Apparel Point of Sale (APOS) System

Project Plan

CMSC 495

University of Maryland University College (UMUC)

Team Tenacious Turtles

Signatures:

John Walsh

Kevin Boyer

Manoj Kumar

Ryan Harvell

Sean Walters

Vance Molhusen

Approval by Professor Mark Wireman:

|  |  |  |
| --- | --- | --- |
| **Revision** | **Date** | **Description** |
| PP-1 | 09/04/16 | Initial Version |
|  |  |  |

INDEX

**1. Project Overview** 5

1.1 Project Summary: 5

1.1.1 Purpose, scope, and objectives 5

1.1.2 Assumptions 5

1.1.3 Constraints 5

1.1.4 Project deliverables 5

1.1.5 Schedule 6

1.1.6 Evolution of the plan 6

**2. References** 6

**3. Definitions** 6

**4. Project Context** 6

4.1 Project Model 6

4.2 Process improvement plan 6

4.3 Infrastructure plan 7

4.4 Methods, tools and techniques 7

4.5 Product acceptance plan 7

4.6 Project organization 7

4.6.1 External interfaces 7

4.6.2 Internal interfaces 8

4.6.3 Authorities and responsibilities 8

**5. Project planning** 10

**5.1 Project initiation** 10

5.1.1 Estimation plan 10

5.1.2 Staffing plan 10

5.1.3 Resource acquisition plan 10

5.1.4 Project staff training plan 10

**5.2 Project work plans** 10

5.2.1 Work activities 10

5.2.2 Schedule allocation 12

5.2.3 Resource allocation 13

5.2.4 Budget allocation 14

5.2.5 Procurement plan 14

**6. Project assessment and control** 14

6.1 Requirements management plan 14

6.2 Scope change control plan 14

6.3 Schedule control plan 14

6.4 Budget control plan 14

6.5 Quality assurance plan 15

6.6 Subcontractor management plan 15

6.7 Project closeout plan 15

**7. Product delivery** 15

**8. Supporting process plans** 15

8.01 Project supervision and work environment 15

8.02 Decision management 16

8.03 Risk management 16

8.04 Configuration management 16

8.05 Information management 16

8.06 Documentation 16

8.07 Communication and publicity 16

8.08 Quality assurance 17

8.09 Measurement 17

8.10 Reviews and audits 17

8.11 Verification and validation 17

**9. Additional plans** 17

**APPENDIX** 18

# **1. Project Overview**

**1.1 Project Summary**:

Our group is designing a simple apparel retail store Point-of-Sale (POS) system. This system will be a cash register management application which will be capable of handling sales and returns with cash payment.

**1.1.1 Purpose, scope, and objectives**:

The scope of the project is development of a cash register application only. It will have user interfaces for: sales associate log-in, sales processing, and store inventory viewing. The application will adjust the store inventory based on sales and returns. Daily sales transactions can be viewed on a report (displayed or printed). The system will display and print two receipts – one for the store and one for the customer after each transaction. Sales associate log-in will be required before performing any interaction with the system.

This system will be designed and developed in five weeks.

## **1.1.2 Assumptions**

A. Maintenance of the following information will be outside the scope of this project. The assumption is that the following is populated by company’s ERP (Enterprise Resource Planning) system. For this project, all of the following information will be pre-populated.

1. Store inventory: sales item, units, location, etc. Replenishment is handled by a separate ERP system.
2. Sales associate information: name, designation, etc. This information is handled by company's personnel management system.
3. Store information: store name, address, manager, etc.
4. Sales Item information: season, style, color, size, gender code, UPC code, etc.
5. Other master information.

B. Payment posting will be outside the scope of this application. The company's ERP system collects sales transactions from each store and then transfers detailed information to the company’s database.

## **1.1.3 Constraints**

As credit card validation is not feasible for this project, the application will be a “cash only” register. The system will not have other integrated peripherals such as UPC code scanner, electronic signature pad, or check reader.

## **1.1.4 Project deliverables**.

1. Project Plan
2. Project Requirements Specifications
3. Systems specifications.
4. Project Design
5. User’s guide
6. Test Plan and results
7. Code development – Phase I
8. Phase I report.
9. Code development – Phase II
10. Phase II report
11. Code development – Phase III
12. Phase III report
13. Implementation

**1.1.5 Schedule**

Project Plan: 09/04/16

Project Requirements Specifications: 09/04/16

Systems specifications: 09/04/16

User guide and test plan: 09/11/16

Project Design: 09/18/16

Code development Phase I: 09/25/16

Code development Phase II: 10/02/16

Code development Phase III: 10/09/16

Implementation with all documents: 10/16/16

## **1.1.6 Evolution of the plan**

Every week from 09/04/16. All team members will be responsible for providing their own content submissions for the separate sections of the plan which will be disseminated via chat channel/s and stored in a central repository accessible by all team members. Members will be responsible for providing updated versions of plan to the entire team for review and approval.

# **2. References**

IEEE System and Software Engineering – Life Cycle Process – Project Management: International Standards. ISO/IEC/IEEE/16326.

# **3. Definitions**

POS: Point of Sale

IEEE: Institute of Electrical and Electronics Engineers

JDK: Java Development Kit

PMP: Project Management Plan

# **4. Project Context**

# **4.1 Project Model**

Design and development of a cash register application using object oriented techniques in five weeks. User guide and test plan will be prepared in first week, system design will be done in second week, development phase 1 will be completed in third week, phase 2 in fourth week, phase 3 in fifth week, and implementation by the end of sixth week.

# **4.2 Process improvement plan**

At the end of every week, the team will be responsible for assessing the project development and determining areas for improvement. Processes may evolve over time due to identifying more efficient development practices and techniques.

# **4.3 Infrastructure plan**

A PC or Laptop with Windows Vista SP2 or higher, 128MB RAM, 124MB HD space and internet connection. Team members will be responsible for ensuring that their equipment has all of the necessary patches and updates to provide the platform functionality that is needed.

# **4.4 Methods, tools and techniques**

Object oriented analysis and design.

System design using UML diagrams. UML design tool will be Violet.

Programming language: Java SDK 8.00 using Netbeans

Database: MySql

# **4.5 Product acceptance plan**

# Product requirements will be re-visited to verify that have been satisfied. Test plan results will be reviewed to verify that all of the functions of the product were thoroughly tested. If not, team members may draft additional testing procedures to be implemented. Once all of the acceptance plan objectives have been met, final approval of product can be established.

# **4.6 Project organization**

Team name: Tenacious Turtles

Team members: John Walsh, Kevin Boyer, Manoj Kumar, Ryan Harvell, Sean Walters, and Vance Molhusen

Team leader: Rotated every week

Note taker: Rotated every week

# **4.6.1 External interfaces**

External interface for the project will be Professor.



# Figure 4.7-1 External Interface

# **4.6.2 Internal interfaces**

Internal interfaces are the team members.

# **4.6.3 Authorities and responsibilities**

Positions are rotated every week. Team members are responsible for finishing assigned task in time. Team leader creates agenda for weekly meeting and preside the meeting. Team leader is also responsible for following up on the activities for a week. The note taker (Quality Assurance Team) is responsible for preparing meeting minutes. All assigned tasks have to be finished by Thursday of every week so that other members of the team can review.



The APOS team consists of the following internal teams, with the number of personnel assigned to each team in parentheses next to the team name:

Program Manager (1) – The program manager is responsible for ensuring the project remains on schedule, and interfacing between the different organizations. He will mitigate any risks, and take any problems raised by the Quality Assurance team on behalf of the customer and find a solution.

Quality Assurance Team (1) – The note taker. This team interfaces with the customer to ensure that the product that is being delivered meets the needs of the customer. Any problem they cannot resolve at their level they will take and pass on to the program manager. They deal with the customers so the engineers don't have to.

Configuration Management Team (1) – The configuration management team is responsible for ensuring the three principle activities of configuration management are performed: configuration identification, configuration control, and configuration accounting.

Software Engineering Team (2) – This is the team which does the programming. The programming instructions are guided by the SDD and controlled by the configuration management team. These are two JAVA developers who are capable of programming this system and developing the user interface. The amount of developers assigned to this project was calculated using the COCOMO model, and discussed in section 5.1.3.

Verification and Validation Team (1) – This team works in conjunction with the Product Test Team in order to ensure that the testing solution covered under the STS has met all the requirements. They also identify and shortfalls during the testing process and provide a solution for resolving them – they are a catch-all for anything which may have been overlooked.

Product Test Team (1) – This team implements the test plan identified by the STS. They perform the tests and route the results up to the V&V team. This will be the same person as the QA team, but they will be dual-hatted for the week they are assigned this duty.

# **5. Project planning**

# **5.1 Project initiation**

## **5.1.1 Estimation plan**

## **5.1.2 Staffing plan**

## The project is staffed by six UMUC CMSC495 students

## **5.1.3 Resource acquisition plan**

## No plan to acquire to more resources.

## **5.1.4 Project staff training plan**

# Team members will seek help from other team members.

# **5.2 Project work plans**

## **5.2.1 Work activities**

|  |  |  |
| --- | --- | --- |
| **Task Name** | **SDLC Task Category** | **SDLC Sub-Task** |
| Project Development Phase I |  |  |
| Generate Teams | Analysis | Analysis |
| Identify Customer Requirements | Analysis | Analysis |
| Develop Goals and Milestones | Analysis | Analysis |
| Develop Budget | Analysis | Analysis |
| Develop Internal Requirements | Design | Analysis |
| Develop Project Plan | Analysis | Analysis |
| Perform Project Plan Review | Design | Design |
| Develop Software Design Document | Design | Design |
| Perform Software Design Document Review | Analysis | Design |
| Develop Software Testing Specification | Design | Design |
| Perform Software Testing Specification Review | Analysis | Design |
| Build and Release APOS Alpha v0.1 | Code | Build |
| Project Development Phase II |  |  |
| QA/Client User Engagement | Test | Testing |
| Develop Initial User Documentation | Code | Documentation |
| Build and Release APOS Beta v0.5 | Code | Build |
| QA/Client User Engagement | Test | Testing |
| Produce Final User Documentation | Code | Documentation |
| Project Development Phase III |  |  |
| Build and Release APOS v1.0 | Code | Build |
| Perform STS Testing | Test | Testing |
| QA/Client User Engagement | Test | Analysis |
| Delivery of APOS | Code | Build |

## Figure 5.2.1-1 Work Activities Chart

The tasks in this chart are mapped to the Gantt chart (Figure 5.2.1-1), which gives more details about the beginning and end of each task. They are grouped together by SDLC task category and sub-category to identify the owning group.

## 

## 

## **5.2.2 Schedule allocation**

## 

## 

## 

## 

Figure 5.2.2-1 Schedule Allocation Gantt Chart

## **5.2.3 Resource allocation**

To calculate the amount of resources we will need for the project and calculate the staff size associated with each piece of the project, we will use the COCOMO (Constructive Cost Model) method.

APOS will focus on the “Organic” class of software project, where the assumption is that the team is small and cohesive, working with fluid requirements. The COCOMO solution will be effective for a quick estimation of the cost of developing this project, but it is not perfect and does not account for personnel quality.

We are working under the assumption that the project will require 500 lines of code.

The coefficients for an organic project are given in the following table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Software project** | ***ab*** | ***bb*** | ***cb*** | ***db*** |
| Organic | 2.4 | 1.05 | 2.5 | 0.38 |

We will insert the required information into the below formulas to estimate the effort, development time, and people required to complete this project:

**Effort Applied (E)** = ab(KLOC)bb

**Development Time (D)** = cb(Effort Applied)db

**People required (P)** = Effort Applied / Development Time

We will first calculate the equation for human effort:

Effort = 2.4 (.5) ^ 1.05 = .48

Now that we know the human effort, we can estimate the development time:

D = 2.6 (.48) ^ .38 = .75 Months

Finally, we can estimate our development staff size by taking the ratio of effort over development time in months:

P = 10.29 / 6.3 = 1.56 Developers required

We of course will round up, so this means the project will require two full time developers on the development team. This estimation does not include the testers, support personnel, program manager, etc. – it only covers the developers themselves. The Gantt chart and schedule reflects the time requirements as calculated by the COMOMO model.

## **5.2.4 Budget allocation**

## This project requires no budget. All software being utilized for development is free.

## **5.2.5 Procurement plan**

# All procurement is done by simply downloading the development programs, ie. Netbeans, Violet, MySQL.

# **6. Project assessment and control**

# **6.1 Requirements management plan**

Team members will be monitoring the functionality of the project to ensure it is on track to meet the established requirements. Upon a need to add or change a requirement, meetings will be held to discuss the need for the change and to analyze any possible impacts that may be caused by the implementation. The outcome of this collaboration will be thoroughly documented and filed accordingly. If needed due to the change, re-evaluations will be conducted to identify any modifications that may need to be made to the project cost, schedule, or risk management analysis. Any changes to these factors will be updated in the PMP documentation.

# **6.2 Scope change control plan**

Any changes to the scope of the project will be evaluated and approved by the entire team. The same as 6.1, team members will meet to discuss the need for the change and to analyze any possible impacts that may be caused by the implementation. The outcome of this collaboration will be thoroughly documented and filed accordingly. Any changes to the project’s functional requirements and/or specifications caused by the scope change will be updated in the PMP documentation.

# **6.3 Schedule control plan**

Any changes to the schedule of the project will be evaluated and approved by the entire team. The same as 6.1, team members will meet to discuss the need for the change and to analyze any possible impacts that may be caused by the implementation. The outcome of this collaboration will be thoroughly documented and filed accordingly. Any changes to the project’s milestones or resources caused by the schedule change will be updated in the PMP documentation.

# **6.4 Budget control plan**

There is no budget allocation for this project. However, if any costs arise during development of the project, the source and sum of the costs will be updated within the PMP documentation.

# **6.5 Quality assurance plan**

The team members responsible for developing the functionality components of the project will periodically work with other members of the team as they complete their tasks in order to review and assess how they are generating their components. This inter-collaboration will ensure that only reliable and accurate processes and techniques are being used to complete the project. Any deviations from these reliable processes will be identified and discussed within the team and corrected or left in place as determined by the team as a group.

# **6.6 Subcontractor management plan**

There will be no use of a subcontracted product for the completion of this project.

# **6.7 Project closeout plan**

At the conclusion of the project the team will validate if the end product meets or does not meet the project requirements. The closeout plan will include the team documenting lessons that were learned during the development of the project and also any roadblocks that were encountered along the way. Recommendations on changes or modifications will be provided in order to assist with similar projects in the future. Team members will ensure that all project documentation has been submitted as needed and any project inquiries have been answered. Any software or tools used for the project will be uninstalled or removed by the individual members themselves.

# **7. Product delivery**

The project will be submitted through the online classroom in sections according to the project schedule below:

Project Plan: 09/04/16

Project Requirements Specifications: 09/04/16

Systems specifications: 09/04/16

User guide and test plan: 09/11/16

Project Design: 09/18/16

Development Phase I: 09/25/16

Phase I Report: 09/25/16

Development Phase II: 10/02/16

Phase II Report: 10/02/16

Development Phase III: 10/09/16

Phase III Report: 10/09/16

Implementation with all documents: 10/16/16

# **8. Supporting process plans**

# **8.01 Project supervision and work environment**

The project will be supervised by the team leader, which will rotate on a weekly basis. The project work environment will be freelance, with individual members contributing from their own work spaces, coordinating with team members through Slack.

# **8.02 Decision management**

Decisions regarding the project and its management will be decided upon a majority vote among the team members.

# **8.03 Risk management**

This project has two main areas of risk: loss of personnel due to course withdrawal, and failure of personnel to meet weekly deadlines. Due to the short time period of the project, loss of personnel during the project with no possibility of replacement would severely negatively impact the project. Due to the importance of this course in the degree plan, this risk has a high impact, but a low likelihood. Failure of personnel to meet weekly deadlines has a moderate impact and moderate likelihood, but consistent communication, and several days between team deadlines and project deadlines, will moderate the risk by allowing other personnel to provide assistance to ensure project deadlines are met.

# **8.04 Configuration management**

This project will use configuration management to ensure the software code, database structure, and documents are archived properly for version control during the project lifespan. The project will use GitHub repositories to handle configuration management.

# **8.05 Information management**

The project will contain information in the forms including but not limited to: text documents, spreadsheets, Java source code, database information. All team members are responsible for all information in the project. As such, all team members will have appropriate access to make revisions as necessary. Documents, spreadsheets, and source code will be held within the project GitHub repository, and database information will be held in the project’s MySQL database.

# **8.06 Documentation**

This project will use documentation throughout the software design implementation and testing. All documents produced will adhere to documentation guidelines. Version controls will be implemented, and all team members will have access to the most recent documentation.

# **8.07 Communication and publicity**

The primary stakeholder for this project is Professor Wireman. Project communication deadlines have been pre-determined which will provide weekly and in-depth updates as to the progress of the project. Communication will primarily be done through the online classroom’s discussion board within the appropriate weekly topic, or through email. The project leader for the week holds responsibility for making sure that communications with the professor are provided at the appropriate intervals.

# **8.08 Quality assurance**

This project will use quality assurance to ensure a high quality for the developed software while conforming to industry standards.

# **8.09 Measurement**

The project will require a set of test database data to function. The data to be generated will be done so at the discretion of the team members as needed.

# **8.10 Reviews and audits**

This project will use reviews and audits to ensure quality control, as well as tracking the team's progress against the schedule in software development. The work of each member of the team will be submitted to the group for review on Thursday of each week during the project to ensure quality standards are met and the project remains on schedule.

# **8.11 Verification and validation**

This project will use verification and validation to provide traceability, reviews, modeling, testing, and inspection of the application software. All aspects of the application software will be properly tested and go through Quality Assurance, as well as proper reviews.

# **9. Additional plans**

None

# **APPENDIX**