PROJECT PLAN

Coffee Cart Rewards Management System for Android

Version 1.0
July 7, 2014

TEAM 04

Casey Bennett
Jassimran Kaur
David Welker
Jeremy Luinstra

Table of Contents

Introduction

Brad and Janet, the owners of the local Atlanta coffee shop "***Lame Ducks***", are planning to install a number of coffee carts in various local spots and wants to put in place a reward system based on VIP cards on which customers can accrue points based on the amount of money they spend and get benefits based on their points. Brad and Janet want the coffee carts' managers to be able to manage the rewards system on their Android devices, and want an app to facilitate the same.

Purpose

This document is the overall plan and guide for the execution of this project. It defines the approach to be used by the Project team to deliver the intended project management scope of the project. It lists "**Why**?" is this solution being developed, "**What**?" product will be delivered, "**Who**?" will be involved & what will be their roles and responsibilities and "**When**?" will the project be completed?

Scope

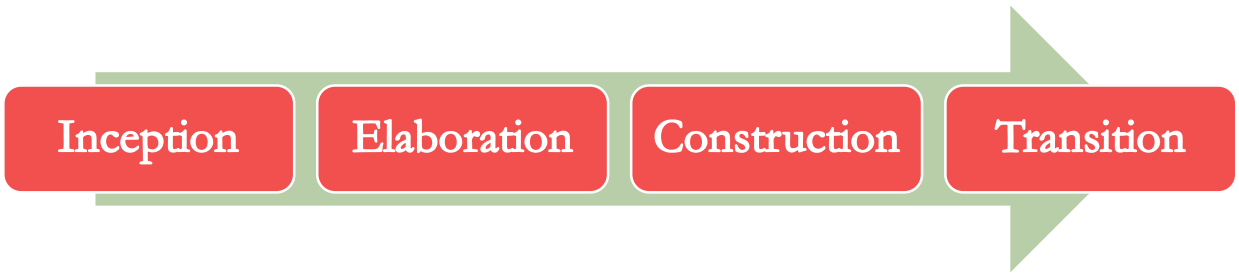
This project aims at creating an android application for managers of "Lame Ducks" to help manage the rewards earned by customers at their coffee carts. The software will be named "**Coffee Cart Rewards Management System**" and will give managers the ability to add, edit customers to VIP list and give points automatically for all purchases. The software will be completed in four weeks by July 20, 2014. It will be simple to use with a friendly graphical interface. This software is neither an inventory management system for "Lame Ducks" nor does it have anything to do with non-VIP customers.

Description of the Project

While developing the Coffee Cart Rewards Management System Android App, Unified Software Development Process, also known as Rational Unified Process (RUP), will be followed. It is an iterative and incremental software process model which can be customized for specific projects. Software components with well-defined interfaces and tight coupling with UML are key features of RUP.

Project Lifecycle

Unified Software Development Process is Use-Case driven, architecture centric, risk focused, iterative and incremental process with following four phases – **Inception, Elaboration, Construction** and **Transition**. Each phase will last for one week. More timeline details have been provided in Estimates section below. The application will be delivered in one cycle.



Phase 1: Inception

- **Activity Name**: Inception
- **Activity Description**: Inception is the smallest phase in this project. Requirements engineering is started in Inception phase along with Analysis and Design of the android application. An approximate project vision and project scope are developed. Use-cases and key requirements are outlined along with identification of initial risks. Preliminary project plan showing different phases and iterations, one or more prototypes are prepared. Project Manager understands the requirements, develops project plan and business case and is responsible for the smooth delivery of the Vision Document, Initial Use-Case model, Initial risk assessment document and a prototype.
- **Entrance Criteria:** Completion of first draft of Project Plan.
- **Exit Criteria:** Completion of Vision Document, an initial version of use-case model, initial risk assessment document, preliminary project plan highlighting different phases and iterations, rough estimates, roles & responsibilities and at least one prototype of the software.

Phase 2: Elaboration

- **Activity Name**: Elaboration
- **Activity Description**: The primary focus in Elaboration phase is to mitigate known risks and to establish and validate the system architecture. Development Manager and his team comes up with the Development Plan and must stabilize the system architecture such that the baseline architecture supports the key software functionality and exhibit the right behavior in terms of performance and scalability. Project Manager, along with Development Manager, Quality Assurance Manager and Documentation Manager understands the requirements and completes Initial design model and any supplementary requirements. 90% complete use-case model and an executable prototype is developed which includes possibly all user interfaces. Quality Assurance Manager comes up with the Test Plan and his team starts writing test cases after understanding the requirements. Revised risk assessment document, revised project plan with sufficient details and accuracy must be delivered. Preliminary user manual should be delivered by Documentation team.
- **Entrance Criteria:** Completion of Phase 1
- **Exit Criteria:** Completion of almost complete use-case model, Supplementary Requirements document, Software architecture, Initial design model, Test cases, Executable prototype, Revised risk assessment, Revised project plan, Preliminary user manual

Phase 3: Construction

- **Activity Name**: Construction
- **Activity Description**: Construction is the largest phase in this project in which building of the software system takes place. The development of components and features of the system is the main area of work in Construction Phase. The bulk of the coding takes place in this phase and all major Use cases will be coded. To divide the use cases into manageable segments, there will be 3 iterations:
    1. First iteration will cover 60% of the use cases.
    2. Second iteration will cover up to 75% of the use cases
    3. The third iteration will cover all remaining use cases.

  After completion of third iteration, an executable software release (Alpha release) will be handed over to the Quality Assurance team for initial Integration and System testing using the test cases created in previous phases. Quality Assurance manager will deliver initial system test results. If the quality of the product is not satisfactory, another testing iteration may be performed. User manual should be released for preliminary reviews and feedback. Other deliverables are: a completed use case model, traceability information for a couple of use cases and an initial version of the software product.
- **Entrance Criteria:** Completion of Phase 2
- **Exit Criteria**: Complete use case model, Traceability information for a couple of use cases, Initial version of the software product, Initial system tests results, User manual

Phase 4: Transition

- **Activity Name:** Transition

- **Activity Description**: Transition phase will bring us to the end of this project. Developers will address enhancement requests and unit test their respective code modules. There will be three regression testing cycles and reported bugs must be fixed and made available for testing in next cycle by development team. Code refactoring, if any be will also be done. Quality team will perform Acceptance testing and ensure that product meets all the expectations of the customer, deployment is smooth and system test results are as per standards. Software will only be released once the System test results are acceptable and there are no known issues or failed test cases. All the documents, code, supporting libraries must be ready for shipping along with necessary artifacts.
- **Entrance Criteria:** Completion of Phase 3
- **Exit Criteria**: Completion of the project and submission of complete code and related artifacts.

Project Tracking, Communication and Control

Project manager will track and control schedule, quality, functionality and etc. throughout the project. To achieve the same, communication will be a key factor. All project communication will be over email and Google Hangouts. One hour long bi-weekly meeting will be held at 21:00hrs Eastern depending on the availability of the team members. The agenda and MOM of the meeting will be shared by Project Manager to make sure everyone is on the same page. Status of work can be shared by team members via email as soon as they are done with their work, so that timely review can be done by the peers and managers.

Environment and Technology

All project documents will be prepared in markdown or pdf format. Gmail, Georgia Tech Email and Google Hangouts will be the communication tools. Eclipse IDE will be used for development of the Application. Android SDK (Android 4.4 API 19 or above) and corresponding Eclipse plugin will allow development from within Eclipse. Windows, Mac or Linux operating system can be used for this project. All systems must have JDK 7 installed for Eclipse to work. Genymotion will be used instead of default Android emulator as it is much faster and easier to use. Google App Engine is used for centrally storing the data. Python will be used for server side coding. Git is used for version controlling and repository for code and related artifacts.

Team Structure

Team Members

Casey Bennett, David Welker, Jassimran Kaur & Jeremy Luinstra are part of the team who are going to work towards developing Coffee Cart Rewards Management System Android App.

Roles

Project Manager	Project Manager is responsible for planning, executing, managing resources, scope of the project and leading the project from its inception to completion.
Development Manager	Development Manager comes up with the development plan, organizes and ensures that development team is able to deliver the product efficiently by following the software development processes. Additionally, since this is a small team, development manager is going to share additional responsibilities of Technical Team Leader, which primarily include the technical delivery of the software by delegating tasks among team members and clarify any issues or problems faced by them.

Developer	Developer designs and develops the solution based on the requirements and unit tests it to make sure it is working as per the design.
Quality Assurance Manager	Quality Assurance manager creates test plan, reviews the requirements and ensures that they are met by setting standards for the quality of the product, preparing test plan, monitoring quality & performance of the product by gathering relevant data and producing statistical reports, against set standards. The QA manager is also responsible for delegating tasks among team members and clarifying any issues or problems faced by them.
Tester	Tester analyzes the requirements, prepares test scenarios, test data, test environment, writes test cases for integration and system testing, tracks defects, and prepares suggestion documents to improve the quality of the application.
Documentation Manager	Documentation Manager is responsible for development & publishing of software product documentation such as user manual, implementation guides and any other necessary documents.
Technical Writer	Technical writer develops the user manual, installation manual, implementation guides and any other necessary documents.

Team Responsibilities

Project Manager	Jassimran Kaur
Development Manager	Jeremy Luinstra
Developers	Casey, David , Jassimran
Quality Assurance Manager	Casey Bennett
Testers	David, Jassimran, Jeremy
Documentation Manager	David Welker
Technical Writers	Casey, Jassimran, Jeremy

Estimates

Schedule

This project will be four weeks long and final delivery date is 20th July, 2014. Time estimations for this project are:

ACTIVITY	TIMELINES
Phase 1	23rd June, 2014 – 29th June, 2014
Phase 2	30th June, 2014 – 6th July, 2014
• Requirement Gathering	30th June, 2014 – 2nd July, 2014
• Design & Analysis	30th June, 2014 – 2nd July, 2014
• Prototype Development	2nd July, 2014 – 5th July, 2014
Phase 3	7th July, 2014 – 13th July, 2014
• Iteration 1	7th July, 2014 – 9th July, 2014
• Iteration 2	9th July, 2014 – 11th July, 2014
• Iteration 3	11th July, 2014 – 12th July, 2014

• Iteration 4, (if necessary)	12th July, 2014 – 13th July, 2014
Phase 4	14th July, 2014 – 20th July, 2014
• Refactoring	14th July, 2014 – 17th July, 2014
• Testing (3 iterations)	17th July, 2014 – 19th July, 2014
• Reviewing documents	19th July, 2014 – 20th July, 2014

Effort Hours

An approximate of 40-50hours per team-member is expected to be spend on this project

Lines of Code

There will be approximately 5000-6000 lines of code.

Defects

The total number of defects expected to be found in this project depend on numerous factors like understanding the requirements fully, effective project plan, using correct tools and techniques, number of developers and their experience level, domain knowledge of technology and etc. We can expect to find anywhere from 50 to 150 defects while testing the software. However, all these issues will be fixed before it is released to customers.

Supporting Plans

Development Plan

After reviewing use case model and finalizing the Software architecture, development team will start UI design of the Application. Each team member will be assigned a Use Case to work upon. In Initial prototype development, focus will be laid on implementing basic functionality of only major use cases –

- Add Customer,
- Edit Customer,
- Purchase Item and
- Preorder Item.

Most of the Use Cases will be completed in 3rd phase. During the 4th phase of the project, refactoring of already implemented use cases will be undertaken. UI design will be fine-tuned and integration with the database server will be worked upon. As the various testing cycles begin, bugs and enhancements will be fixed by corresponding members who developed the Use-case.

Test Plan

This software will be tested using Black box testing methodology. All the bugs will be tracked and reported using Git's Issue Tracker. Enhancement requests, if any, will also be tracked using Git Tracker. Unit testing will be responsibility of the developers. Quality Assurance team will focus on System and Integration testing.

Immediately after the alpha release, application will be deployed by Quality Assurance team and testing activity will start. Regression testing will be performed in 4th phase of project, i.e. Transition and there will be three testing iterations. Timeline details have been elaborated in Schedule.

There will be no Beta testing involved.

*****