To Do Application Documentation

[Functional Requirements](#vstdootvk08b) [2](#vstdootvk08b1)

[Non-Functional Requirements](#yswt30wqlzxi) [2](#yswt30wqlzxi1)

[System Architecture Diagram](#e1azc2pdg6ik) [4](#e1azc2pdg6ik1)

[Use Case](#on4hp5x1xd7y) [4](#on4hp5x1xd7y1)

[Sequence Diagram](#ci3tl09po97m) [5](#ci3tl09po97m1)

[Database Design](#epr8scvplspb) [6](#epr8scvplspb1)

[Class Diagram](#m7yyh4u9sms) [7](#m7yyh4u9sms1)

[Test Case](#pwdnva48mbh) [7](#pwdnva48mbh1)

[Code](#bookmark) [8](#bookmark1)

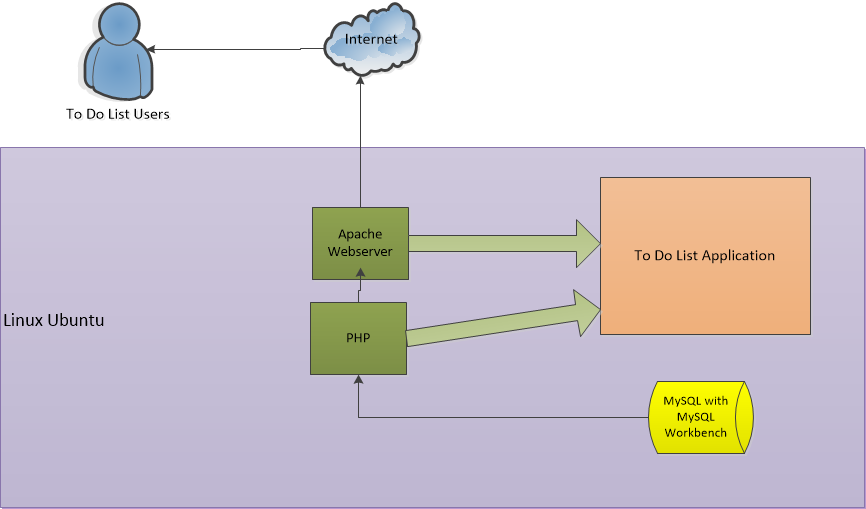
## Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement ID/Name** | **Requirement Statement** | **Required/Desired** | **Comments** |
| TD1/ Add Task | The website will have a button to add new task to the list | Required |  |
| TD2/View Task | The website will have the ability to display all task(s) | Required |  |
| TD3/Delete Task | The website will have the a button to delete a task when its is finished | Required |  |

## Non-Functional Requirements

|  |  |
| --- | --- |
| Service Attribute | |
| Non-Functional requirement NFR statement | Target Value |
| Is this a mission critical service? | Yes |
| What is the current service status? | In progress |
| What days in a week does the service need to be available? | Sun-Sat |
| Performance | |
| Non-Functional requirement NFR statement | Target Value |
| Average length of user session | 3 mins |
| Maximum number of unique users | 100 |
| Transaction Processing time | 1 second |
| Availability | |
| Non-Functional requirement NFR statement | Target Value |
| What geographies and time zones impact the system? | None |
| Are there different availability requirements for different geographies? | No |
| When should the system be available for its users | 99.999 |

## System Architecture Diagram



Made in Visio 2010

## Use Case

|  |  |  |
| --- | --- | --- |
| **Use case: Add task** | **Use Case: View task** | **Use Case: Delete task** |
| Actors: User | Actors: User | Actors: User |
| 1.User type out a task then hits the add button | 1.User task are viewable | 1. When user’s task are finished they can push the delete button |
| 2. Website displays task |  | 2.Website will delete the given task |
| Alternative:User wants to delete task | Alternative: | Alternative:User wants to add task |

## Sequence Diagram

## Snip20170119_1.png

Done with GenMyModel

## Database Design

**Users - Table**

idusers(AI PK) int(11)

Usernames varchar(45)

Password varchar(45)

**Items - Table**

Iditems(AI PK) int(11)

listitem text

**State - Table**

idState(PK) int(11)

Done varchar(45)

## Class Diagram

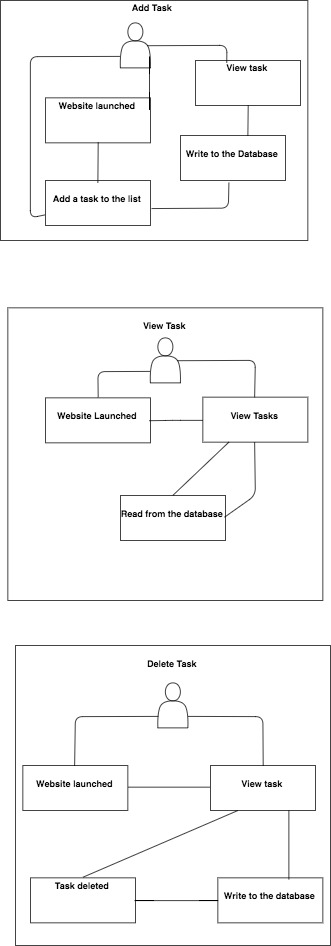
## Snip20170119_3.png

Done with Star UML

## Test Case

|  |  |
| --- | --- |
| **Project Name:** | |
| **Test Case Template** | |
| **Test Case ID: Add task** | **Test Designed by:** cc1771 |
| **Test Priority (Low/Medium/High):** High | **Test Designed date: 9**/4/17 |
| **Module Name:** To do list | **Test Executed by:** cc1771 |
| **Test Title:** Add task to list and database table | **Test Execution date: 9**/4/17 - 9/6/17 |
| **Description:** Test the add task feature |  |
|  |  |
|  |  |
| **Pre-conditions:** User has valid login | |
| **Dependencies:** | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Step** | **Test Steps** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Navigate to web site | localhost/folder/todoINDEX.html | To do list interface should display | User is navigated to | Pass |  |
| 2 | Does the text box flag the user if no text is enter | “pass Classes” | Please fill out field | Please fill out field | Pass |  |
| 3 | Enter text for task and it displays | “pass Classes” | Enter database and is displayed on page | text is displayed | Pass |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Data Flow Diagram

## Code

todoINDEX.html

<!DOCTYPE html>

<html>

<head>

<title>To Do List for CSC 4996</title>

</head>

<body>

<head> Insert your task </head>

<form action="to\_Do.php" method="post">

<input type="text" name="taskname" required placeholder="Title" />

<input type="text" name="taskdes" required placeholder="Description of task" />

<input id="submit" type="submit" name="add" value="Add">

</form>

<br>

<head> View Tasks List</head>

<form action="to\_Do.php" method="get">

<input id="submit" type="submit" name="view" value="Click to view your tasks">

</form>

<br>

<head> Delete Tasks</head>

<form action="to\_Do.php" method="post">

<input type="text" name="taskname" required placeholder="Which task # will be deleted" />

<input id="submit" type="submit" name="delete" value="Delete Task ">

</form>

</body>

</html>

to\_Do.php

<?php

require "http://localhost/todo\_app/mysqli\_connect.php"

if(isset($\_POST['listitem'])){

$listitem = trim($\_POST['listitem']);

if(!empty($listitem)) {

$addedQuery = $db->prepare("INSERT INTO items(listitem) VALUES (:listitem)");

$addedQuery->execute(['$listitem']);

}

}

header('Location: index.php');

?>

DEFINE ('DB\_USER', 'root');

DEFINE ('DB\_PASSWORD','senior');

DEFINE ('DB\_HOST','localhost');

DEFINE ('DB\_NAME','ToDo\_App');

$link = mysql\_connect(DB\_HOST, DB\_USER, DB\_PASSWORD);

if (!$link){

die('Could not connect to MySQL:'.mysql\_error());

}

$db\_selected = mysql\_select\_db(DB\_NAME, $link);

if (!$db\_selected){

die('Could not use'. DB\_NAME . ': '. mysql\_error());

}

echo'Connected';

$value = $\_POST['listitem'];

$sql = "INSERT INTO items (listitem) VALUES ('$value')";

if (!mysql\_query($sql)){

die('Error:'. mysql\_error());

}

mysql\_close();

?>