**CHAPTER IV**

**MANAGERIAL PROCESS PLAN**

**4.1 Start-Up Plan**

**4.1.1 Estimates**

**Table 4.1 Estimates**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PERSONNEL** | **RATE/HR** | **# PERSONNEL** | **HOUR RENDERED** | **COST** |
| Project Manager | 500 | 1 | 315 | 157500 |
| Analyst | 400 | 1 | 306 | 122400 |
| Database Manager | 400 | 3 | 306 | 122400 |
| Programmer | 300 | 5 | 503 | 150900 |
| Graphic Designer | 200 | 1 | 660 | 132000 |
| Trainer | 100 | 1 | 4 | 400 |
| QA Analyst | 100 | 1 | 115 | 11500 |
| Documentarist | 100 | 1 | **22** | 2200 |
| **TOTAL** |  | **14** | **2231** | **699300** |

Table 4.1 specifies the estimated cost of the personnel, the estimated wages, headcount, hours, and cost in conducting the project. The estimation is based on analogy, rule of thumb, standard unit of size, cost mode, and historical data.

**4.1.2 Staffing**

**Table 4.2 Staffing**

|  |  |
| --- | --- |
| **PERSONNEL** | **# PERSONNEL** |
| Project Manager | 1 |
| Analyst | 1 |
| Database Manager | 3 |
| Programmer | 5 |
| Graphic Designer | 1 |
| Trainer | 1 |
| QA Analyst | 1 |
| Documentarist | 1 |
| **TOTAL** | **14** |

Table 4.2 specifies the number of staff required per position for the whole project. The estimation is based from the number of personnel needed per project phase and duration of personnel requirement. Data from 4.3 is the main basis for estimation.

**4.1.3 Resource Acquisition**

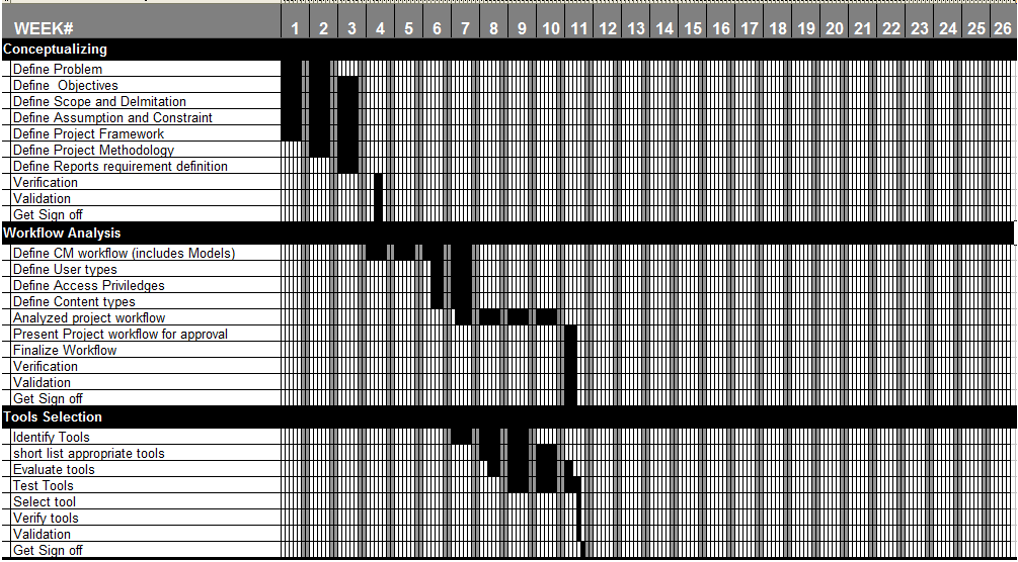
Since the system is a web-based system with PHP as its language, the proponent will utilize any Phpstorm as the integrated development environment. MySQL is used for the Entity Relationship Diagram, but the actual database will utilize Microsoft Access or MySQL Server. The resources have installers available in the market or in the internet. The server that the system will use once implemented will be provided by the Information and Technology Resource Office (ITRO) of Asia Pacific College.

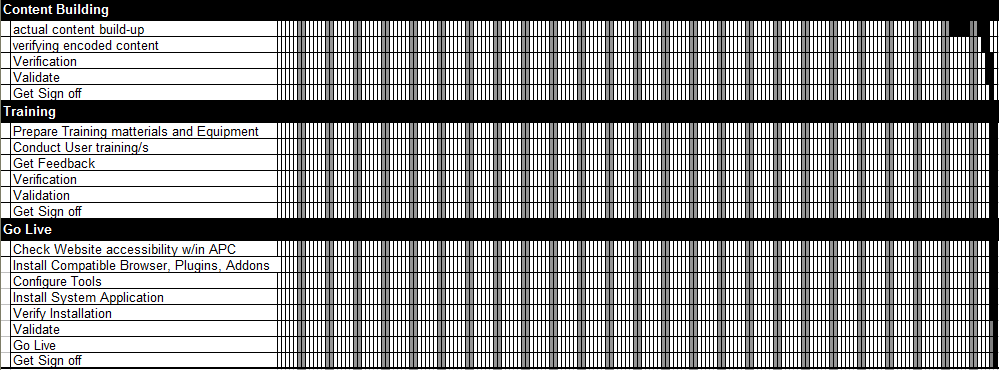
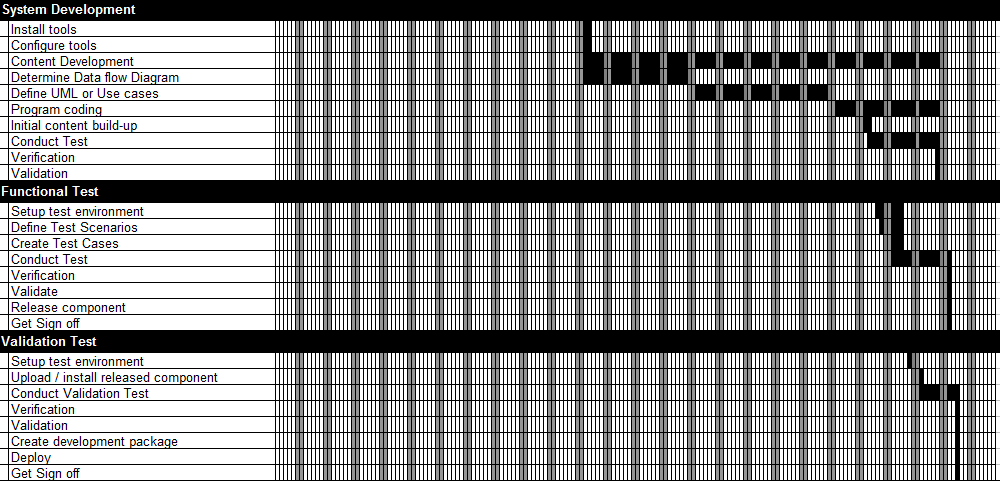
**4.2 Work Plan**

**4.2.1 Work Breakdown Structure**

Table 4.3 shows the Work Breakdown Structure in a graphical illustration of the phases of the project. The rows in the tables shows the major activities and its actual activities. The column is divided by weeks and a week is divided by days with corresponding date. If an activity is expected to start and end to date range, it will be highlighted on the table allowing the reader to easily manage the project and view the project status.

**Table 4.3 Work Breakdown Structure**





**4.6. Schedule Allocation**

**Table 4.4 Work Breakdown Structure per Major Activity**

|  |  |  |  |
| --- | --- | --- | --- |
| **MAJOR ACTIVITY** | **START DATE** | **END DATE** | **DAYS** |
| Conceptualizing | 15-Jan-18 | 26-Jan-18 | 12 |
| Workflow Analysis | 05-Feb-18 | 28-Mar-18 | 52 |
| Tools Selection | 26-Feb-18 | 30-Mar-18 | 33 |
| System Development | 02-Apr-18 | 01-Jul-18 | 91 |
| Functional Test | 14-Jun-18 | 02-Jul-18 | 19 |
| Validation Test | 22-Jun-18 | 04-Jul-18 | 13 |
| Content Building | 02-Jul-18 | 12-Jul-18 | 11 |
| Training | 12-Jul-18 | 12-Jul-18 | 1 |
| Go Live | 12-Jul-18 | 13-Jul-18 | 2 |

Table 4.4 shows the specified start and end estimated date per major activity and the expected days for completion.

**Table 4.5 Work Breakdown Structure per with Actual Activity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Conceptualizing** | **Start Date** | **End Date** | **Man Hours** |
| Define Problem | 15-Jan-18 | 01-Feb-18 | 12 |
| Define Project Objectives | 15-Jan-18 | 02-Feb-18 | 12 |
| Determine requirements | 15-Jan-18 | 02-Feb-18 | 12 |
| Define Business Process | 15-Jan-18 | 02-Feb-18 | 12 |
| Define Project Framework | 15-Jan-18 | 02-Feb-18 | 12 |
| Define Project Methodology | 19-Jan-18 | 02-Feb-18 | 24 |
| Define Reports requirement definition | 22-Jan-18 | 02-Feb-18 | 12 |
| Verification | 07-Feb-18 | 08-Feb-18 | 1 |
| Validation | 07-Feb-18 | 08-Feb-18 | 1 |
| get sign off | 07-Feb-18 | 08-Feb-18 | 1 |
| **Workflow Analysis** | **Start Date** | **End Date** | **Man Hours** |
| Define CM workflow | 05-Feb-18 | 02-Mar-18 | 96 |
| Define User types | 21-Feb-18 | 02-Mar-18 | 12 |
| Define Access privileges | 22-Feb-18 | 02-Mar-18 | 18 |
| Define Content types | 23-Feb-18 | 02-Mar-18 | 18 |
| Analysed project workflow | 27-Feb-18 | 23-Feb-18 | 138 |
| Present Project workflow for approval | 26-Mar-18 | 28-Mar-18 | 3 |
| Finalize Workflow | 26-Mar-18 | 28-Mar-18 | 15 |
| Verification | 26-Mar-18 | 28-Mar-18 | 8 |
| Validation | 26-Mar-18 | 28-Mar-18 | 8 |
| get sign off | 26-Mar-18 | 28-Mar-18 | 1 |
| **Tools Selection** | **Start Date** | **End Date** | **Man Hours** |
| Identify Tools | 26-Feb-18 | 16-Mar-18 | 81 |
| short list appropriate tools | 05-Mar-18 | 23-Mar-18 | 81 |
| Evaluate tools | 07-Mar-18 | 27-Mar-18 | 81 |
| Test Tools | 12-Mar-18 | 29-Mar-18 | 72 |
| Select tool | 29-Mar-18 | 29-Mar-18 | 3 |
| Verify tools | 29-Mar-18 | 29-Mar-18 | 3 |
| Validation | 29-Mar-18 | 29-Mar-18 | 3 |
| get sign off | 30-Mar-18 | 30-Mar-18 | 1 |
| **System Development** | **Start Date** | **End Date** | **Man Hours** |
| Install tools | 02-Apr-18 | 03-Apr-18 | 7 |
| Configure tools | 02-Apr-18 | 03-Apr-18 | 7 |
| Content Development | 02-Apr-18 | 29-Jun-18 | 669 |
| Determine Data Flow Diagram | 02-Apr-18 | 27-Apr-18 | 162 |
| Define UML or Use cases | 30-Apr-18 | 08-Nov-08 | 210 |
| Program coding | 02-Jun-18 | 29-Jun-18 | 159 |
| Initial content build-up | 11-Jun-18 | 12-Jun-18 | 4 |
| Conduct Test | 12-Jun-18 | 29-Jun-18 | 46 |
| Verification | 29-Jun-18 | 29-Jun-18 | 3 |
| Validation | 29-Jun-18 | 29-Jun-18 | 1 |
| **Functional Test** | **Start Date** | **End Date** | **Man Hours** |
| Setup test environment | 14-Jun-18 | 20-Jun-18 | 27 |
| Define Test Scenarios | 15-Jun-18 | 20-Jun-18 | 25 |
| Create Test Cases | 18-Jun-18 | 20-Jun-18 | 20 |
| Conduct Test | 18-Jun-18 | 02-Jul-18 | 38 |
| Verification | 02-Jul-18 | 02-Jul-18 | 3 |
| Validate | 02-Jul-18 | 02-Jul-18 | 1 |
| Release component | 02-Jul-18 | 02-Jul-18 | 1 |
| Get Sign off | 02-Jul-18 | 02-Jul-18 | 1 |
| **Validation Test** | **Start Date** | **End Date** | **Man Hours** |
| Setup test environment | 22-Jun | 22-Jun | 4 |
| Upload / install released component | 26-Jun | 26-Jun | 4 |
| Conduct Validation Test | 26-Jun | 04-Jul | 18 |
| Verification | 04-Jul | 04-Jul | 3 |
| Validation | 04-Jul | 04-Jul | 1 |
| Create development package | 04-Jul | 04-Jul | 6 |
| Deploy | 04-Jul | 04-Jul | 1 |
| Get Sign off | 04-Jul | 04-Jul | 1 |
| **Content Building** | **Start Date** | **End Date** | **Man Hours** |
| Actual content build-up | 03-Jul-18 | 11-Jul-18 | 18 |
| Verifying encoded content | 10-Jul-18 | 11-Jul-18 | 3 |
| Verification | 11-Jul-18 | 12-Jul-18 | 3 |
| Validate | 11-Jul-18 | 12-Jul-18 | 3 |
| Get sign off | 13-Jul-18 | 13-Jul-18 | 1 |
| **Training** | **Start Date** | **End Date** | **Man Hours** |
| Prepare Training materials and Equipment | 12-Jul-18 | 13-Jul-18 | 8 |
| Conduct User training/s | 12-Jul-18 | 13-Jul-18 | 3 |
| Get Feedback | 12-Jul-18 | 13-Jul-18 | 3 |
| Verification | 12-Jul-18 | 13-Jul-18 | 3 |
| Validation | 12-Jul-18 | 13-Jul-18 | 3 |
| Get sign off | 12-Jul-18 | 13-Jul-18 | 1 |
| **Go Live** | **Start Date** | **End Date** | **Man Hours** |
| Check Website accessibility w/in APC | 12-Jul-18 | 13-Jul-18 | 4 |
| Install Compatible Browser, Plugins, Addons | 12-Jul-18 | 13-Jul-18 | 2 |
| Configure Tools | 12-Jul-18 | 13-Jul-18 | 2 |
| Install System Application | 12-Jul-18 | 13-Jul-18 | 2 |
| Verify Installation | 12-Jul-18 | 13-Jul-18 | 4 |
| Validate | 12-Jul-18 | 13-Jul-18 | 3 |
| Go Live | 13-Jul-18 | 13-Jul-18 | 1 |
| Get Sign off | 13-Jul-18 | 13-Jul-18 | 1 |

Table 4.5 shows the specified start and end estimated date of completion per actual activity. The actual activity is grouped by major activity.

**Table 4.6. Resource Allocation**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | **Project Manager** | **Analyst** | **Database Manager** | **Programmer** | **Graphic Designer** | **Trainer** | **QA Analyst** | **Documentarist** |
|
|
|
|  | | **Manpower** | | | | | | | |
| **Conceptualizing** | | **PM** | **AN** | **DM** | **PR** | **GD** | **TR** | **QA** | **DO** |
|  | Define Problem | 1 |  |  |  |  |  |  |  |
|  | Define Project Objectives | 1 |  |  |  |  |  |  |  |
|  | Determine requirements | 1 |  |  |  |  |  |  |  |
|  | Define Business Process | 1 |  |  |  |  |  |  |  |
|  | Define Project Framework | 1 |  |  |  |  |  |  |  |
|  | Define Project Methodology | 1 |  |  |  |  |  |  |  |
|  | Define Reports requirement definition | 1 |  |  |  |  |  |  |  |
|  | Verification | 1 |  |  |  |  |  |  |  |
|  | Validation | 1 |  |  |  |  |  |  |  |
|  | Get Sign off | 1 |  |  |  |  |  |  |  |
| **Workflow Analysis** | | **PM** | **AN** | **DM** | **PR** | **GD** | **TR** | **QA** | **DO** |
|  | Define CM workflow | 1 | 1 | 1 |  |  |  |  |  |
|  | Define User types | 1 | 1 | 1 |  |  |  |  |  |
|  | Define Access privileges | 1 | 1 | 1 |  |  |  |  |  |
|  | Define Content types | 1 | 1 | 1 |  |  |  |  |  |
|  | Analysed project workflow | 1 | 1 | 1 |  |  |  |  |  |
|  | Present Project workflow for approval | 1 |  |  |  |  |  |  |  |
|  | Finalize Workflow | 1 | 1 | 1 |  |  |  |  |  |
|  | Verification | 1 | 1 | 1 |  |  |  |  |  |
|  | Validation | 1 | 1 | 1 |  |  |  |  |  |
|  | Get Sign off | 1 |  |  |  |  |  |  |  |
| **Tools Selection** | | **PM** | **AN** | **DM** | **PR** | **GD** | **TR** | **QA** | **DO** |
|  | Identify Tools | 1 | 1 | 1 |  |  |  |  |  |
|  | short list appropriate tools | 1 | 1 | 1 |  |  |  |  |  |
|  | Evaluate tools | 1 | 1 | 1 |  |  |  |  |  |
|  | Test Tools | 1 | 1 | 1 |  |  |  |  |  |
|  | Select tool | 1 | 1 | 1 |  |  |  |  |  |
|  | Verify tools | 1 | 1 | 1 |  |  |  |  |  |
|  | Validation | 1 | 1 | 1 |  |  |  |  |  |
|  | Get Sign off | 1 |  |  |  |  |  |  |  |
| **System Development** | | **PM** | **AN** | **DM** | **PR** | **GD** | **TR** | **QA** | **DO** |
|  | Install tools | 1 | 1 | 1 | 1 |  |  |  |  |
|  | Configure tools | 1 | 1 | 1 | 1 |  |  |  |  |
|  | Content Development | 1 |  |  |  | 5 |  |  |  |
|  | Determine Data flow Diagram | 1 | 1 | 1 | 3 |  |  |  |  |
|  | Define UML or Use cases | 1 | 1 | 1 | 3 |  |  |  |  |
|  | Program coding | 1 | 1 | 1 | 3 |  |  |  |  |
|  | Initial content build-up |  |  |  |  |  |  |  | 1 |
|  | Conduct Test | 1 | 1 | 1 | 1 |  |  | 1 |  |
|  | Verification | 1 | 1 | 1 |  |  |  |  |  |
|  | Validation | 1 |  |  |  |  |  |  |  |
| **Functional Test** | | **PM** | **AN** | **DM** | **PR** | **GD** | **TR** | **QA** | **DO** |
|  | Setup test environment | 1 | 1 | 1 |  |  |  | 1 |  |
|  | Define Test Scenarios | 1 | 1 | 1 |  |  |  | 1 |  |
|  | Create Test Cases | 1 | 1 | 1 |  |  |  | 1 |  |
|  | Conduct Test | 1 | 1 | 1 |  |  |  | 1 |  |
|  | Verification | 1 | 1 | 1 |  |  |  |  |  |
|  | Validate | 1 |  |  |  |  |  |  |  |
|  | Release component |  |  |  |  |  |  | 1 |  |
|  | Get Sign off | 1 |  |  |  |  |  |  |  |
| **Validation Test** | | **PM** | **AN** | **DM** | **PR** | **GD** | **TR** | **QA** | **DO** |
|  | Setup test environment |  |  |  |  |  |  | 1 |  |
|  | Upload / install released component |  |  |  |  |  |  | 1 |  |
|  | Conduct Validation Test |  |  |  |  |  |  | 1 |  |
|  | Verification | 1 | 1 | 1 |  |  |  |  |  |
|  | Validation | 1 |  |  |  |  |  |  |  |
|  | Create development package |  |  |  | 3 |  |  |  |  |
|  | Deploy |  |  |  | 1 |  |  |  |  |
|  | Get Sign off | 1 |  |  |  |  |  |  |  |
| **Content Building** | | **PM** | **AN** | **DM** | **PR** | **GD** | **TR** | **QA** | **DO** |
|  | actual content build-up |  |  |  |  |  |  |  | 1 |
|  | verifying encoded content | 1 | 1 | 1 |  |  |  |  |  |
|  | Verification | 1 | 1 | 1 |  |  |  |  |  |
|  | Validate | 1 | 1 | 1 |  |  |  |  |  |
|  | Get Sign off | 1 |  |  |  |  |  |  |  |
| **Training** | | **PM** | **AN** | **DM** | **PR** | **GD** | **TR** | **QA** | **DO** |
|  | Prepare Training matterials and Equipment |  |  |  | 3 |  | 1 |  |  |
|  | Conduct User training/s |  |  |  | 1 |  | 1 |  |  |
|  | Get Feedback | 1 | 1 | 1 |  |  |  |  |  |
|  | Verification | 1 | 1 | 1 |  |  |  |  |  |
|  | Validation | 1 | 1 | 1 |  |  |  |  |  |
|  | Get Sign off | 1 |  |  |  |  |  |  |  |
| **Go Live** | | **PM** | **AN** | **DM** | **PR** | **GD** | **TR** | **QA** | **DO** |
|  | Check Website accessibility w/in APC |  |  |  | 1 |  |  |  |  |
|  | Install Compatible Browser, Plugins, Addons |  |  |  | 1 |  |  |  |  |
|  | Configure Tools |  |  |  | 1 |  |  |  |  |
|  | Install System Application |  |  |  | 1 |  |  |  |  |
|  | Verify Installation | 1 | 1 | 1 | 1 |  |  |  |  |
|  | Validate | 1 | 1 | 1 |  |  |  |  |  |
|  | Go Live | 1 |  |  |  |  |  |  |  |
|  | Get Sign off | 1 |  |  |  |  |  |  |  |

**4.2.4. Budget Allocation**

**Table 4.7 Budget Allocation**

|  |  |  |
| --- | --- | --- |
| **Conceptualizing** | | **Cost** |
| Define Project Objectives | | **3,600** |
| Define Project Objectives | | **3,600** |
| Determine requirements | | **3,600** |
| Define Business Process | | **3,600** |
| Define Project Framework | | **3,600** |
| Define Project Methodology | | **7,200** |
| Define Reports requirement definition | | **3,600** |
| Verification | | **300** |
|  | Validation | **300** |
|  | Get Sign Off | **300** |
|  | **Total Conceptualizing** | **29,700** |
| **Workflow Analysis** | | **Cost** |
|  | Define CM workflow | **24,000** |
|  | Define User types | **3,000** |
|  | Define Access privileges | **4,500** |
|  | Define Content types | **4,500** |
|  | Analysed project workflow | **34,500** |
|  | Present Project workflow for approval | **900** |
|  | Finalize Workflow | **3,600** |
|  | Verification | **1,950** |
|  | Validation | **1,950** |
|  | Get sign Off | **300** |
|  | **Total Workflow Analysis** | **79,200** |
| **Tools Selection** | | **Cost** |
|  | Identify Tools | **18,900** |
|  | Short List Appropriate Tools | **18,900** |
|  | Evaluate tools | **18,900** |
|  | Test Tools | **16,800** |
|  | Select tool | **750** |
|  | Verify tools | **750** |
|  | Validation | **750** |
|  | Get Sign Off | **300** |
|  | **Total Selection** | **76,050** |
| **System Development** | | **Cost** |
|  | Install tools | **1,550** |
|  | Configure tools | **1,550** |
|  | Content Development | **101,700** |
|  | Determine Data Flow Diagram | **33,300** |
|  | Define UML or Use cases | **43,500** |
|  | Program coding | **32,850** |
|  | Initial content build-up | **400** |
|  | Conduct Test | **6,300** |
|  | Verification | **750** |
|  | Validation | **300** |
|  | **Total System Development** | **222,200** |
| **Functional Test** | | **Cost** |
|  | Setup test environment | **4,800** |
|  | Define Test Scenarios | **4,750** |
|  | Create Test Cases | **3,800** |
|  | Conduct Test | **5,600** |
|  | Verification | **750** |
|  | Validate | **300** |
|  | Release component | **100** |
|  | Get Sign off | **300** |
|  | **Total Functional Test** | **20,400** |
| **Validation Test** | | **Cost** |
|  | Setup test environment | **400** |
|  | Upload / install released component | **400** |
|  | Conduct Validation Test | **1,800** |
|  | Verification | **750** |
|  | Validation | **300** |
|  | Create development package | **1,200** |
|  | Deploy | **200** |
|  | Get Sign off | **300** |
|  | **Total Validation Test** | **5,350** |
| **Content Building** | | **Cost** |
|  | actual content build-up | **1,800** |
|  | verifying encoded content | **750** |
|  | Verification | **750** |
|  | Validate | **750** |
|  | get sign off | **300** |
|  | **Total Content Building** | **4,350** |
| **Training** | | **Cost** |
|  | Prepare Training materials and Equipment | **1,400** |
|  | Conduct User training/s | **400** |
|  | Get Feedback | **750** |
|  | Verification | **750** |
|  | Validation | **750** |
|  | get sign off | **300** |
|  | **Total Training** | **4,350** |
| **Go Live** | | **Cost** |
|  | Check Website accessibility w/in APC | **800** |
|  | Install Compatible Browser, Plugins, Addons | **400** |
|  | Configure Tools | **400** |
|  | Install System Application | **400** |
|  | Verify Installation | **950** |
|  | Validate | **750** |
|  | Go Live | **300** |
|  | Get Sign off | **300** |
|  | **Total Go Live** | **4,300** |
|  | **GRAND TOTAL** | **445,900** |

**Table 4.8 Summary of Project Cost**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PERSONNEL** | **RATE/HR** | **# PERSONNEL** | **HOUR RENDERED** | **COST** |
| Project Manager | 500 | 1 | 315 | 157500 |
| Analyst | 400 | 1 | 306 | 122400 |
| Database Manager | 400 | 3 | 306 | 122400 |
| Programmer | 300 | 5 | 503 | 150900 |
| Graphic Designer | 200 | 1 | 660 | 132000 |
| Trainer | 100 | 1 | 4 | 400 |
| QA Analyst | 100 | 1 | 115 | 11500 |
| Documentarist | 100 | 1 | **22** | 2200 |
| **TOTAL** |  | **14** | **2231** | **699300** |

**Table 4.9 Staff Salary**

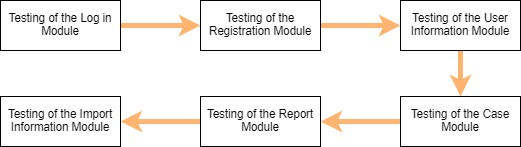
|  |  |  |
| --- | --- | --- |
| **Position** | **Position Code** | **Hourly rate** |
| Project Manager | **PM** | 300 |
| Analyst | **AN** | 250 |
| Database Manager | **DM** | 200 |
| Programmer | **PR** | 200 |
| Graphic Designer | **GD** | 150 |
| Trainer | **TR** | 100 |
| QA Analyst | **QA** | 100 |
| Documentarist | **DO** | 100 |

Table 4.9 Shows the staff position, the position code, and the corresponding hourly rate.

**4.3 Test Plan**

**4.3.1 Development Test Plan**

**4.3.1.1 Test Phases**



**Figure 4.1 Test Phases**

Figure 4.1 shows the major test phases. There will be 6 major phases in our Academic & Curricular Advising Module System. From the Login Module where the sign in, and password recovery are used. Next, we have the Registration Module where the Admin user creates new accounts. After that, we have the User Information Module which is one of the Admin's modules where account information can be updated. Another module which can be accessed by all users is the Case Module. It is a module where users can create, and view cases. Also, it is where the case notes are utilized. And we have the Reporting Module where only the Executive Director (XD) and Program Director (PD) has. It is a module where they can generate reports depending on how they want the attributes to be sorted and compiled the reports. Lastly, we have the Information Import Module where again, the XD and PD can import information from the databases: subjects, users(students), and cases.

**4.3.1.2 Test Cases and Test Scripts**

**Table 4.10 Test Cases and Test Scripts**

|  |  |
| --- | --- |
| **Test Phase 1:** | |
|  | **Description** |
| **Test Case 1.1-** Admin Login Test | This test is where the admin tests his/her login information. |
| **Test Case 1.2-** Student Login Test | This test is where the student test his/her login information. |
| **Test Case 1.3-** Adviser/Standard Faculty Login Test | This test is where the adviser or standard faculty test their login information. |
| **Test Case 1.4-** XD/PD Login Test | This test is where the XD or PD test their login information. |
| **Test Case 1.5-** Admin Login Failed Test (Wrong User ID) | This test is for the checking of the incorrect login popup in the admin login test. |
| **Test Case 1.6-** Student Login Failed Test (Wrong Password) | This test is for the checking of the popup for incorrect login using student information test account. |
| **Test Case 1.7-** Adviser/Standard Faculty Login Failed Test (Blank/Empty) | This test is for the adviser and or standard faculty incorrect login popup test. |
| **Test Case 1.8-** XD/PD Login Failed Test (Case Sensitivity) | This test is for the XD/PD incorrect login popup test. |
| **Test Phase 2:** | |
| **Test Case 2.1-** Add New Student User Test (1) | The first out of six steps in the user registration module for student users. The checking of the functionality of a quick link: Add New User |
| **Test Case 2.2-** Add New Student User Test (2) | The second out of six steps in the user registration module for student users. This is where the admin selects the account type as student and enters the new student ID. |
| **Test Case 2.3-** Add New Student User Test (3) | The third out of six steps in the user registration module for student users. Testing the button for submission. |
| **Test Case 2.4-** Add New Student User Test (4) | The fourth out of six steps in the user registration module for student users. This is where the admin enters the new student information. And the test of the button for submission. |
| **Test Case 2.5-** Add New Student User Test (5) | The fifth out of six steps in the user registration module for student users. Additional account information such as security questions and the user’s password. |
| **Test Case 2.6-** Add New Student User Test (6) | The last step out of six steps in the user registration module for student users. It tests the button for confirmation. |
| **Test Case 2.7-** Add New Faculty(Standard)/Adviser/XD/PD User Test (1) | The first out of six steps in the user registration module for Faculty(Standard)/Adviser/XD/PD users. The checking of the functionality of a quick link: Add New User |
| **Test Case 2.8-** Add New Faculty(Standard)/Adviser/XD/PD User Test (2) | The second out of six steps in the user registration module for Faculty(Standard)/Adviser/XD/PD users. This is where the admin selects the account type as Faculty(Standard)/Adviser/XD/PD and enters the new faculty ID. |
| **Test Case 2.9-** New Faculty(Standard)/Adviser/XD/PD User Test (3) | The third out of six steps in the user registration module for Faculty(Standard)/Adviser/XD/PD users. Testing the button for submission. |
| **Test Case 2.10-** Add New Faculty(Standard)/Adviser/XD/PD User Test (4) | The fourth out of six steps in the user registration module for Faculty(Standard)/Adviser/XD/PD users. This is where the admin enters the new faculty information. And the test of the button for submission. |
| **Test Case 2.11-** Add New Faculty(Standard)/Adviser/XD/PD User Test (5) | The fifth out of six steps in the user registration module for Faculty(Standard)/Adviser/XD/PD users. Additional account information such as security questions and the user’s password. |
| **Test Case 2.12-** Add New Faculty(Standard)/Adviser/XD/PD User Test (6) | The last step out of six steps in the user registration module for Faculty(Standard)/Adviser/XD/PD users. It tests the button for confirmation. |
| **Test Case 2.13-** Add New Student User Test: Case Sensitivity | This test is for the checking of the case sensitivity of the input of the user ID of a student user account. |
| **Test Case 2.14-** Add New Faculty(Standard)/Adviser/XD/PD User Test: Case Sensitivity | This test is for the checking of the case sensitivity of the input of the user ID of a Faculty(Standard)/Adviser/XD/PD user account. |
| **Test Case 2.15-** Add New Student User Test: Blank/Empty Input | This test is for the checking of the blank and empty fields in the user ID input upon entering the user ID. |
| **Test Case 2.16-** Add New Faculty(Standard)/Adviser/XD/PD User Test: Blank/Empty Input | This test is for checking the input if the fields are blank or empty. |
| **Test Case 2.17-** Invalid Student User Email Input Test (Wrong format) | This test is for the format of the email required and is checked in the algorithm of the program. A label message will popup once this test is enabled. |
| **Test Case 2.18-** Invalid Faculty(Standard)/Adviser/XD/PD Email Input Test (Does not exist) | This test is to check whether the email address exists in the email database. |
| **Test Case 2.19-** Blank input for Security Questions Test | This test is when the fields in the security questions are empty. A label message popup type will appear once the test is enabled. |
| **Test Phase 3:** | |
| **Test Case 3.1-** Update Password Test | This test is for the utilization of buttons in the update password quick link of the admin user. |
| **Test Case 3.2-** Update Security Questions and Password Test | This test is to see the transition of the security module of the admin page. Correct information is inputted in this test. |
| **Test Case 3.3-** Update Password Test: Mismatch | This test is when the admin incorrectly enters the password twice in the new password and confirm password textbox, an error input popup will be triggered once the button submit is clicked. |
| **Test Case 3.4-** Update Security Questions Test: Blank Input | This test is when the admin left a text empty in the new password and confirm password textbox, an error popup will be triggered once the button submit is clicked. |
| **Test Phase 4:** | |
| **Test Case 4.1-** Create a Case Test (Step 1) | This test is for the test of a tool button to launch the Case Module. |
| **Test Case 4.2-** Create a Case Test (Step 2) | This test is for the input of the new case information. |
| **Test Case 4.3-** Create a Case Test (Step 3) | This test is for the generation of a case number. |
| **Test Case 4.4-** Create a Case Test (Step 4) | This test is for the checking of the functionality of the Case Notes tool. |
| **Test Case 4.5-** Create a Case Test (Step 5) | This test is for the checking of the functionality of the popup of the Add Case button. |
| **Test Case 4.6-** Create a Case Test (Step 2): Blank Title | This test is for the checking for invalid input such as empty fields, etc. A label message will appear when the test is triggered. |
| **Test Case 4.7-** View Case by Case ID Test (Step 1) | This test is where the Case ID is checked for validity and existence. |
| **Test Case 4.8-** View Case by Case ID Test (Step 2) | This test is where the system is tested if it will generate a new Case ID. |
| **Test Case 4.9-** View Case by Case ID Test (Step 2): Case Sensitivity | This test is where the Case ID is checked for validity, format, and existence. |
| **Test Case 4.10-** View Case by Case ID Test (Step 2): Blank Input | This test is where the Case ID is checked for validity, format, and existence. |
| **Test Phase 5:** | |
| **Test Case 5.1-** View All Cases(Selected) Test | This test is for report generation depending on the specifications entered by the XD/PD. Specification: All Cases Per Faculty ID |
| **Test Case 5.2-** View All Cases Per Faculty ID Test | This test is for report generation depending on the specifications entered by the XD/PD. Specification: All Cases by Faculty ID |
| **Test Case 5.3-** View All Cases by Faculty ID Test | This test is for report generation depending on the specifications entered by the XD/PD. Specification: All Cases by Faculty ID |
| **Test Case 5.4-** View All Cases by Student ID Test | This test is for report generation depending on the specifications entered by the XD/PD. Specification: All Cases by Student ID |
| **Test Case 5.5-** View All Cases by Student ID (Can also be for Faculty ID): Case Sensitivity Test | This test is for case sensitivity test. |
| **Test Phase 6:** | |
| **Test Case 6.1-** View Case by Case ID Test (Step 1) | This is to test the importing of the cases depending on the specifications entered by a user. And exported via Generate Case button in the Report Module. |
| **Test Case 6.2-** View Case by Case ID Test (Step 2) | This test is the second step of the generation of cases for exporting. |
| **Test Case 6.3-** View Case by Case ID Test (Step 2): Case Sensitivity | This test is to check for case sensitivity in the input of the Case ID (special characters, etc.). The label message for invalid input is triggered. |
| **Test Case 6.4-** View Case by Case ID Test (Step 2): Blank Input | This test is to check for the label message if it will be triggered when not entering anything in the setting of the case specifications for export. |

**4.3.2 Functional Test Plan**

**Log in Module**

Test Case: Login

**Table 4.11 Test # 1 Test Scripts and Expected Output**

|  |  |
| --- | --- |
| **Test Script** | **Expected Output** |
| The user logs in his/her account using their User ID (School ID) and password. | The user will be directed to their designated homepage. |

Test Case: Forgot Password

**Table 4.12 Test # 2 Test Scripts and Expected Output**

|  |  |  |
| --- | --- | --- |
| **Test Instruction** | **Expected Output** | |
| The user clicks the “forgot password” button and answers the security questions | | The user will be notified with a popup that their security question answers are correct and will be redirected to the enter new password page. |

Test Case: Admin creating a new account

**Table 4.13 Test # 3 Test Scripts and Expected Output**

|  |  |
| --- | --- |
| **Test Instruction** | **Expected Output** |
| Admin clicks the “create user” button and fills it with new user information. | The admin will be notified with a popup that the account is successfully created. |

Test Case: Verifying the creation of a new account

**Table 4.14 Test # 4 Test Scripts and Expected Output**

|  |  |
| --- | --- |
| **Test Instruction** | **Expected Output** |
| Usually the admin, but any user can check whether the new account that was created will show in the account/users list. | The newly created account will be shown in the users list. |

**User Information Module**

Test Case: Updating a user’s information

**Table 4.15 Test # 1 Test Scripts and Expected Output**

|  |  |
| --- | --- |
| **Test Instruction** | **Expected Output** |
| The admin selects the user he/she will update and clicks the corresponding “update information” button in the admin menu. The admin will then input the new account information. | The admin will be notified with a popup that the account is successfully updated. |

Test Case: Verifying whether the information of the user was updated

**Table 4.16 Test # 2 Test Scripts and Expected Output**

|  |  |
| --- | --- |
| **Test Instruction** | **Expected Output** |
| The admin will select the user he/she updated and check whether the newly inputted information will properly show. | Once searched by the admin what he/she updated recently, the new and correct information will show in the corresponding user’s account. |

**Case Module**

Test Case: Creating a case

**Table 4.17 Test # 1 Test Scripts and Expected Output**

|  |  |
| --- | --- |
| **Test Instruction** | **Expected Output** |
| Any faculty member will click the “create a case” button and will enter the case information along with the student involved. | A popup will notify the faculty member that their case is successfully created. |

Test Case: Check if the case created and its information is the same with the corresponding information inputted in it

**Table 4.18 Test # 2 Test Scripts and Expected Output**

|  |  |
| --- | --- |
| **Test Instruction** | **Expected Output** |
| Any user, preferably the one who created the case, will view the newly created case and check whether the details one had enter from the test case of creating a case will be correct. He/she can check it by pressing the “view cases” and search for their corresponding cases. | The newly created case will show the correct information inputted by its creator. |

Test Case: Check if the case was created and if the people involved are notified

**Table 4.19 Test # 3 Test Scripts and Expected Output**

|  |  |
| --- | --- |
| **Test Instruction** | **Expected Output** |
| For the people involved in the newly created case, they must check if there is a notification that a case that was newly created with them involved will show. | The notification in their menu page will show because of the newly created case that they are involved with. |

Test Case: Viewing of the cases selected

**Table 4.20 Test # 4 Test Scripts and Expected Output**

|  |  |
| --- | --- |
| **Test Instruction** | **Expected Output** |
| When searching for a case, the user must enter the case details he/she would like to view from the “view case” button and the following criteria/attributes of the case will show, and the user will enter and select their specifications. | The cases with the corresponding attributes selected by the user will be gathered/appended and will be showed to the user who called upon the cases. |

Test Case: Adding Case Notes

**Table 4.21 Test # 5 Test Scripts and Expected Output**

|  |  |
| --- | --- |
| **Test Instruction** | **Expected Output** |
| In adding case notes, a user must first select the case they want to add case notes on by viewing the case first, then pressing the “add case notes” button for them to create a thread for the case notes. Then the user will enter the notes he/she would like to enter. | A popup will show that the case notes is successfully added to the case notes thread of that case. |

Test Case: Check if the case notes are properly created and people who are involved in the case are updated by the latest case notes created

**Table 4.22 Test # 6 Test Scripts and Expected Output**

|  |  |
| --- | --- |
| **Test Instruction** | **Expected Output** |
| After creating the newly added case notes, the users involved in the case will be notified that the case notes thread for the case is updated. | Notifications in the user involved will appear in their corresponding menus. |

**Report Module**

Test Case: Generating a report

**Table 4.23 Test # 1 Test Scripts and Expected Output**

|  |  |
| --- | --- |
| **Test Instruction** | **Expected Output** |
| This test is only for the X.D. and P.D. users. They must press the “generate a report” button in their menus and select the attributes that they want to generate in the report. | A popup will notify the X.D./P.D. that their report is successfully generated. |

Test Case: Check if the correct attributes called by the XD/PD are printed correctly

**Table 4.24 Test # 2 Test Scripts and Expected Output**

|  |  |
| --- | --- |
| **Test Instruction** | **Expected Output** |
| The X.D./P.D. can view their newly created reports in an exported file(document/pdf/etc.) | The exported file after the generation of the report has the correct information entered in it. |