**Work break down:**

Name of the task: Installing source tree and create the repository and testing it

Estimated No of hours: 30 min

Actual time spent: 2 hours

Notes: I followed a YouTube video to perform this step; however, I needed some research on which GUI I should choose for this course

Issues: The installing part was a straight forward process. I spent the time on how to add the contributors to my repository. Also, I committed a wrong file that I needed to remove from the repository.

Name of the task: Installing WAMP server

Estimated No of hours: 10 min

Actual time spent: 45 min

Notes: I had to research the difference between WAMP and XAMPP. My research concluded that WAMP is mainly for Window operating system and it uses mySQL which I am familiar with. I also found that it is not recommended to install both on the same machine because it will cause conflicts in the port number.

Issues: No issues.

Name of the task: Connecting to the database using mySQL

Estimated No of hours: 10 min

Actual time spent: 30 min

Notes: no need for password if logged in as root.

Issues: I had previously worked on a database project and I had set up a password for logging in. I assumed that the connection password for WAMP would be the same. It turned out there is no need for the password.

Name of the task: Installing Netbeans

Estimated No of hours: 30 min

Actual time spent: 1:30 hr.

Notes: must use JDK8 instead of JDK9

Issues: the installation part was straight forward. But, when trying to create a new project the options list was not showing up to select the project type. After a long time researching the cause of the problem, I found out that I should delete JDK9 and install JDK8 and set the netbeans configuration file to the new JDK8 (destination path).

Name of the task: Linking the project to MySQL database

Estimated No of hours: 30 min

Actual time spent: 15 min

Notes: created config.php file that has the credentials to connect the project to the database. This file is required in all the php file in the project.

Issues: no issues

Name of the task: create an easy interface

Estimated No of hours: 1 hr

Actual time spent: 3 hr

Notes: I looked up how to create animated buttons, colors, fonts, etc.

Issues: no issues

Name of the task: activating each button to perform a certain task

Estimated No of hours: 2 hr

Actual time spent: 10 hr

Notes: this was the most challenging part because the ‘add task’ and ‘Delete’ button performed tasks related to the database.

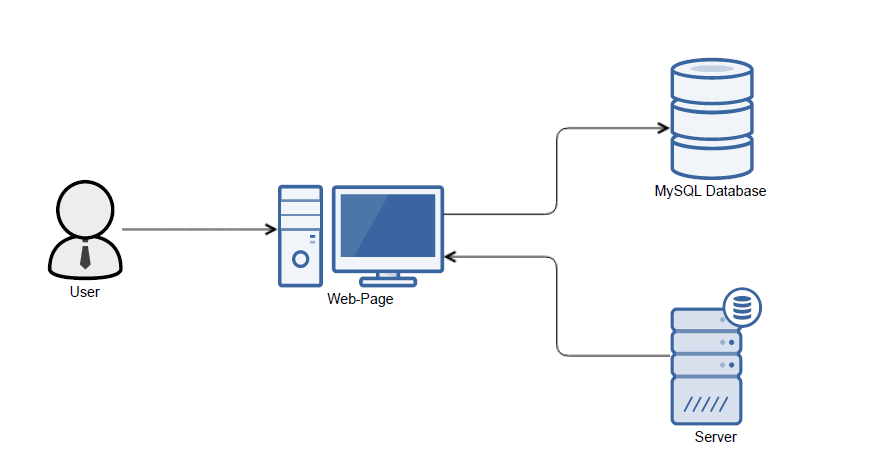
Issues: I needed to look up how to extract the data from the inputs, activated the checkbox and presenting the data from the database.

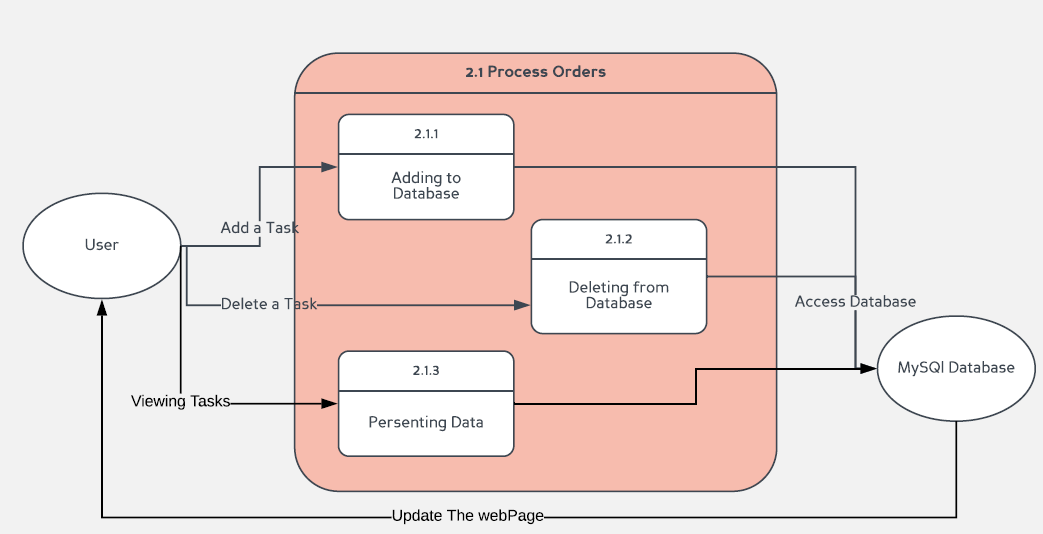
**Functional Requirements**

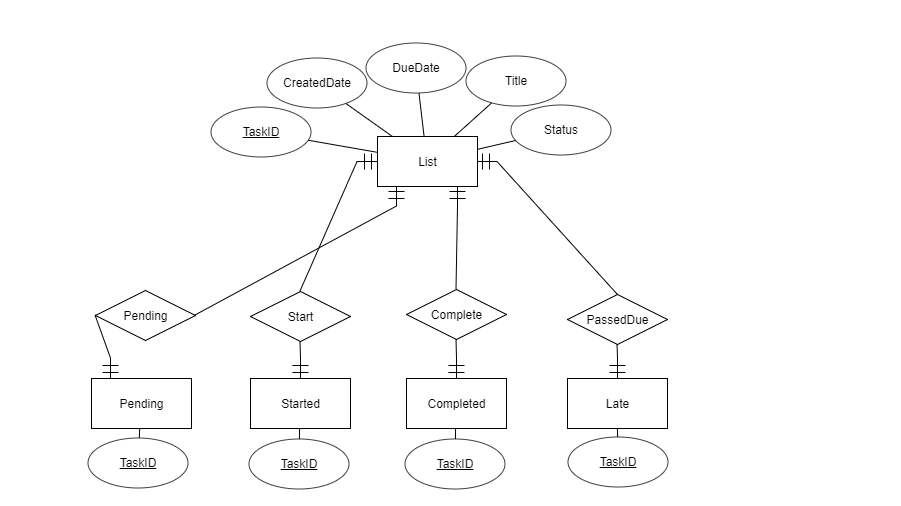
1. All the WAMP services should be running.
2. The project must be successfully connected to MySQL database in order when accessing the web-page.
3. The presentation of the data should be reading the database content.
4. Added tasks will automatically be synced to the MySQL database.
5. Deleted tasks will automatically be removed from the MySQL database.
6. Each task category web-page must present the number of tasks that are within the category.

**Non-Functional Requirements**

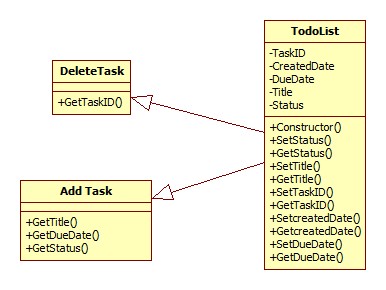
1. Create a .sql file that contains all the tables and some pre-inserted data in order to check if functionality of the project.
2. Easy to navigate website with pleasant colors and fun experience for the different types of users.
3. Limiting the input options by using dropdown menu and date picker.
4. Utilizing the checkbox feature to delete multiple tasks in one click.
5. Created a style file that allow the developer to change or improve the customization of the user interface.
6. Unifying the interface across all the web-pages.

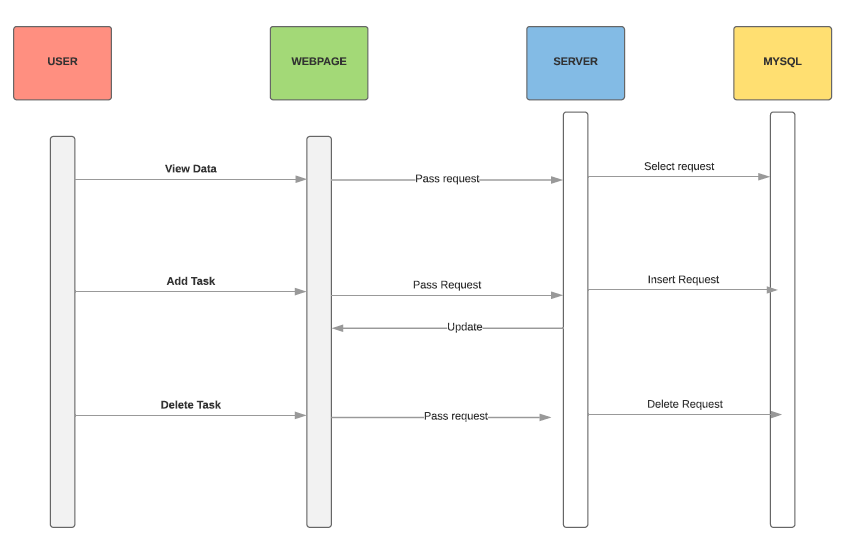
 **System Architecture Diagram**

**Data Flow Diagram (Level 2)**

**Database Design**

**Class Diagram**



**Sequence Diagram **

**Test Cases**

1. **Title:** Check the MySQL connection

**Description:** The web page should present the pre-inserted data with do errors.

**Test Steps:**

* 1. Run the project using the default browser
  2. Browse through the buttons on the top of the pages

**Expectation**: A list of tasks on each button click

1. **Title:** Check the number of tasks per task category page

**Description:** each task category web page should present the number of tasks in each category.

**Test Steps:**

* 1. Run the project using the default browser
  2. Browse through the buttons on the top of the pages
  3. Check the bottom of each page for the number of tasks
  4. Check if the number matches the actual number of tasks

**Expectation**: The number of tasks presented in the bottom of each task category web-page should match the actual number of tasks.

1. **Title:** validating SQL statement used in this project

**Description:** to avoid having any type of errors while using the to-do list web application. Each SQL statement should be written correctly to present the expected output or perform the expected action.

**Test Steps:**

* 1. Install WAMP
  2. Go to http://localhost/phpmyadmin/
  3. Log in
  4. Click SQL tap
  5. Input each SQL query.
  6. Check if it is meeting the expectation.

**Expectation**: Each SQL query in this project is written correctly.

1. **Title:** Adding a note.

**Description:** Adding a note will automatically update the all tasks button and the category which the task is listed under. Adding a task to MySQL database should lead to update the presentation of the data

**Test Steps:**

* 1. Launch the browser
  2. Click in add Task
  3. Input all the required data
  4. Click Add Task button
  5. Click on all tasks and the task category which you chose

**Expectation**: The new note show be present in both tabs.

1. **Title:** Preventing the user from inputting incomplete information when adding a task

**Description:** the system should prevent adding incomplete information when adding a note to the database

**Test Steps:**

* 1. Launch the browser
  2. Click in add Task
  3. Input incomplete data.
  4. Click Add Task button

**Expectation**: The new note show be present in both tabs.

1. **Title:** Deleting a note.

**Description:** Deleting a note will automatically update the all tasks button and the category which the task is listed under. Deleting a task to MySQL database should lead to update the presentation of the data

**Test Steps:**

* 1. Launch the browser
  2. Click in All Tasks
  3. Check the box next to the task you want to delete
  4. Click Delete Task button
  5. Click on all tasks and the task category which the task was listed under

**Expectation**: The deleted task show be deleted in both tabs.