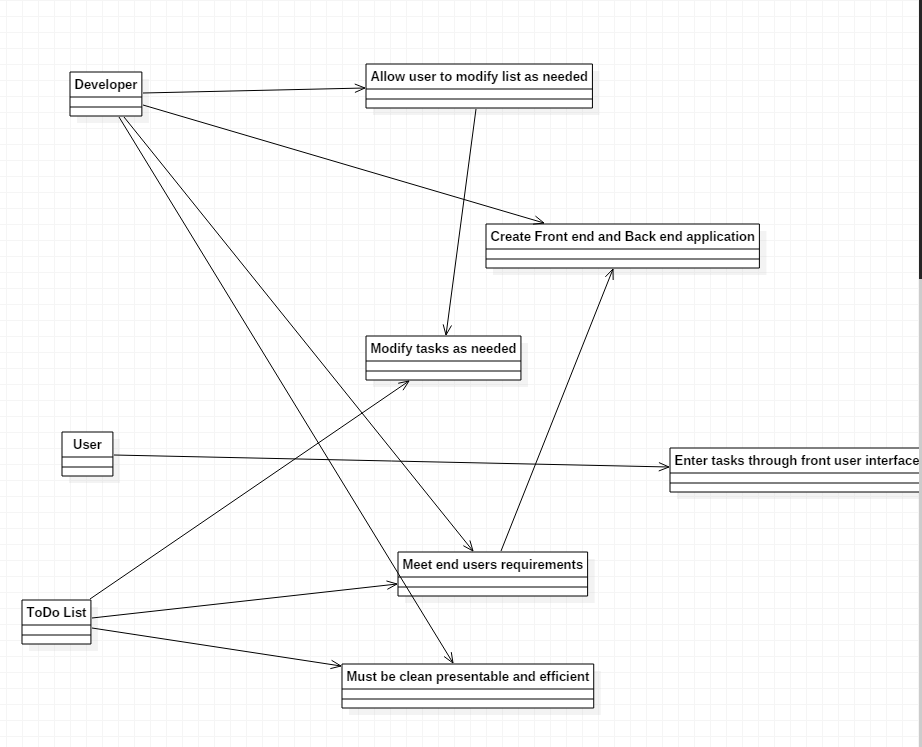
1. The ToDo list should be able to add, view, and delete tasks.
2. The deployment of this list is through the Apache server using PHP and MySql
3. Documentation for this document is a must, as it will provide the user with information on how to use the application properly.

System Architecture Diagram

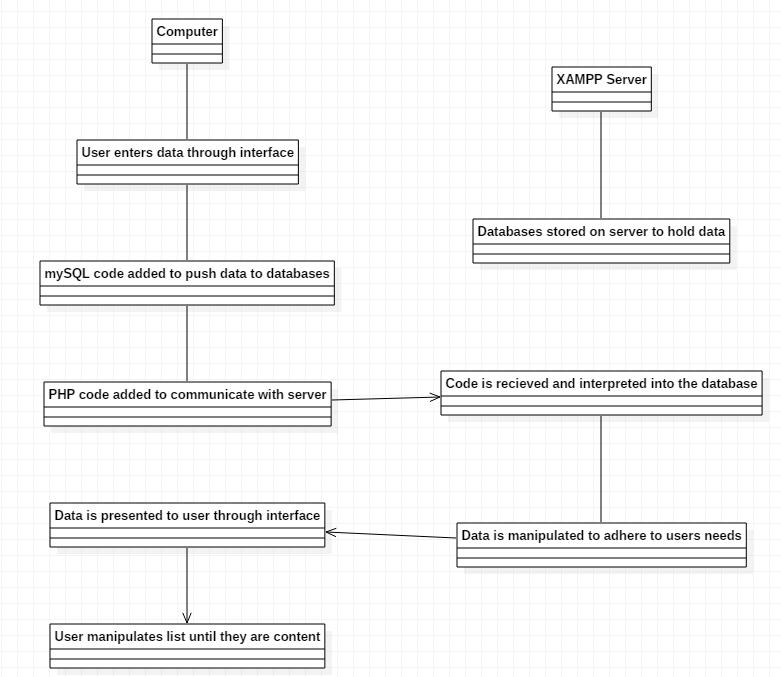


Data Flow Diagram



Use Cases

Sequence Diagram



Database Design

My database design consists of 3 tables: tasks, completed and users. The tasks table stores the information that is to be entered by the user, the completed table will hold the information that the task has been completed, and the users table will hold the information pertaining to the user and their list.

Class Diagram

No classes were added in this assignment, but if there was to be 3 classes added, I would have:

* storeData class, that would have the code pushed to the database and stored.
* pushData class, which would push the data from the PHP query to the SQL database
* arrangeTasks class, which would be more so for the user front end, allowing the user to move tasks up or down in the list and rank them by priority.

Test Case

**Name of task:** Learning how to use XAMPP and myPHPAdmin

**Estimated No of Hours**: 1

**Actual time spent:** 2

**Notes:** Had to learn how to use both programs, and add the data to the databases through the php and MySQL coding

**Issues:** Finding the proper material to learn how these programs function

**Name of task:** Learning PHP and mySql

**Estimated No of Hours:** 3

**Actual time spent:** 6

**Notes:** Using an HTML interface for the user and a PHP and mySql backend, I was able to connect to the database, allowing me to store and view information that was used in my code.

**Issues:** My biggest issue is being able to use the add and remove button to use for the tasks. I am able to hardcode the tasks into the database, but struggled with being able to use the interface.

**Name of task:** Completing documentation

**Estimated No of Hours:** 1

**Actual time spent:** 2

**Notes:** Created detailed information and diagrams about how my todo list application was setup.

**Issues:** Making sure the diagrams properly portrayed what was needed seemed to be the most difficult for me.