

**Computer Science and Engineering**

**Integrated University Department Information System**

**Software Project Management Plan**

**Version 1.0**

Document Number: SPMP-001

Project Team Number: **A6**

Project Team Members: **Munieshwar Ramdass (mr3420)**

**Ajay Shenoy (as7195)**

**August Tan (aot221)**

**Review and Approvals**

|  |  |  |  |
| --- | --- | --- | --- |
| **Printed Name and Title** | **Function** | **Date** | **Signature** |
| Professor Strauss | Advisor | November 9, 2014 | On File |
| Munieshwar Ramdass | Author, Reviewer | November 9, 2014 | On File |
| Ajay Shenoy | Author, Reviewer | November 9, 2014 | On File |
| August Tan | Author | November 9, 2014 | On File |

**Revision Level**

|  |  |  |
| --- | --- | --- |
| **Date** | **Revision Number** | **Purpose** |
| November 9, 2014 | Version 1.0 | Initial Release |

**Table of Contents**

1. **OVERVIEW ………………………………………………………………………...……… 6**
   1. PROJECT SUMMARY …………………………………………………………...……. 6
   2. PURPOSE, SCOPE, AND OBJECTIVES …………………………………………...… 6
   3. ASSUMPTIONS AND CONSTRAINTS …………………………………………….... 6
   4. PROJECT DELIVERABLES ………………………………………………………...… 7
   5. SCHEDULE AND BUDGET SUMMARY …………………………………………..... 7
   6. EVOLUTION OF THE PLAN ………………………………………………………..... 7
2. **REFERENCES …………………………………………………………………………...… 7**
3. **DEFINTIONS ………………………………………………………………………………. 7**
4. **PROJECT ORGANIZATIONS …………………………………………………………... 7**
   1. EXTERNAL INTERFACES ………………………………………………………….... 8
   2. INTERNAL STRUCTURE …………………………………………………………..… 8
   3. ROLES AND RESPONSIBILITIES ………………………………………………….... 8
5. **MANAGEMENT PROCESSES …………………………………………………………... 8**
   1. START-UP PLAN ……………………………………………………………………… 8
      1. *Estimation Plan ………………………………………………………………………….. 8*
      2. *Staffing Plan ……………………………………………………………………………... 8*
      3. *Resource Acquisition Plan ……………………………………………………………... 8*
      4. *Training Plan ………………………………………………………………………….…. 9*
   2. WORK PLAN ……………………………………………………………………….….. 9
      1. *Work Activities ……………………………………………………………………….….. 9*
      2. *Schedule Allocation ………………………………………………………………….….. 9*
      3. *Resource Allocation …………………………………………………………………….. 9*
      4. *Budget Allocation ……………………………………………………………………..… 9*
   3. CONTROL PLAN ……………………………………………………………………… 9
      1. *Requirement Control and Traceability ……………………………………………… 10*
      2. *Schedule Tracking and Adjustment ………………………………………………….. 10*
      3. *Budget Tracking and Adjustment …………………………………………………….. 10*
      4. *Quality Control ………………………………………………………………………… 10*
      5. *Reporting Mechanisms ………………………………………………………………… 10*
      6. *Metrics Collection Plan ……………………………………………………………….. 10*
   4. RISK MANAGEMENT ………………………………………………………………. 11
   5. POST IMPLEMENTATION PLAN ………………………………………………….. 11
6. **TECHNICAL PROCESSES …………………………………………………………...… 11**
   1. PROCESS MODEL …………………………………………………………………… 11
   2. METHODS, TOOLS, AND TECHNIQUES …………………………………………. 12
   3. INFRASTRUCTURE PLAN ………………………………………………………….. 12
   4. PRODUCT ACCEPTANCE AND MIGRATION PLAN ……………………………. 12
7. **SUPPORTING PROCESSES PLAN …………………………………………………… 12**
   1. CONFIGURING MANAGEMENT PLAN …………………………………………... 13
   2. QUALIFICATION (VERIFICATION AND VALIDATION) PLAN ……………….. 13
   3. DOCUMENTATION (LIBRARY) PLAN …………………………………………… 13
   4. QUALITY ASSURANCE PLAN …………………………………………………….. 13
   5. REVIEW AND AUDITS ……………………………………………………………... 14
   6. PROBLEM RESOLUTION PLANS …………………………………………………. 14
   7. ENVIRONMENT MANAGEMENT PLANS ………………………………………... 14
   8. PROCESS IMPROVEMENT PLAN …………………………………………………. 14
8. **ADDITIONAL PLANS …………………………………………………………………... 14**
9. **INDEX …………………………………………………………………………………….. 14**
10. **RATIONALE ……………………………………………………………………………... 14**
11. **NOTES …………………………………………………………………………………….. 14**
12. **APPENDICES …………………………………………………………………………….. 14**
    1. SCHEDULE TRACKING ……………………………………………………... 15
    2. DEFECT TRACKING …………………………………………………………. 15
    3. GANTT CHART/MICROSOFT PROJECT SCHEDULE …………………….. 16

**1. OVERVIEW**

**1.1. PROJECT SUMMARY**

The Integrated University Departmental Information System is a smaller part of the larger university system that will interact with the budget system and the database system for the computer science department. The IUDIS is concerned with the users of the IUDIS, such as students, professors, advisors, other officials, and third party goods/service providers. The goal of the IUDIS is to please the user by having fast loading, easy to read, clearly arranged, and seamless navigation to all pages. Other factors that may affect the system is the load, the hard disk space, and the amount of down time the server faces. This product may be appear to be a combination of NYU’s Albert and Bursar systems.

**1.2. PURPOSE, SCOPE, AND OBJECTIVES**

The Integrated University Departmental Information System is an organized approach intended to help universities regulate student records, laboratory administration, the ordering of goods and services, and the management of account receivable and payable. It should be linked to a wider University system responsible for staff salaries, student transcripts, etc. The IUDIS structure provides a new information system that is faster with a slicker user interface then the outdated and archaic systems, which makes it easier for students to access their records. Simultaneously, it will be effortless for staff and faculty to use the system for submitting grades and financial information. The new user interface allows for users to interact with the system seamlessly while being aesthetically pleasing, accommodating a broader spectrum of users. The structure of the information system will be rigorously secure to protect the confidentiality, integrity and availability of the individual user.

Priority is placed on this project since the integration of the departmental budget of the computer science and engineering department is necessary for the account of the university’s budget itself.

**1.3. ASSUMPTIONS AND CONSTRAINTS**

The project will assume that a larger University information system already exists. The project will also assume that the University information system has its own functional budget system. Since hardware is not provided by the team, the team will assume that whatever hardware is used will be compatible to the IUDIS.

Deadline must be met. Budget must be used efficiently. The IUDIS must be user-friendly and reliable. Architecture must be able to conform to new features that may appear in the future.

**1.4. PROJECT DELIVERABLES**

Deliverables: Due Date:

Project Proposal 10/7/2014

Software Requirements Specification (SRS) 10/13/2014

Software Project Management Plan (SPMP) 11/15/2014

Software Analysis Specification (SAS) 11/19/2014

Software Design Document (SDD) 12/1/2014

**1.5. SCHEDULE AND BUDGET SUMMARY**

Refer to the Gantt chart in section 12.3.

**1.6. EVOLUTION OF THE PLAN**

This document may be altered in the future. Any changes to the SPMP will result in a revision document with its revision summary and number marked on the Revision Table on page 3 of the document. Specific changes to sections will be indicated along with its revision number after team approval of that change.

**2. REFERENCES**

Integrated University Department Information System, Team A6, Project Proposal Version 1.1

Integrated University Department Information System, Team A6, Software Requirements Specifications Version 1.1

**3. DEFINITIONS**

IUDIS – Integrated University Department Information System

SQA – Software Quality Assurance [Group]

PM – Project Manager

**4. PROJECT ORGANIZATION**

**4.1. EXTERNAL INTERFACES**

The SPMP will be processed by the SQA. This is done to make sure that this artifact is up to standards. The client will also receive a copy of this artifact. S/he will inspect it to make sure it addresses the appropriate specifications and approve it if fit. The client will also be aware of the project deliverables and other scheduling.

**4.2. INTERNAL STRUCTURE**

The team is a democratic one. The only leader present would be the advisor. Each member will do self and peer reviews to make sure quality standards are met. Each member will take equal responsibility for all aspects of this project. Walkthroughs and inspection will be done by the SQA. This is for thorough fault results. The SQA will also do a review of on all artifacts.

**4.3. ROLES AND RESPONSIBILITIES**

|  |  |  |  |
| --- | --- | --- | --- |
| **Role** | **Name** | **Email** | **Phone** |
| Advisor | Professor Strauss | fs817@nyu.edu |  |
| Documents Author, Coder, Tester, Reviewer | Munieshwar Ramdass | mr3420@nyu.edu |  |
| Documents Author, Coder, Tester, Reviewer | Ajay Shenoy | as7195@nyu.edu |  |
| Documents Author, Coder, Tester, Reviewer | August Tan | aot221@nyu.edu |  |

The advisor as indicated above will be the PM of this project. He will review all artifacts when it is due.

**5. MANAGEMENT PROCESSES**

**5.1. START-UP PLAN**

There are three members in team A6. Training for this project is required. Further details are in section 5.1.4.

**5.1.1. Estimation Plan**

**5.1.2. Staffing Plan**

**5.1.3. Resource Acquisition Plan**

**5.1.4. Training Plan**

All members must have knowledge on Databases. A course in Databases is necessary. All members must have intermediate coding skills in Java which can be attained by taking a course. All members must be familiar with Maven which can be attained by text and/or online resources. All members must have familiarity with web applications. This includes familiarity with HTML5/CSS3 which can be learned through text and/or online resources.

**5.2. WORK PLAN**

**5.2.1. Work Activities**

Refer to the Gantt chart in section 12.3. [Specifically tasks]

**5.2.2. Schedule Allocation**

Refer to the Gantt chart in section 12.3. [Specifically dates]

**5.2.3. Resource Allocation**

Refer to the Gantt chart in section 12.3. [Specifically person(s)]

**5.2.4. Budget Allocation**

**5.3. CONTROL PLAN**

Requirements will be monitored on a regular basis. To do so, team members will have to refer to the SRS. To monitor budget and schedule, metrics will be used.

**5.3.1. Requirement Control and Traceability**

All requirements can be traced to SRS section 7. The requirements for this system have been outlined by the software development team. The team will be visiting universities to see what their bursar and registrar systems are in need of. Then the documents will be outlined with as much of the university’s needs fulfilled while maintaining the system’s integrity of being fast and easy to use. The requirements document will then be presented to the university for them to review and we will work with them to include any features they would like into our system without compromising its efficiency. The system will be made in accordance to the priority of the requirements. With this in mind, a basic database will be made first along with its web interface. Once this is done then additional features such as user friendliness and system optimization will be made. These requirements will be periodically communicated to the universities that we are planning on selling this system to. A document of requirements that have been fulfilled and ones that are currently in progress will be sent with each version release. The interval in which this happens will increase or decrease depending on how on track the system is with what is desired by our buyers. This way the requirement control and its traceability will be documented every step of the way and will be effectively communicated between the developer and the client.

**5.3.2. Schedule Tracking and Adjustment**

Schedule tracking will be maintained by a Microsoft Project document as displayed in section 12.3 of this document. It will outline the progress that is intended to be made on a daily basis and what should be accomplished by each member. Each day at a specific time there will be a developer meeting to discuss the progress on the system. This progress will be measured against what was scheduled on the Microsoft Project document. If progress is behind the document’s schedule then work will be pushed onto days where there is enough time to accomplish it along with another goal. If there is no available time for a reschedule then the project’s release date will be pushed back to give the developers more time. If progress is ahead of schedule then there will be additional features added on to the project or the system will be pushed for an early release date. All deliverables are submitted to the advisor at its due date.

**5.3.3. Budget Tracking and Adjustment**

**5.3.4. Quality Control**

Quality is of utmost importance for this system and its reliability will be what separates it from existing systems. The SQA group will handle Quality Control. They will test using plan that they have developed and report to the development team with defects/faults that they have found.

**5.3.5. Reporting Mechanisms**

**5.3.6. Metrics Collection Plan**

Metrics will be collected when an artifact is completed during the development process to ensure that the team’s performance is optimal and any hitches within the system are caught. This can be seen in SRS section 14. There will be weekly meetings with the development team looking over faults in the documentation as well as discussion as on how progress is faring in accordance to the schedule. Scheduling time is an estimation and is also monitored when each artifact is completed to give the team a better understanding of estimating for future artifacts.

**5.4. RISK MANAGEMENT**

Below are risks that are in consideration and their respective likelihood of risk, person responsible for monitoring risk, mitigation of the risk and status of risk.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risk** | **Probability** | **Responsible Person** | **Mitigation** | **Status** |
| Late Submission | 25% | Munieshwar Ramdass | Request extension from Advisor for deliverable | Ongoing |
| Product not built to Specifications | 15% | Ajay Shenoy, August Tan | Check all requirements and specifications | Ongoing |
| Illness of a Member | 20% | Partner of the ill | Use the Agile Process. Another member can take up the sick member’s work. | Ongoing |
| Illness of Partnership | 4% | Remaining Members | Other members must be informed on the work of the ill | Ongoing |
| Member fails/drops Course and Project | 1% | Remaining Members | Terminate project or get one more person to make a team of 3 | Ongoing |

**6. TECHNICAL PROCESSES**

**6.1. PROCESS MODEL**

The construction of the Integrated University Departmental Information System will follow the waterfall lifecycle model once the project is a certified by NYU Polytechnic School of Engineering Computer Science and Engineering department and approved by the advisor. During construction, members with exceptional coding skills will follow the object oriented programming. This includes data encapsulation along with information hiding and class inheritance to shape the initial design of the system’s software and then refine the software to the final product. Team member will work in rotating groups of two among three members on separate computers for maximum throughput. After construction, Engineers will test the reliability and construction of the software. The SQA will be conducting all reviews of all artifacts generated in the project. Since this project is a smaller part of the larger university system, the software has to go through many evolutionary stages within the first few months after its development to seamlessly integrate with the ever growing university system. Once the University is satisfied with the project, the Operation of the software will be handled by the staff of the University's CS department. The procedures for support will take place once the software has been delivered to University's CS department.

**6.2. METHODS, TOOLS, AND TECHNIQUES**

Currently, Microsoft Word 2013 is used for documentation. Microsoft Project 2010 is used for Gantt charts. Google Diagram was used for UML diagrams. Google Docs is most frequently used to synchronize team members’ work for all documentations. Netbeans IDE 8.0.1 will be used in the future for implementation (or Visual Studio 2013 as a secondary option). HTML5/CSS3 will be used in the future for design. SQL will be used in the future for the database. The IUDIS will be based on an open architecture using Windows NT 8.1, which allows for the use of object-oriented methods and tools for analysis, design, and implementation.

**6.3. INFRASTRUCTURE PLAN**

All team members are will work within the offices of designated by the university, having access to the center of operations. All team member will have access to an existing physical workspace, the software engineering lab, to work on documentation, coding, testing and group project discussions. The PM will have to sign a server admission request form, because the access to the server room is granted on an as-needed basis by the University and retracted when access is no longer required. Engineers will synchronization all data files and snapshots for every group member over Dropbox and all documentation on Google Drive, which allows for group members to share and exchange documents. Group members will communicate over emails, phone calls and face to face group meetings.

**6.4. PRODUCT ACCEPTANCE AND MIGRATION PLAN**

**7. SUPPORTING PROCESSES PAN**

**7.1. CONFIGURATION MANAGEMENT PLAN**

Each team member has a copy of all baseline artifacts stored on their hard disk. However, to synchronize all artifacts, Google Drive (specifically Google Docs) is used between all members. Final copies of all artifacts are submitted to NewClasses associated by the team number A6.

The requirements document will be consistently reviewed on a weekly basis. Whenever a new feature is being implemented, the requirements will be referred to in order to see if the feature is consistent with what our system needs. In turn, the requirements document will change in the event that it is decided by the client that a new feature is required.

**7.2. QUALIFICATION (VERIFICATION AND VALIDATION) PLAN**

There will be weekly meeting amongst the developers and the clients/client representatives. Work will be verified by tracing back to the SRS. Self-reviews will be done. This is only done by authors of each artifact. Peer reviews are also done similarly. Walkthroughs are done for faults. Inspections are done for detailed reasons and are documented.

**7.3. DOCUMENTATION (LIBRARY) PLAN**

All members are responsible for the documentation of the SRS, SPMP, SAS, and SDD and any other future documents. For coders, inline comments are required. All members are required to do self-review, peer reviews, walkthroughs and inspections. The advisor will provide the team with comments (based on the IEEE standards) for each document. Each member contribute to each deliverable. Requirements for each document can be traced since they will have a number associated with them. All documents will have metrics done by the team members for the benefit of the team members. Inspections are documented. Defects are reported and any actions taken to fix them. Baseline requirements are in this document section 1.4.

**7.4. QUALITY ASSURANCE PLAN**

SQA will be responsible for ensuring that these documents fulfill the system’s requirements. To do this, the SQA team will generate test cases for our system and run them through the documentation to see if they can be worked through. SQA team will continue to generate test cases for our system as well as review the documentation to ensure it fulfills the client’s needs.

**7.5. REVIEW AND AUDITS**

The review and auditing process will be handled by a joint developer/client team. They will be conducted whenever there is a document ready for submission. During each submission review every member of the team will conduct his or her own fault detection of the documentation. Once they have done this they will convene and have a discussion of the listed faults they agree on under a person leading the discussion while another person makes note of faults agreed on.

**7.6. PROBLEM RESOLUTION PLANS**

Once a fault has been detected there will be an analysis of how to fix it. This analysis includes where the fault is located in the documentation, the level of detail this fault entails, and how the correction of this fault will affect the rest of the documentation. Once these factors have been determined then the members most capable of fixing these faults will be assigned to them.

**7.7. ENVIRONMENT MANAGEMENT PLANS**

**7.8. PROCESS IMPROVEMENT PLAN**

**8. ADDITIONAL PLANS**

**9. INDEX**

**10. RATIONALE**

The rationale behind the Integrated University Department Information System is to allow a seamless integration between huge university systems and the department system. The computer science department may have many small and independent functions from the main university system and so it should prove to be easy if the department handles its budget by itself while the larger university system tallies the department’s budget into its own budget. Furthermore, students’ grades and registration should be handled likewise.

**11. NOTES**

**12. APPENDICES**

**12.1. SCHEDULE TRACKING**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Artifact or Deliverable** | **Who** | **Estimated** | **Actual** | **Difference** |
| Initial SRS | Munieshwar Ramdass | 5 | 6 | 1 |
| Initial SRS | Ajay Shenoy | 6 | 5.5 | 0.5 |
| Initial SRS | August Tan | 4 | 5 | 1 |
| Initial SRS | Summary | 15 | 16.5 | 1.5 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Artifact or Deliverable** | **Who** | **Estimated** | **Actual** | **Difference** |
| Final SRS | Munieshwar Ramdass | 3 | 3 | 0 |
| Final SRS | Ajay Shenoy | 4 | 3 | 1 |
| Final SRS | August Tan | 3 | 3 | 0 |
| Final SRS | Summary | 10 | 9 | 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Artifact or Deliverable** | **Who** | **Estimated** | **Actual** | **Difference** |
| SPMP | Munieshwar Ramdass | 5 | 5 | 0 |
| SPMP | Ajay Shenoy | 4 | 3 | 1 |
| SPMP | August Tan | 4 | 4 | 0 |
| SPMP | Summary | 13 | 12 | 1 |

**Cumulative**

|  |  |  |  |
| --- | --- | --- | --- |
| **Who** | **Estimated** | **Actual** | **Difference** |
| Munieshwar Ramdass | 13 | 14 | 1 |
| Ajay Shenoy | 14 | 11.5 | 2.5 |
| August Tan | 11 | 12 | 1 |
| Summary | 38 | 37.5 | 1.5 |

**12.2. DEFECT TRACKING**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Artifact or Deliverable** | **Who** | **Estimated** | **Actual** | **Difference** |
| Initial SRS | Munieshwar Ramdass | 16 | 10 | 6 |
| Initial SRS | Ajay Shenoy | 10 | 10 | 0 |
| Initial SRS | August Tan | 15 | 10 | 5 |
| Initial SRS | Summary | 41 | 30 | 11 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Artifact or Deliverable** | **Who** | **Estimated** | **Actual** | **Difference** |
| Final SRS | Munieshwar Ramdass | 5 | 5 | 0 |
| Final SRS | Ajay Shenoy | 2 | 5 | 3 |
| Final SRS | August Tan | 4 | 5 | 1 |
| Final SRS | Summary | 11 | 15 | 4 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Artifact or Deliverable** | **Who** | **Estimated** | **Actual** | **Difference** |
| SPMP | Munieshwar Ramdass | 4 |  |  |
| SPMP | Ajay Shenoy | 4 |  |  |
| SPMP | August Tan | 4 |  |  |
| SPMP | Summary | 12 |  |  |

**Cumulative**

|  |  |  |  |
| --- | --- | --- | --- |
| **Who** | **Estimated** | **Actual** | **Difference** |
| Munieshwar Ramdass | 25 |  |  |
| Ajay Shenoy | 16 |  |  |
| August Tan | 23 |  |  |
| Summary | 64 |  |  |

**12.3. GANTT CHART/MICROSOFT PROJECT SCHEDULE**







**Entire Process**

