Project Proposal

On

**Mobile Shop Management System**

A close up of a machine

Description automatically generated

Rupesh Thapa

00175010

Computing Project

Level 5 in Computing

Softwarica College of IT & E-commerce

Kathmandu, Nepal

04/03/2019

Submitted To: Niman Maharjan

Table of Contents

[**Table of Figures** 3](#_Toc5566003)

[**Chapter 1: Introduction** 4](#_Toc5566004)

[**1)** **Project Introduction** 4](#_Toc5566005)

[**2)** **Justification for the project** 4](#_Toc5566006)

[**a)** **Background of the project** 4](#_Toc5566007)

[**b)** **Problem Statement** 4](#_Toc5566008)

[**3)** **Description of the project** 5](#_Toc5566009)

[**a)** **Features of the system** 5](#_Toc5566010)

[**4)** **Overview of the project** 5](#_Toc5566011)

[**Chapter 2: Scope of the project** 6](#_Toc5566012)

[**1)** **Scope** 6](#_Toc5566013)

[**2)** **Limitation** 6](#_Toc5566014)

[**3)** **Aims** 6](#_Toc5566015)

[**4)** **Objectives** 6](#_Toc5566016)

[**5)** **Overview of the Scope** 6](#_Toc5566017)

[**Chapter 3: Development Methodology** 7](#_Toc5566018)

[**1)** **Waterfall Model** 7](#_Toc5566019)

[**2)** **Design Pattern** 8](#_Toc5566020)

[**3)** **System Architecture** 9](#_Toc5566021)

[**Chapter 4: Project Plan Work Breakdown Structure** 10](#_Toc5566022)

[**1)** **Work Breakdown Structure** 10](#_Toc5566023)

[**2)** **Milestones** 12](#_Toc5566024)

[**3)** **Scheduling** 13](#_Toc5566025)

[**a)** **Time Estimation Table and Gantt Chart** 13](#_Toc5566026)

[**Chapter 5: Risk Management** 15](#_Toc5566027)

[**Chapter 6: Configuration Management** 17](#_Toc5566028)

[**Chapter 7: Conclusion** 18](#_Toc5566029)

[**Chapter 8: References** 19](#_Toc5566030)

# **Table of Figures**

[Figure 1: Waterfall Model 7](#_Toc5565968)

[Figure 2: MVC pattern 8](#_Toc5565969)

[Figure 3: 2-tier architecture 9](#_Toc5565970)

[Figure 4: WBS of Mobile Shop Management System 10](#_Toc5565971)

[Figure 5: Time estimation table Gantt chart 14](#_Toc5565972)

[Figure 6: Gantt chart for project 14](#_Toc5565973)

[Figure 7: Files added on git hub 17](#_Toc5565974)

[Figure 8: Tree structure of Mobile Shop MS 18](#_Toc5565975)

# **Chapter 1: Introduction**

# **Project Introduction**

The technology has advanced rapidly from past few years and people want to do the daily task faster and easier. People want easy and stress less life. They don’t want to stand and wait on a long queue to complete their task. Technology such as smartphone, computer, internet and software systems help people make task simpler and easier. So, the software systems or application save time and the proposed system is Mobile shop Management System where employee use it to make task easier and provide good service to customers.

# **Justification for the project**

## **Background of the project**

Mobile shop management system is the desktop application for a mobile shop where employee use it to keep the stock records, customer and invoice records as well as employee information.

In this, desktop application employee is the user as all the tasks are performed by them and application perform overall works of the shop. It also records customer information and their invoice information. Employee need to register and logged in for using the system.

The system is a user-friendly application and user can easily understand to use it. User can record, update, delete and search the data easily through this application. This application is developed using C-sharp programming language and SQL Server for recording data as back end database.

## **Problem Statement**

The old system is paper based, and they store their records manually. It is time consuming to search any record. Wastages of resources to store old record books. When a customer requests for the model, employee might have to check availability of the stock in storage physically. The current system is outdated and old.

The proposed system is expected to overcome all these problems and makes the day to day business or work easier than now. It will make the system more secure reduce the use of paper and help creating invoice as well.

# **Description of the project**

## **Features of the system**

The system consists of following features which makes the work of shop faster, reliable and easier.

* Employee can add employee, customer and invoice record to the system:

The system allows user/employee to register and add their details and add customer, invoice details as well as stock records.

* Employee can update and remove the recorded details:

The system allow user to remove the details as well as update according to the requirements.

* Bills can be generated using this system for customers:

This system allows an employee to create invoice for customers and store it as well.

* Employee can search, record and view information:

Employee can track the stock details as well as search all the information of employee and customer.

* Owner can keep track of all details:

The owner can usually keep track of profit and loss through invoice and stock records.

# **Overview of the project**

The system that I proposed to develop will contain above mentioned features and function, but some extra features might be added at the final product as the proper analysis of requirement will be done later. The proposed mobile shop management system is for removing the problems faced by old methods and techniques in shop. The system is being developed to remove the problem mentioned in problem statement.

# **Chapter 2: Scope of the project**

# **Scope**

Mobile shop management system is the desktop system for employee of the shop. It helps the employee to easily create an invoice of a customer rather than paper-based bill. The system also helps keeping track of shop transaction on daily basis. All the data are stored on a secure database. It provides user friendly interface for employees which reduces paper work. It will make functioning of store faster and will reduce human error. The proposed system is more efficient than the existing system.

# **Limitation**

The limitation of proposed system are as follows:

* End user should have the basic knowledge of the computers.
* This system will only available till the system on which it is installed is running.
* The system is not portable as it is a standalone application running on single system with no shared database.

# **Aims**

The main aim of the project is to create a digital system and remove the old paper based working environment. It also aims to create efficient system than existing one, to reduce human error and to create a faster functioning system to reduce time and cost of the organization or shop. It also aims reducing time and cost with application capable of securing data.

# **Objectives**

The objectives of proposed system are as follows:

* It will reduce human error and make functioning of store faster.
* It will provide user friendly interface for employees which reduces paper work.
* The system is more efficient than the existing system.
* It helps creating record, updating existing data and removing unnecessary data.

# **Overview of the Scope**

The proposed system will cover above mentioned scope for the shop along with aims and objectives but there is some limitation of it as well which can be improved later.

# **Chapter 3: Development Methodology**

# **Waterfall Model**

For, the development of this software, I have applied waterfall model. Waterfall model is a traditional approach where the process is carried out step wise after the completion of one step another one is carried out. The system proposed is a small project and all the objectives are clear so, waterfall model is best approach. Waterfall model has six steps which is carried out one at a time. The waterfall model is a sequential design process in which progress is seen as flowing steadily downwards (like a waterfall) through the phases of Requirement Analysis, Design, Implementation, Testing, Deployment and Maintenance.

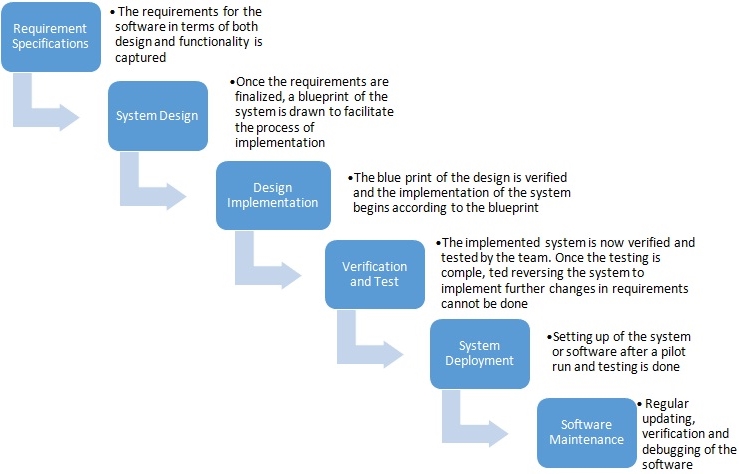


Figure 1: Waterfall Model

In, the first stage requirements of the system are gathered and understand what needs to be design. Then from the reference of this stage, design is made of the system. From design stage the system is implemented using programming language and once implemented the testing of the system is carried out to find any error or bug. Then the system is deployed and maintained regularly with updates.

# **Design Pattern**

A design pattern is a general repeatable solution to a commonly occurring problem in software design. Design pattern are battle tested solution but not a complete one. Design patterns can speed of the development process by providing tested, proven development paradigms. I am using MVC (Model View Controller) design pattern in this project. In, MVC pattern codes are separated according to their nature in system.

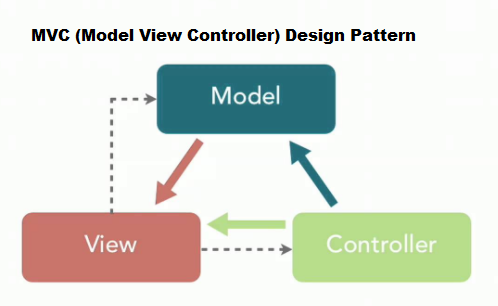


Figure 2: MVC pattern

**Model:** Model controls the data that user work with. It is the central component of the pattern. It is the applications dynamic data structure, independent of the user interface. It directly manages the database and data related code.

**View:** It handles the user interface part of the system. It represents information for user to interact with the system.

**Controller:** It acts as an intermediary between model and view to process incoming request. It is the logic part of system. The controller responds to the user input and perform interaction on the data model objects.

# **System Architecture**

System architecture is the structural design of systems and it is a conceptual model. System architecture serves as the blueprint for both the system and the project developing it, defining the work assignments that must be carried out by design and implementation teams. (Software Architecture, 2017) It shows the relationship between the system. For, this project I will be using 2-tier architecture.

A screenshot of a cell phone

Description automatically generated

Figure 3: 2-tier architecture

**Database Tier:** It is the database server. In this layer data and information are stored and from here data are retrieved. This is independent of other layer and business logic.

**Client Tier:** This layer displays information to the user through interface. It is the communication layer for user to communicate with database.

I will be using two tier architecture in the project as the proposed application is a desktop application. It also provides added security to the DBMS as it is not exposed to the end user directly.

# **Chapter 4: Project Plan Work Breakdown Structure**

# **Work Breakdown Structure**

WBS is a key project deliverable that organizes the team’s work into manageable sections. The work breakdown structure visually defines the scope into manageable chunks that a project team can understand, as each level of the work breakdown structure provides further definition and detail. (Work breakdown structure, 2019) This method is used by most of company to complete project in time. The project team creates the project work breakdown structure by identifying the major functional deliverables and subdividing those deliverables into smaller systems and sub-deliverables. These sub-deliverables are further decomposed until a single person can be assigned.

A blue and white sign

Description automatically generated

Figure 4: WBS of Mobile Shop Management System

|  |  |  |
| --- | --- | --- |
| **WBS** | **TASK NAME** | **NUM OF DAYS** |
| **0** | **Mobile Shop Management System** | **110** |
| **1**  1.1  1.2  1.3 | **PROJECT PROPOSAL**  Scope  Planning  Monitoring | **16**  5  8  3 |
| **2**  2.1  2.2  2.3  2.4 | **ANALYSIS**  Feasibility Study  Gathering Requirements  Use Case  Class Diagram | **29**  8  8  6  7 |
| **3**  3.1  3.2  3.3  3.4 | **DESIGN**  Structural Model  Behavioral Model  UI Design  Database Design | **26**  6  6  7  7 |
| **4**  4.1  4.2 | **IMPLEMENTATION**  Build Database  Coding | **21**  5  16 |
| **5**  5.1  5.2  5.3  5.4 | **TESTING**  Unit Testing  Integration Testing  Black Box Testing  White Box Testing | **7**  2  1  2  2 |
| **6**  6.1  6.2 | **FINAL DOCUMENTATION**  User Manual  Final Report | **11**  5  6 |

# **Milestones**

Milestones refers to the information which tracks the work or task that is being performed. The milestone created for the project helps tracking start and end date of a process in a software development.

|  |  |  |
| --- | --- | --- |
| **TITLE** | **START DATE** | **END DATE** |
| 1. **Proposal** | **25TH March 2019** | **9th April 2019** |
| 1.1 Scope | 25th March 2019 | 29th March 2019 |
| 1.2 Planning | 30th March 2019 | 6th April 2019 |
| 1.3 Monitoring | 7th April 2019 | 9th April 2019 |
| 1. **Analysis** | **10th April 2019** | **8th May 2019** |
| 2.1 Feasibility Study | 10th April 2019 | 17th April 2019 |
| 2.2 Gathering Requirements | 18th April 2019 | 25th April 2019 |
| 2.3 Use Case | 26th April 2019 | 1st May 2019 |
| 2.4 Class Diagram | 2nd May 2019 | 8th May 2019 |
| 1. **Design** | **9th May 2019** | **3rd June 2019** |
| 3.1 Structural Model | 9th May 2019 | 14th May 2019 |
| 3.2 Behavioral Model | 15th May 2019 | 20th May 2019 |
| 3.3 UI Design | 21st May 2019 | 27th May 2019 |
| 3.4 Database Design | 28th May 2019 | 3rd June 2019 |
| 1. **Implementation** | **4th June 2019** | **24th June 2019** |
| 4.1 Build Database | 4th June 2019 | 8th June 2019 |
| 4.2 Coding | 9th June 2019 | 24th June 2019 |
| 1. **Testing** | **25th June 2019** | **1st July 2019** |
| 5.1 Unit Testing | 25th June 2019 | 26th June 2019 |
| 5.2 Integration Testing | 27th June 2019 | 27th June 2019 |
| 5.3 Black Box Testing | 28th June 2019 | 29th June 2019 |
| 5.4 White Box Testing | 30th June 2019 | 1st July 2019 |
| 1. **Final Documentation** | **2nd July 2019** | **12th July 2019** |
| 6.1 User Manual | 2nd July 2019 | 6th July 2019 |
| 6.2 Final Report | 7th July 2019 | 12th July 2019 |

Description of Milestone:

**Proposal**

For, the proposal submission I have assigned 16 days which is divided into three parts 5 days for scope identifying, 8 days for planning and 3 days for monitoring the project.

**Analysis**

I allocated total 29 days for this process i.e. 8 days for feasibility study, 8 days for gathering requirements, 6 days for use case diagram and 7 days for class diagram.

**Design**

For, design process 26 days is allocated, and it is divided into structural model 6 days, behavioral model for 6 days and UI design as well as Database design with 7-7 days each.

**Implementation**

Total 21 days is allocated for implementation phase which is divided into database building 5 days and 16 days for coding.

**Testing**

Testing phase is allocated 7 days which is divided into 2 days each for unit testing, black box, white box testing and 1 day for integration testing.

**Final Document**

For, final document and software deployment total 11 days is allocated where 5 days is allocated for user manual as well as 6 days for final report.

# **Scheduling**

## **Time Estimation Table and Gantt Chart**

Gantt chart refers to the representation of a graphical chart showing the start date, end date and duration of a task. A Gantt chart, commonly used in project management, is one of the most popular and useful ways of showing activities (tasks or events) displayed against time. Time estimation table is showed on the left side and chart representing time scale is shown on right side. It is graphical illustration that helps planning, tracking, coordinating and scheduling different task and sub-task of the project. The below chart shows the time scale of project with duration, start date, end date and overlapped activities of project.

A screenshot of a cell phone

Description automatically generated

Figure 5: Time estimation table Gantt chart

A screenshot of a video game

Description automatically generated

Figure 6: Gantt chart for project

# **Chapter 5: Risk Management**

Software development is activity that uses a variety of technological advancements and requires high levels of knowledge. The success of a software development project depends quite heavily on the amount of risk that corresponds to each project activity. To achieve a successful outcome, project leadership must identify, assess, prioritize, and manage all the major risks. Risk Management is the process of identify, analyzing of risk factor in project. It should be part of planning process to figure out risk in the project and control risk for future events. (Risk Management, 2019)

The list of possible risks for the project are as follows:

* Natural Disaster
* Hard Disk Failure
* Requirement does not meet
* Equipment Failure
* Server Failure
* Lack of Skilled Manpower
* Insufficient Resources

The risks impact is calculated, and solution is given below through likelihood and consequences table of a risks.

**Impact = Likelihood \* Consequence**

Risk Likelihood values are shown as follows

|  |  |
| --- | --- |
| **Likelihood** | **Value** |
| Low | 1 |
| Medium | 2 |
| High | 3 |

Risk Consequence values are shown below

|  |  |
| --- | --- |
| **Consequence** | **Value** |
| Very low | 1 |
| Low | 2 |
| Medium | 3 |
| High | 4 |
| Very High | 5 |

Risk Consequences values are shown below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.NO** | **Risks** | **Likelihood** | **Consequences** | **Impact** | **Solution** |
| 1 | Natural Disaster | 1 | 4 | 4 | Backup Plan |
| 2 | Hard Disk Failure | 1 | 5 | 5 | Data should be backed up for example: cloud storage e.t.c. |
| 3 | Requirement  Does not meet | 2 | 4 | 8 | Requirement analysis should be properly done. |
| 4 | Equipment Failure | 2 | 4 | 8 | System should be checked time to time and regular maintenance. |
| 5 | Server Failure | 1 | 5 | 5 | Proper backup of data and should be kept securely. |
| 6 | Lack of Skilled Manpower | 1 | 3 | 3 | Proper training should be provided. |
| 7 | Insufficient Resources | 2 | 3 | 6 | All required resources should made available in time. |

# **Chapter 6: Configuration Management**

Configuration management refers to the system or term which tracks software, hardware and related information of the system. Configuration managementfocuses on creating and continuing steadiness of a product's performance, and its functional and physical attributes with its requirements, design, and operational information throughout its life. (Configuration Management, 2017)

Version controls are the category of software tools that helps to manage source code for the software team. It allows you to revert selected files back to a previous state, revert the entire project back to a previous state, compare changes over time, see who last modified something that might be causing a problem, who introduced an issue and when, and more. Using a Version Control System also generally means that if you screw things up or lose files, you can easily recover.

Git is one of the Distributed Version Control Systems where clients don’t just check out the latest snapshot of the files; rather, they fully mirror the repository, including its full history. Thus, if any server dies, and these systems were collaborating via that server, any of the client repositories can be copied back up to the server to restore it.

Git id for this project is: <https://github.com/rupeshthapa123>

A screenshot of a cell phone screen with text

Description automatically generated

Figure 7: Files added on git hub

A screen shot of a social media post

Description automatically generated

Figure 8: Tree structure of Mobile Shop MS

# **Chapter 7: Conclusion**

Mobile shop management system is a customized and user-friendly desktop application which manages most of the work of shop like stock records, customer records and invoices as well as employee information. Waterfall methodology and MVC design pattern is selected for project development and different tasks such as risk management, configuration management is carried out. The duration, timeline of different project task is also planned in the proposal.

# **Chapter 8: References**

1. Software Architecture (Dec 2017). *Carnegie Mellon University.* [Online] Available at: [https://www.sei.cmu.edu/research-capabilities/all work/display.cfm?customel\_datapageid\_4050=21328](https://www.sei.cmu.edu/research-capabilities/all%20work/display.cfm?customel_datapageid_4050=21328) [Accessed: 30 March 2019].
2. Work breakdown structure (2019). WBS. [Online] Available at: <https://www.workbreakdownstructure.com/> [Accessed: 02 April 2019].
3. Risk Management (2019). CAST. [Online] Available at: <https://www.castsoftware.com/research-labs/risk-management-in-software-development-and-software-engineering-projects> [Accessed: 04 April 2019].
4. Configuration Management (5 June 2017). C2S consulting group. [Online] Available at: <https://c2sconsultinggroup.com/the-importance-of-configuration-management/> [Accessed: 06 April 2019].