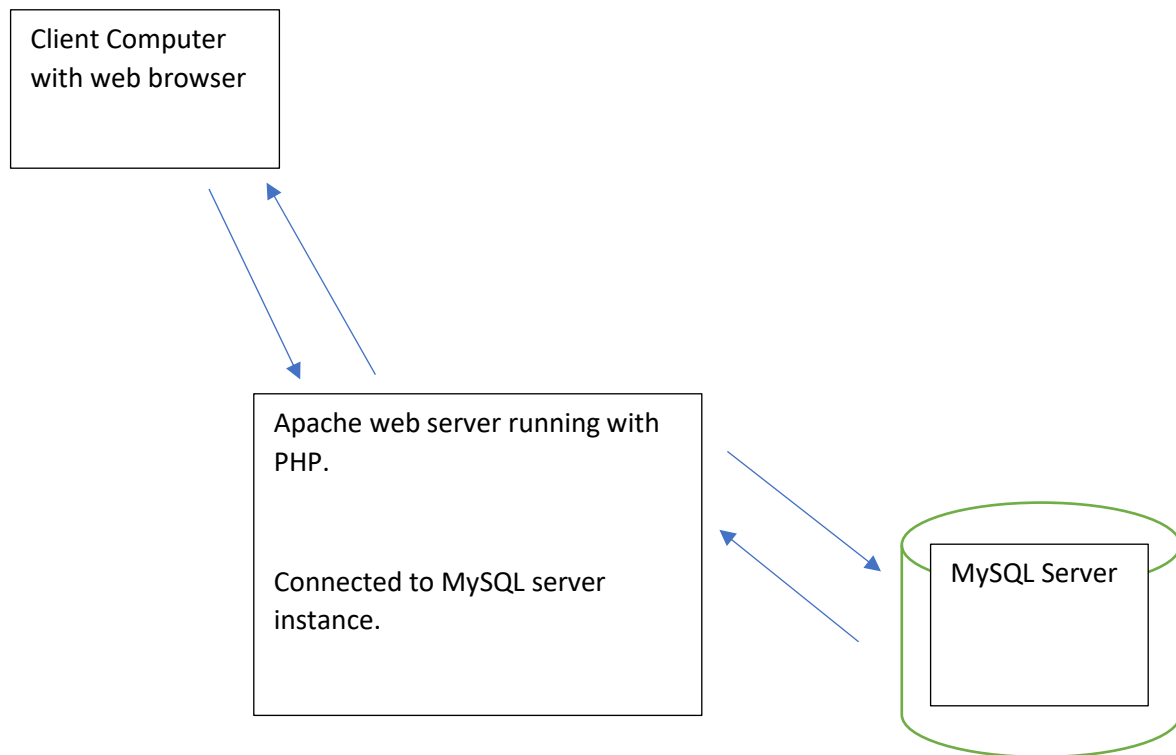Functional Requirements:

1. The system must store the to do tasks in a relational data base.
2. The system must support add and delete operations into a relational data base.
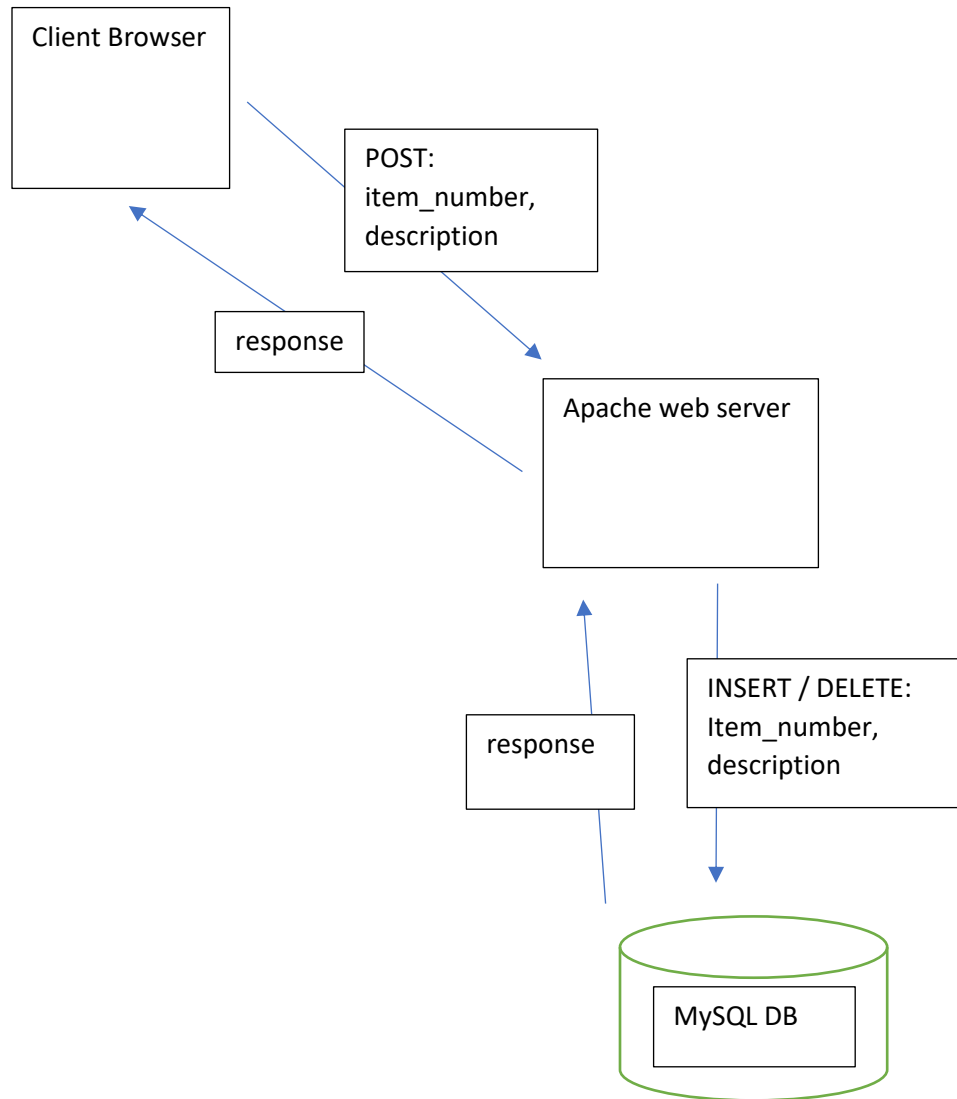3. The system must be web based.

Non-Functional Requirements:

1. The system must be built with WAMP tech stack.
2. The system must have a black and white color theme.
3. The system must have a button.

System Architecture:

Client Computer with web browser

Apache web server running with PHP.

Connected to MySQL server instance.

MySQL Server

Data Flow Diagram:

Client Browser

POST:
item_number,
description

response

Apache web server

response

INSERT / DELETE:
Item_number,
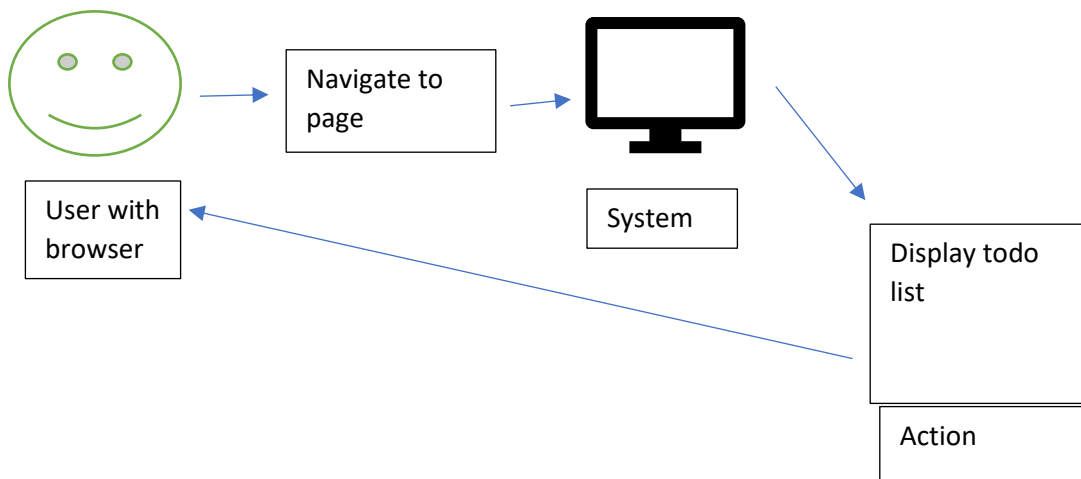description

MySQL DB

Use Cases:

As a user of the todo application I would like to see the items on the todo list.

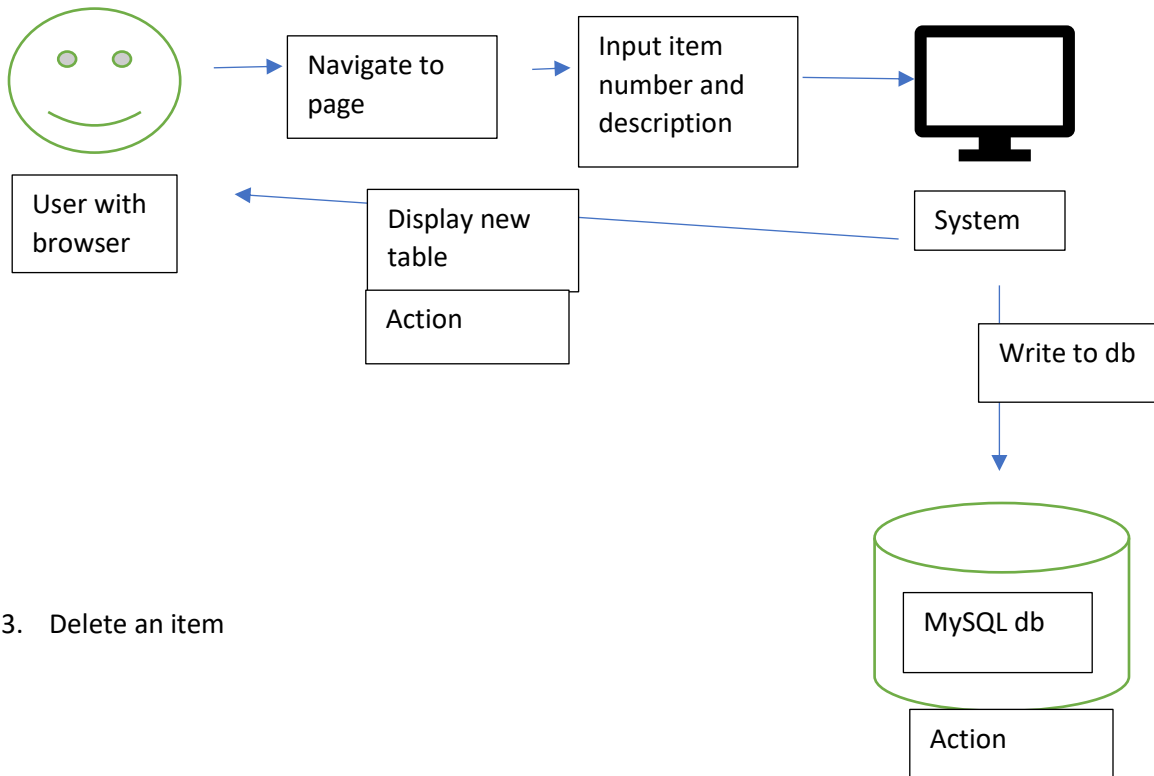As a user of the todo application I would like to be able to add a task to the list.

As a user of the todo application I would like to be able to remove tasks from the list.
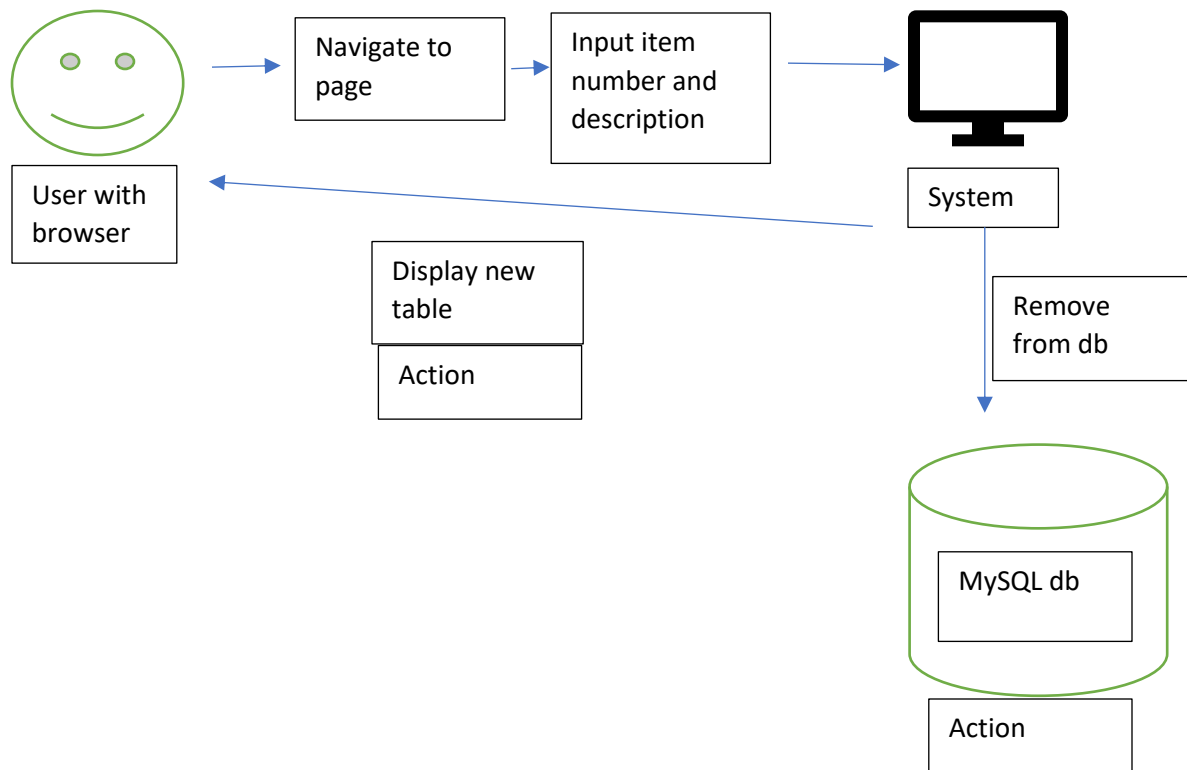
Sequence Diagram:

1. View List

| Navigate to page |
| User with browser |
| System |
| Display todo list |
| Action |

2. Add item

| Navigate to page |
| Input item number and description |
| System |
| Display new table |
| Action |
| User with browser |
| Write to db |
| MySQL db |

3. Delete an item

| Action |

User with browser

Navigate to page

Input item number and description

System

Display new table

Action

Remove from db

MySQL db

Action

DB Design:

Table

Primary Key	Field
item_number (INT)	description (TEXT)

Class Diagram:

No classes used

Test Cases:

When adding a todo item with the UI the info is stored in the DB

Cannot add a duplicate item number

When deleting a todo item with the UI the entry is removed from the DB.