# ToDo Application

## Work Breakdown List

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Task | Est. Hours | Actual Hours | Notes | Issues |
|  | Install Tech Stack | 2 | 4.5 | Apache, PHP, MySQL, Git. Each installed separate for security & updatability. | Apache config. Problematic (64 bit Apache & 32 bit PHP is a non-starter), 2.5 hours lost. |
|  | Learn PHP / MySQL syntax | 6 | 7.5 | New language – review 3 examples | Unexpected trailing “\n” on text database fields caused problems matching rows in queries (i.e., “…WHERE name = ‘Greg’” should be =’Greg\n’) |
|  | Functional Requirements | 0.5 | 0.25 |  |  |
|  | Non-Functional Requirements | 0.5 | 0.25 |  |  |
|  | System Architecture | 0.5 |  |  |  |
|  | Data Flow Diagram | 0.5 |  |  |  |
|  | Use Cases | 0.5 |  |  |  |
|  | Sequence Diagrams | 0.5 |  |  |  |
|  | Database Diagram | 0.5 |  |  |  |
|  | Class Diagram | 0.5 |  |  |  |
|  | Test Cases | 0.5 |  |  |  |
|  | Database creation | 1.5 | 2.0 |  | Auto-incrementing index values in mySQL |
|  | PHP Coding | 5 | 6.5 | New language usage |  |
|  | Testing & Rework | 4 |  | New language usage |  |
|  | Deployment Notes | 1 |  |  |  |
|  | Record video demo | 0.5 |  |  |  |
|  | **TOTAL** | 24.5 | 21.0 |  |  |

## Functional requirements (required/desired) – 3

1. The application shall allow the user to add tasks to a list of tasks
2. The application shall allow the user to display the list of tasks
3. The application shall allow the user to delete a task from the list

## Non-functional requirements – 3

1. The default view of the application will be the display of the current list
2. Adding a task will be performed by clicking an “Add Task” button on the task list that will show a new “Add Task” dialog
3. Deleting a task will be performed by clicking the “Delete” button that will be displayed after each task on the list
4. Deletions will be confirmed before the delete is performed

## System Architecture Diagram – one for the complete system

## Data Flow Diagram – one for the complete system (level 2)

## Use cases – 3 use cases

## Sequence Diagram – 3 sequence diagrams

## Database Design – one for the system, 3 tables

## Class diagram – one for the system, 3 classes

## Test Case – 3 operations to be tested

## Deployment Notes