

CARGO TRACING AND BUSINESS ANALYSIS

Group Id- 23

Guide- Prof. Kiran Kumari

1514068- Saurabh Baj

1514074- Niyati Daftary

1514080- Jayesh Gaur

1514085- Aditya Jawalika

SOFTWARE PROJECT MANAGEMENT PLAN

Contents

1	INTRODUCTION	4
1.1	Project Overview	4
1.2	Project Deliverables	5
2	PROJECT ORGANIZATION	6
2.1	Software Process Model	6
2.2	Roles and Responsibilities	7
2.3	Tools and Techniques	8
3	PROJECT MANAGEMENT PLAN	9
3.1	Tasks	9
3.1.1	Task 1 - Requirement analysis	9
3.1.2	Task 2 - Software requirement specification	9
3.1.3	Task 3 - Software design document	10
3.1.4	Task 4 - System Test Document	10
3.1.5	Task 5 - Coding and Hardware Integration	11
3.2	Risk Table	12
3.3	Risk Template	13
3.4	Risk Template	13
3.5	Risk Template	14
3.6	Risk Template	14
4	TIMETABLE	15

1 INTRODUCTION

1.1 Project Overview

The Software Project Management Plan Document gives details about the project deliverables, the organisation of the project and the project plan which will tell us how the project will be moving forward. The Project organisation chapter would consist of the software process model which we will be using for the whole system. It also consists of the Roles and Responsibilities along with the tools and Techniques which will be using for the project. The Management plan would be defining various tasks which need to be carried out. At last, there will be risk identification and categorization along with the impact which those risks will be having on the project. There will be risk sheets prepared with all the details of the risks.

1.2 Project Deliverables

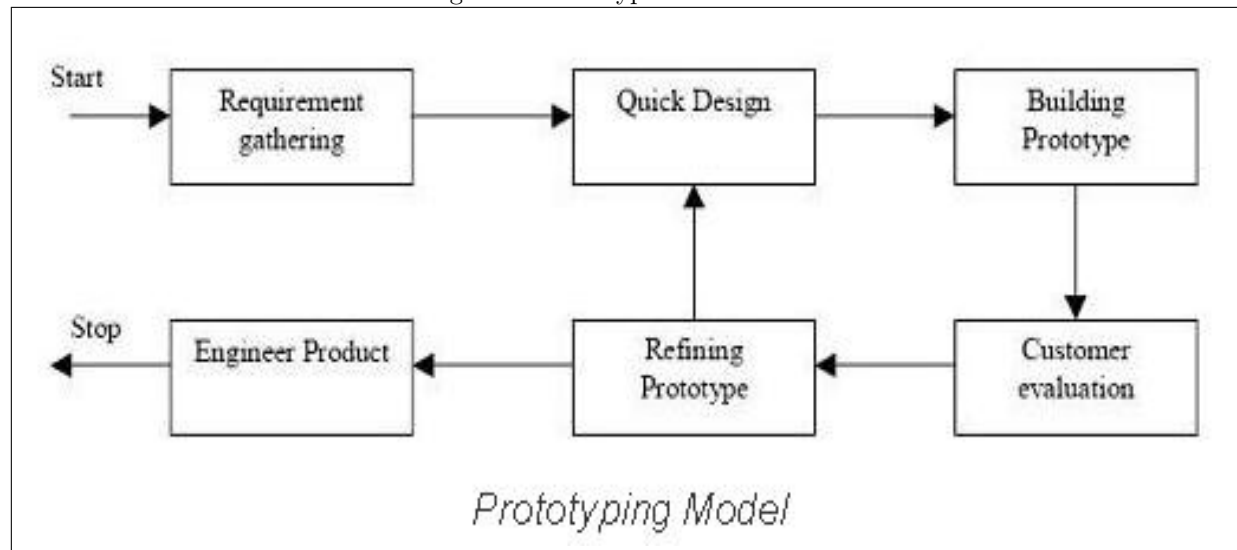
Deliverables	Description	Delivery Date
Software Project Management Plan	A complete formal project plan, including technical and managerial processes that will be implemented in the development and delivery of the system	30/09/2018
Software Requirements Specification	A formal document detailing the functional and non-functional requirements of the system	30/09/2018
Software Design Specification	A formal document detailing the component designs as well as the relationships among components	05/10/2018
Software Test Document	Formal documentation detailing scenarios that must be followed in order to ensure that the product software is satisfactorily tested.	14/10/2018
Implementation of frontend and hardware components.	Software files to form the backbone of the system and their integration into the hardware used	14/02/2019
Final Presentation	A demonstration of the product software and a presentation of the project experience	March 2019

2 PROJECT ORGANIZATION

2.1 Software Process Model

The process model used is Prototype Process Model. Prototype is a working model of software with some limited functionality. The prototype does not always hold the exact logic used in the actual software application and is an extra effort to be considered under effort estimation. Prototyping is used to allow the users evaluate developer proposals and try them out before implementation. It also helps understand the requirements which are user specific and may not have been considered by the developer during product design.

Figure 1: Prototype Model



2.2 Roles and Responsibilities

Roles	Responsibilities
Project Co-ordinator	Motivate the team members to perform their tasks in a organized manner Help the team in allocating deadlines
Project Guide	Works with the team to help formulate the application strategy Approves the project documents Helps the team in analysing the project from every perspective and set the goals
Project Manager	Plans, organizes, and coordinates the team project Schedules and prepares team meetings Resolves conflicts Works as a link between team members Monitors and reports the weekly status of the team Ensures that project deliverables are met
Application Designer	Designs a web application to the problem statement that satisfies the requirements Assists the Technical Writers in documenting the design
Application Developer	Develops the android application Determines the data needs for the solution Determines what hardware and tools are necessary Fixes bugs found by the Testers
Database Developer	Develops and populates Databases Ensures proper operation and interaction with entire system application
Tester	Tests all the application modules
Technical writer	Coordinates the project documents and their review by all team members Collects, proofreads, and integrates document parts Generates the final version of all the documents

2.3 Tools and Techniques

- LATEX for SRS, SPMP, SDD, STD
- Gantt project for planning and to prepare the time-line chart
- Rational rose for UML diagrams
- Microsoft powerpoint for presentation for the users and project personnel
- Python-based kivy framework
- MATLAB to implement the image-processing algorithms
- Python-based image processing libraries

3 PROJECT MANAGEMENT PLAN

3.1 Tasks

3.1.1 Task 1 - Requirement analysis

Description

Definition of the different requirements which will help the users get a good gist of the project. It provides the basic understanding of the problem and nature of the solution.

Deliverables and milestones

The task provides a list of the various requirements and their analysis for paving the path of design phase.

Resources needed

Effort, time and knowledge about the software.

Dependencies and constraints

The requirements must be documented, testable and related to the needs and defined to a level sufficient for system design.

Risks and contingencies

If the team does not have knowledge about the software, then it can gather the information by communicating with the experts in that field.

3.1.2 Task 2 - Software requirement specification

Description

Description of the behaviour of the system to be developed and the features in the scope of the project.

Deliverables and milestones

SRS delineates the features of the project and serves as a guide to the developers.

Resources needed

Latex

Dependencies and constraints

SRS should be documented in a way understandable to other developers to identify the aspects of the system.

Risks and contingencies

There is a high amount of risk if the SRS is not well documented as the features of the system will not be clear.

3.1.3 Task 3 - Software design document

Description

The structure of the software to satisfy the requirements. It specifies the software structure, components, interfaces and data necessary for implementation.

Deliverables and milestones

Architecture design, data design, interface and procedural design.

Resources needed

Latex for documentation and IBM Rational Rose for designing purposes

Dependencies and constraints

SDD is developed according to the SRS, so the SRS should provide an entire overview of the system

Risks and contingencies

Risk is involved if the design does not follow the requirements . The design can be revised by proper communication among the development team.

3.1.4 Task 4 - System Test Document

Description

Specifies the approach that ensures that the features are adequately tested.

Deliverables and milestones

The document includes all the test cases with results done after finishing the development.

Resources needed

Latex and software test plan

Dependencies and constraints

STD should give entire description about features to be tested, amount of testing in order to save time of the testing team

Risks and contingencies

The risk is when the STD does not cover the entire system as this might cause major problem in the future which can be avoided by developing test cases for entire section wise coverage.

3.1.5 Task 5 - Coding and Hardware Integration

Description

Actual programming and functionalities of the application

Deliverables and milestones

The different modules and components of the system.

Resources needed

Arduino, RFID tags and sensors.

Dependencies and constraints

Coding phase depends on the SRS and SDD and should be flexible.

Risks and contingencies

Developers may have insufficient amount of knowledge.

3.2 Risk Table

Risks	Category	Probability	Impact	Preventive measures
Server Crash	TI	10%	1	Maintain a distributed server system
Computer crash	TI	20%	3	Powerful computers capable of handling high load
Late delivery	BU	30%	2	Implementation of basic functionality first and parallelism in work
Deviation from Software Engineering Standards	PI	50%	2	Proper design standards and principles must be followed
Poor Quality Documentation	BU	50%	2	Proper understanding of the requirements
Lack of Database Stability	TI	40%	2	Update DB Structure as the traffic grows
Software failure	TI	20%	1	Maximize portability
Staff is inexperienced	ST	40%	3	Self-learning using various resources providing correct knowledge
No internet Connection	TI	10%	1	Maintain a backup hotspot/tethering service
Conflict with other traffic	TI	10%	1	Shield the high frequency signals from external noise
Failure of Scanner	TI	30%	2	Facilities of updating the database manually.
Damage of RFID	TI	20%	2	Attach the RFIDs in such a way they aren't easily accessible.

Impact Values:

- 1 – Catastrophic
- 2 – Critical
- 3 - Marginal
- 4 - Negligible

3.3 Risk Template

Risk information sheet			
Risk ID: 1	Date: September 30, 2019	Probability: 40%	Impact:2
Description: The database maintained may not be stable which may lead to Database Instability.			
Refinement/Context: Sub-condition 1: The information gathered was misinterpreted.			
Mitigation/Monitoring: 1. Re-gather the information from the user. 2. Understand with modules are improper and correct them.			
Management/Contingency plan/Trigger: Contact the team leader and make a new increment with all the respective changes needed.			
Current status: Mitigation steps have been initialized.			
Originator:		Assigned:	

3.4 Risk Template

Risk information sheet			
Risk ID:2	Date:September 30, 2019	Probability:10%	Impact:1
Description: The internet connection fails.			
Refinement/Context: Sub-condition 1: Connection fails due to fault at the ISP. Sub-condition 2: There is some lose contact of cables to the modem.			
Mitigation/Monitoring: 1. Contact the ISP provider and resolve the issues. 2. Fix the cable properly			
Management/Contingency plan/Trigger: Provide some alternative solution for internet like hotspot. Or make a provision for offline data storage.			
Current status: Mitigation steps have been initialized.			
Originator:		Assigned:	

3.5 Risk Template

Risk information sheet			
Risk ID:3	Date:September 30, 2019	Probability:30%	Impact:2
Description: The scanners fail to read the RFIDs.			
Refinement/Context: Sub-condition 1: RFID / Scanner interface is tampered. Sub-condition 2: Incompatible scanner and RFIDs			
Mitigation/Monitoring: 1. Make sure the interface is clean. Check if the RFID is not damaged. 2. Try scanning the same RFID on another sensor or scan another RFID on the sensor.			
Management/Contingency plan/Trigger: Get technical assistance as soon as possible. Find out the problem location (Scanner or RFID) and take actions accordingly			
Current status: Mitigation steps have been initialized.			
Originator:		Assigned:	

3.6 Risk Template

Risk information sheet			
Risk ID:4	Date:September 30, 2019	Probability:20%	Impact:2
Description: RFID tag gets damaged.			
Refinement/Context: Sub-condition 1: The RFID tag is missing from a cargo when it reaches a scanner. Sub-condition 2: The tag is damaged due to weather conditions / mishandling, etc.			
Mitigation/Monitoring: 1. . Try to identify the cargo with the marking on it. 2. Wait for the user to raise a ticket when he notices that one of his cargo containers has not reached the desired destination yet. The cargo can then be linked with the ticket and the right identity of it can be found. A new RFID can be associated to it at this point and the tracing can continue as before.			
Management/Contingency plan/Trigger: Have spare RFIDs at all counters.			
Current status: Mitigation steps have been initialized.			
Originator:		Assigned:	

4 TIMETABLE

Figure 2: Gantt Chart

Name	Begin date	End date
☐ • Initialisation	4/16/18	4/16/18
• Problem Definition	4/16/18	4/16/18
☐ • Planning	9/24/18	10/19/18
• Project Plan (SPMP)	9/24/18	9/30/18
• Requirements (SRS)	10/1/18	10/7/18
• Design (SDD)	10/8/18	10/11/18
• Test Case Design (STD)	10/12/18	10/19/18
☐ • Implementing the base	10/20/18	1/11/19
• UI development	10/20/18	11/10/18
• Configuration of hardware	11/11/18	12/10/18
• Linking database	12/11/18	1/11/19
☐ • Implementing the logic	1/12/19	3/12/19
• UI phase 2 development	1/12/19	1/31/19
• Implement DM algorithms	2/1/19	3/2/19
• Integreate modules to the mai...	3/3/19	3/12/19
☐ • Final Documentation	3/13/19	4/11/19
• Technical Paper	3/13/19	3/22/19
• Testing	3/23/19	4/11/19
• Demonstration	4/12/19	5/11/19

Figure 3: Gantt Chart

