Shrey Patel

CSC 4996

To Do List Application Documentation

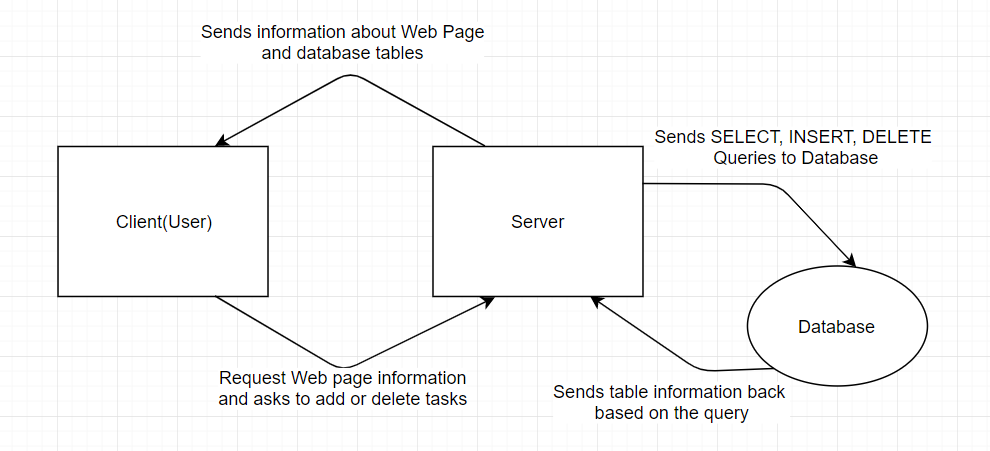
**Functional Requirements**

1. Able to view all the tasks that are stored in the MySQL database
2. Able to add new tasks, their due date, and task description to the database
3. Able to delete tasks from the database

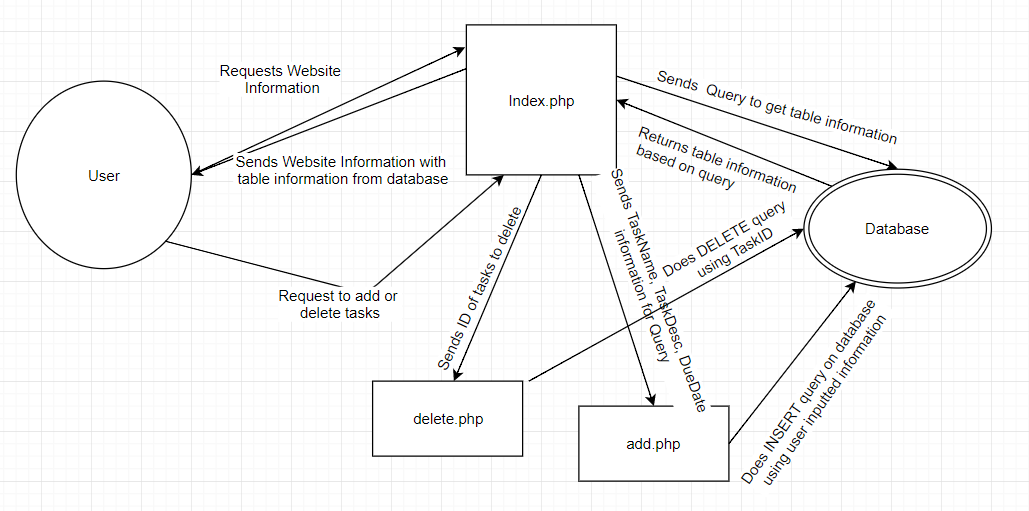
**Non-Functional Requirements**

1. Efficiently performs tasks of adding and deleting rows from the database
2. Can be modified further to add more features or variables to the table in the future
3. Has reusability for future. Can use the code of outputting table from the database and inputting user data into the table or removing data from the table in other projects

**System Architecture Diagram**



**Data Flow Diagram**

****

**Use Cases**

Use Case 1: Add task

Actor: User

Description: The user can input their own task, with information about the task name, a task description, and a due date for the task and insert it into the table. All of the fields in the form must be filled out to be able to add a task.

Conditions: Logged into the database, All information fields for a new task are filled out

Use Case 2: Delete 1 task

Actor: User

Description: The user will select a task from table and in the last column, named Remove Task, click on the checkbox to select that task. Then click on the delete button below the table.

Conditions: Logged into the database, Correct task selected by clicking the checkbox, Table must not be empty

Use Case 3: Delete multiple tasks

Actor: User

Description: The user will select all of the tasks from table and in the last column, named Remove Task, that they wish to remove from the table and click on the checkbox to select them. Then click on the delete button below the table to remove all of them at once.

Conditions: Logged into the database, Multiple tasks selected by clicking the checkbox, Table must have more than 1 task

**Test Cases**

Test Case 1: Add Task

Description: User will add a new task to the database.

Steps:

1. Fill out Task Name, Task Due Date, and Task Description under New Task
2. Make sure all three fields are filled in before clicking on add task
3. The page will redirect itself and the changes will have been made and shown in the table above

Test Case 1: Delete Multiple Tasks

Description: User will remove a task from the database.

Steps:

1. Find the tasks you want to delete and check the checkbox in the last column of each row
2. Click on the Delete button to delete those tasks that are selected
3. The page will redirect itself and the changes will have been made and shown in the table above

Test Case 1: Undo Delete Task (Not Implemented)

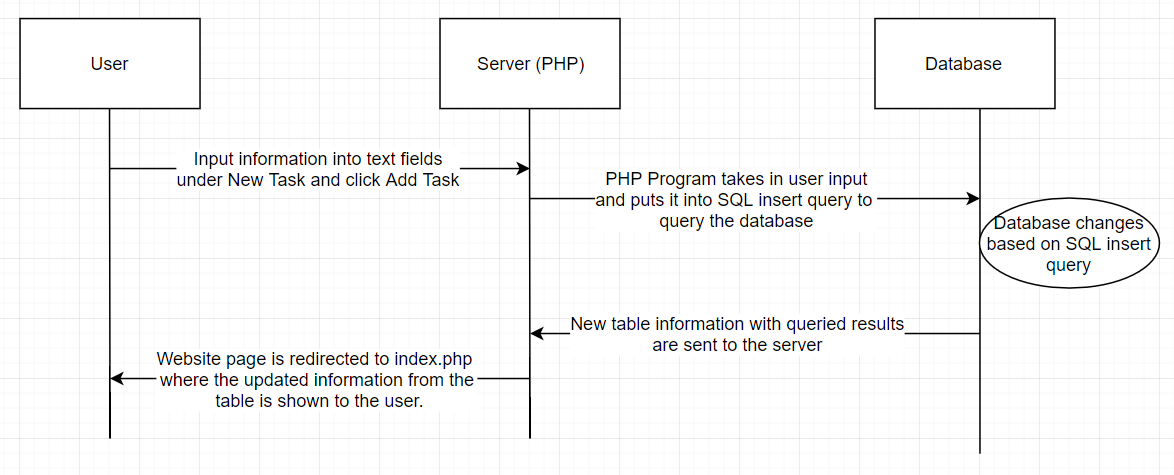
Description: User will undo the deletion of a task or tasks to the database.

Steps:

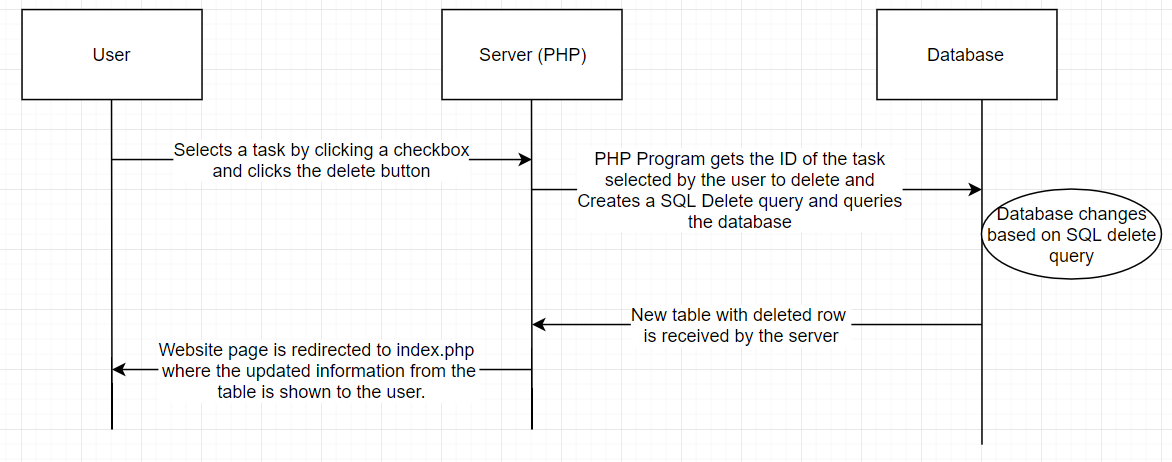
1. If you have just deleted a task, click the Undo Delete besides the delete button below the table
2. The page will redirect itself and the deleted rows will have been added back into the database and shown in the table above

**Sequence Diagrams**

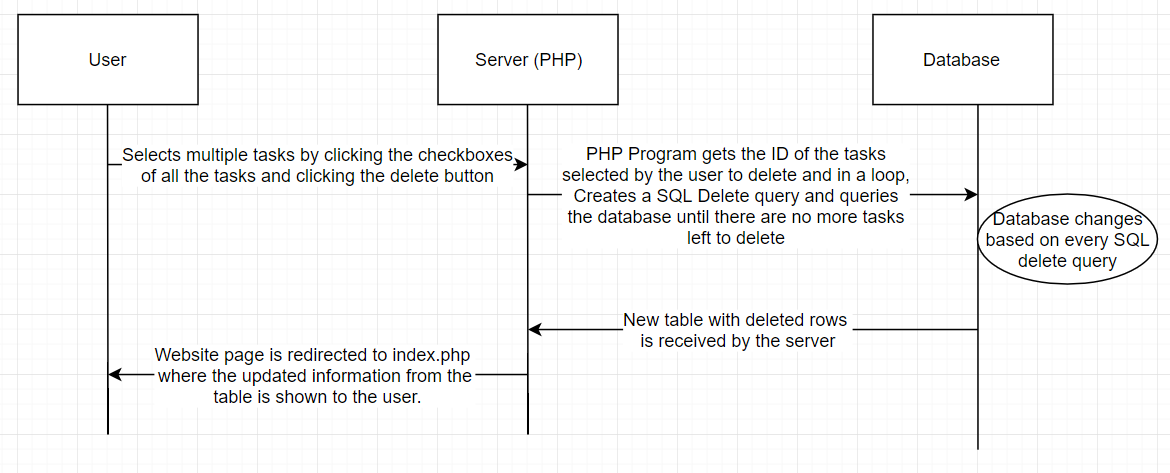
Sequence Diagram 1: Add Task



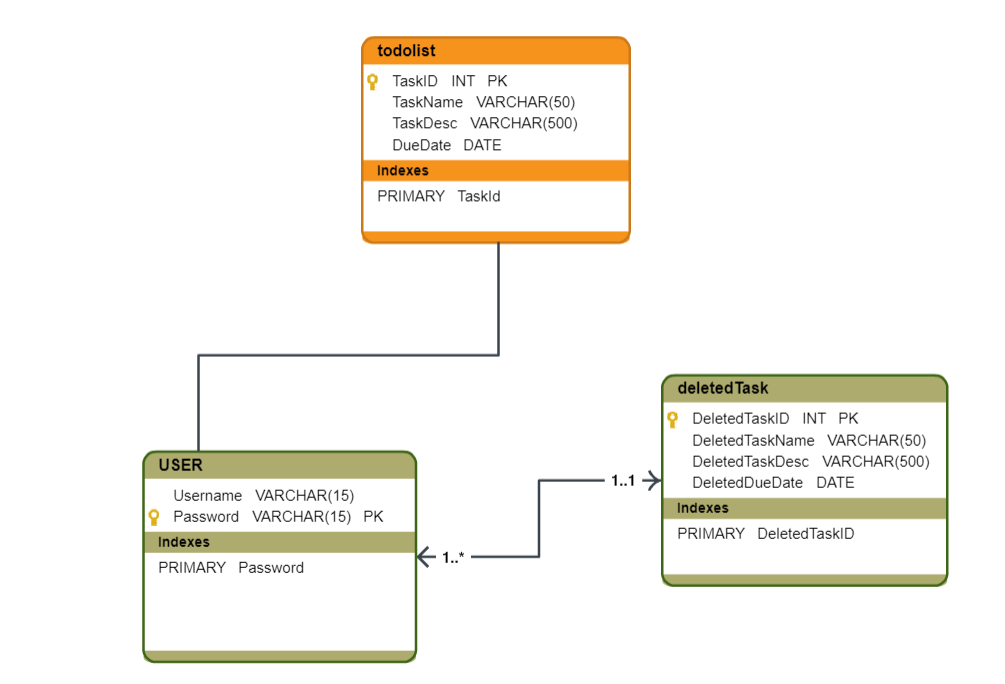
Sequence Diagram 2: Delete 1 Task



Sequence Diagram 3: Delete Multiple Tasks



**Database Design/ Class Diagram**

****

**Work breakdown**

Name of the task: Setting up WAMP

Estimated No of hours: 30 mins

Actual time spent: ~20 mins

Notes: Easy installation

Issues: N/A

Name of the task: PHP/Database Code

Estimated No of hours: 5 Hours

Actual time spent: 6-7 Hours

Notes: Connecting to database was simple, most time was taken on getting user input and using it to send queries to the database and making changes. Decided on adding delete buttons in the Remove Task column for each row but decided to change the design and add a checkbox and one single delete button below so it is more efficient to delete multiple rows from the table instead of a single row each time.

Issues: No big issues. I had to learn while I was doing it so I was going slow but now I feel like it would not take as long.

Name of the task: HTML/CSS Code

Estimated No of hours: 1-2 Hours

Actual time spent: 1.5 Hours

Notes: Time was mostly spent on whether I should

Issues: Had some problems trying to get the form to correctly output the delete button. During coding, it would show up beside every checkbox instead of just one at the bottom because I put the html form in the loop.

Name of the task: Documentation

Estimated No of hours: 2 Hours

Actual time spent: 4-5 Hours

Notes: Took much longer than I expected mostly because some of the terms I was not able to figure out exactly how I should create the diagrams.

Issues: While searching for these, there were many different variations of each diagrams like the Database design and class diagram were almost always similar so it was difficult to understand how to create them.

Name of the task: Video Demo

Estimated No of hours: 10 Minutes

Actual time spent: 10 Minutes

Notes: N/A

Issues: N/A