**Project Proposal Report on**

**Liquor Stock Management System**



Ajar Maharjan

22 C

NCC ID – 00175039

Computing Project

Level 5 Diploma in Computing

Software College of IT & E-Commerce

Kathmandu, Nepal

9th April 2019

Contents

[Chapter 1: Introduction 1](#_Toc5564819)

[1.1 Introduction to the project 1](#_Toc5564820)

[1.2 Background of the project 1](#_Toc5564821)

[1.3 Problem statement 1](#_Toc5564822)

[1.4 Description of the project 2](#_Toc5564823)

[1.4.1 Features of the project 2](#_Toc5564824)

[1.5 Overview of the project 2](#_Toc5564825)

[Chapter 2: Scope 3](#_Toc5564826)

[2.1 Scope 3](#_Toc5564827)

[2.2 Limitation 3](#_Toc5564828)

[2.3 Aims 3](#_Toc5564829)

[2.4 Objectives 3](#_Toc5564830)

[2.5 Overview of the scope 4](#_Toc5564831)

[Chapter 3: Development Methodology 5](#_Toc5564832)

[3.1 Development Methodology 5](#_Toc5564833)

[3.2 design pattern 6](#_Toc5564834)

[3.3 Architecture 7](#_Toc5564835)

[Chapter 4: Project Planning 8](#_Toc5564836)

[4.1 Work Breakdown Structure (WBS) 8](#_Toc5564837)

[4.2 Milestones 9](#_Toc5564838)

[4.3 Gantt Chart 11](#_Toc5564839)

[Chapter 5: Risk Management 13](#_Toc5564840)

[Chapter 6: Configuration Management 16](#_Toc5564841)

[Chapter 7: Conclusion 18](#_Toc5564842)

[Chapter 8: Reference and bibliography 19](#_Toc5564843)

**List of Figures**

[Figure 1: SDLC waterfall model 5](#_Toc5564453)

[Figure 2: MVC design pattern 6](#_Toc5564454)

[Figure 3: Standalone Computer architecture 7](#_Toc5564455)

[Figure 4: Work Breakdown Structure 8](#_Toc5564456)

[Figure 5: Gantt chart 1 11](#_Toc5564457)

[Figure 6: Gantt chart 2 12](#_Toc5564458)

[Figure 7: Configuration Management 17](#_Toc5564459)

**List of Tables**

[Table 1: Milestones table 9](#_Toc5564441)

[Table 2: Likelihood Table 14](#_Toc5564442)

[Table 3: Consequences Table 14](#_Toc5564443)

[Table 4: Impact table 15](#_Toc5564444)

# Chapter 1: Introduction

## 1.1 Introduction to the project

My project is a real-time implementation of a stock control system for an on-site small liquor shop. As the liquor shop is booming this system will offer the user an easier, faster and convenient way to control stock of various types of liquors and store information about day to day transactions. The system will also allow the user to create bills of a transaction and keep records of the stock. The will provide dynamic stock adjustment for each item sold. The system will provide information regarding stock levels and their threshold. If the stock is below threshold level this system will provide information accordingly.

## 1.2 Background of the project

Currently, the shop buys products from various suppliers. All the stock information and transaction between suppliers are kept records manually by the owner. The daily sales are recorded on handwritten bills. To keep the track of the stock levels and sales they have to calculate day to day of how much the item is sold. This process takes up a lot of time and human effort. It is also prone to human error.

## 1.3 Problem statement

The situation poses a problem of a situation that the shopkeeper at liquor shop as well as many other retail shops faces. To keep manually track of sales and stocks and place correct orders to the customer takes up a lot of time. This system will assist tackling the above problem as this system would help remove unnecessary error susceptible to complications and allow convert the unproductive time to something more useful. The system will provide an easier and electronic way to store information regarding stock levels and sales. Therefore, from this project, the client needs will be fulfilled and control the stock will be more efficient and effective.

## 1.4 Description of the project

The system will provide an effective way for the client to store shop’s transaction between their customers and stock control. The system will help client to add stocks and dynamically control stock according to sales. It will provide billing system with loyalty service for customer. The loyalty feature allows customer to get points according to their spending and customer can get discount according to their points. The system will also provide notification according to stock levels.

## 1.4.1 Features of the project

* Pin login feature (initial configuration instead of registration).
* Client can add, edit and remove category of liquor.
* Client can add, edit and remove stocks.
* Client can create bill for daily transaction.
* Stock dynamically changes according to sales.
* Loyalty discount feature for customer.
* Notification of stock level according to threshold.
* Report based on stock sales.

## 1.5 Overview of the project

The main aim of the project is to develop a desktop based application for the client to record and perform stock management process and billing and record transaction of the shop. The system will be time and cost efficient, easy to use and allow the electronic storage of data. It will also provide an additional feature of loyalty discount service for customer and stock level notification so that the shop will have sufficient stocks to meet customer needs. Measuring the change in stock will allow the store to plan for future inventory needs.

# Chapter 2: Scope

## 2.1 Scope

The scope of this project will primarily focus on liquor incoming and outgoing stocks (measuring change in stock) and their transaction information of the shop. Measuring the change in inventory will allow the store to determine the cost of the stock sold during the period.

## 2.2 Limitation

* Desktop based for a stand-alone computer.
* System will not support web service facilities.
* System is not built for a large liquor store.
* System will not have future updates and patches. (single time installation)
* Does not provide the function of planning future stock levels.
* Does not cover unit that is damaged, stolen or scrapped.

## 2.3 Aims

* To build a desktop database system for managing the liquor stocks based on each item sold.
* To automate the liquor stock management process in the liquor shop.
* To automate the billing process in the liquor shop.

## 2.4 Objectives

* To analyze and check whether the project is feasible or not for the shop.
* To make the system user-friendly and easy to learn
* To store the information of stocks in the database systematically.
* To make a secure system. (Pin login system)
* To allow a user to control the stock level.
* To help a user know the stock level to allow future stock prediction and planning.
* To show reports according to sales of the stock.
* To ensure the availability of the stock for meeting consumer demand.
* To provide billing service with a discount feature.
* To test the system to check bugs or errors.

## 2.5 Overview of the scope

The main aim of the project is to develop a desktop application for the client of liquor shop to automate the day to day liquor stock management, sales and customer billing process. The limitation of using a stand-alone system is certain but the automated stock system makes the process easier, simpler, faster and more efficient than manually recording the information.

# Chapter 3: Development Methodology

## 3.1 Development Methodology

The project will be developed using waterfall methodology. In the waterfall methodology, the whole process of system development is divided into different phases. This methodology allows easier planning, cost-efficient and takes less time for development. The step by step phase works well for the small projects as the requirements are well understood. In the waterfall, the task is easy to arrange and milestones are well understood. Each phase has specific deliverables and a review process.

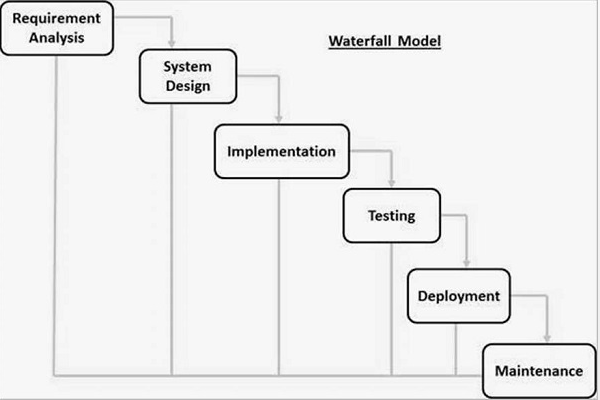


Figure 1: SDLC waterfall model

The major drawback of this methodology is that it is very demanding to go back once the phase is completed. This methodology is not suitable for large and ongoing projects where the requirement is uncertain or keep changing. Since this project is small and has a clear and fixed requirement waterfall method is suitable for the development of this project.

## 3.2 design pattern

This system will be developed using object-oriented modeling methodology. C# .net framework will be used to develop the system. For this project, I will be using the Model View Controller (MVC) design pattern will be implemented.

This design pattern focuses on diving application into three different tiers i.e. (model) data model, (view) presentation information and (controller) control information.

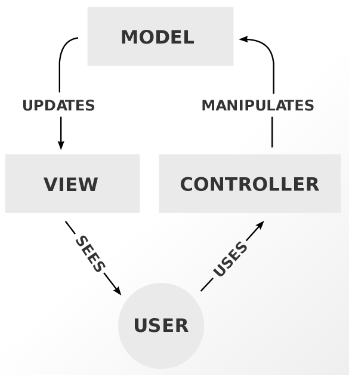


Figure 2: MVC design pattern

In MVC design pattern the separation of concern allows maintaining the code is easier without making changes to the entire system. This is the major advantage of using the MVC design pattern.

## 3.3 Architecture

This project will fully be a stand-alone network-based application as the system will be built for a small liquor store. Stand-alone network or computer means a computer that stands on its own. The computer is not connected to the network and any data associated with the computer are not accessible to others. The system will only require a single computer and does not require any network or internet connection to operate. The application authorizes the user through pin code. The simplicity of the standalone is the major advantage of this architecture. It is a lot easier to manage a single computer. The store is small so stand-alone computer system will be more cost efficient.

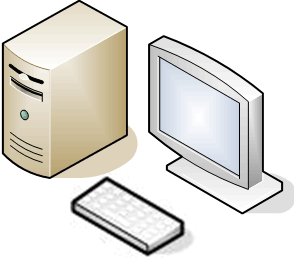


Figure 3: Standalone Computer architecture

# Chapter 4: Project Planning

## 4.1 Work Breakdown Structure (WBS)

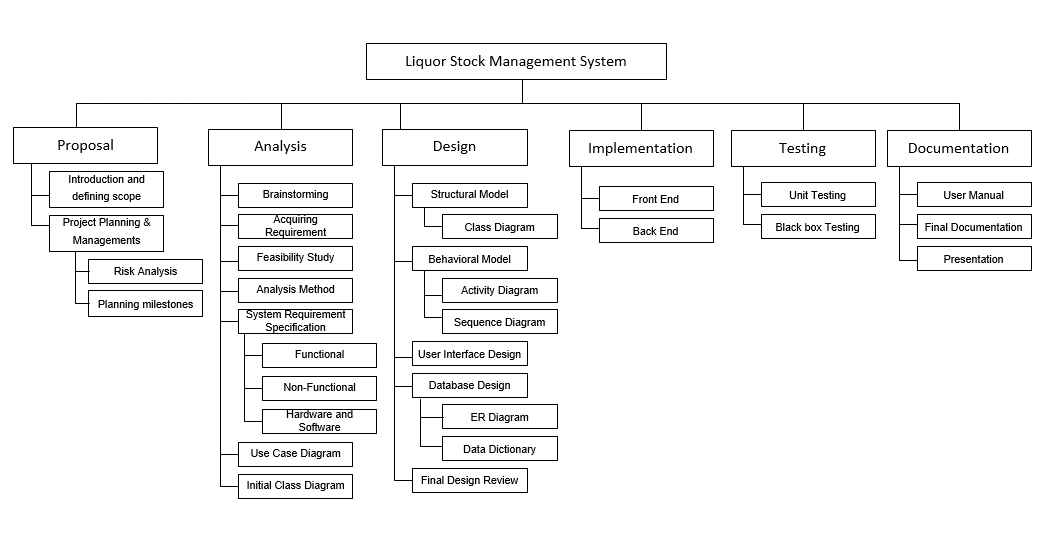


Figure 4: Work Breakdown Structure

## 4.2 Milestones

|  |  |  |  |
| --- | --- | --- | --- |
| **Task Number** | **Task Name** | **Deadline** | **No of Days** |
| **1** | **Project proposal** | **9th April 2019** | **16** |
| **2** | **Analysis** | **8th May 2019** | **29** |
| 2.1 | Brainstorming | 14th April 2019 | 5 |
| 2.2 | Acquiring Requirement | 19th April 2019 | 5 |
| 2.3 | Feasibility Study | 24th April 2019 | 5 |
| 2.4 | Analysis Method | 29th April 2019 | 5 |
| 2.5 | System Requirement Specification | 3st May 2019 | 4 |
| 2.6 | Use Case Diagram | 5th May 2019 | 2 |
| 2.7 | Initial Class Diagram | 8th May 2019 | 3 |
| **3** | **Design** | **3rd June 2019** | **26** |
| 3.1 | Structural Model | 13th May 2019 | 5 |
| 3.1.1 | Class Diagram | 13th May 2019 | 5 |
| 3.2 | Behavioral Model | 21st May 2019 | 8 |
| 3.2.1 | Activity Diagram | 17th May 2019 | 4 |
| 3.2.2 | Sequence Diagram | 21st May 2019 | 4 |
| 3.3 | User Interface Design | 25th May 2019 | 4 |
| 3.4 | Database Design | 31th May 2019 | 6 |
| 3.4.1 | ER Diagram | 28th May 2019 | 3 |
| 3.4.2 | Data Dictionary | 31st May 2019 | 3 |
| 3.5 | Final Design Review | 3rd June 2019 | 3 |
| **4** | **Implementation** | **24th June 2019** | **21** |
| 4.1 | Front End | 13th June 2019 | 10 |
| 4.2 | Back End | 24th June 2019 | 11 |
| **5** | **Testing** | **1st July 2019** | **7** |
| 5.1 | Unit Testing | 29th June 2019 | 5 |
| 5.2 | Black box Testing | 1st July 2019 | 2 |
| **6** | **Documentation** | **12th July 2019** | **11** |
| 6.1 | User Manual | 3rd July 2019 | 2 |
| 6.2 | Final Documentation | 8th July 2019 | 5 |
| 6.3 | Presentation | 12th July 2019 | 4 |

Table 1: Milestones table

The other project will be broken down to 6 different tasks i.e. Proposal, Analysis, Design, Coding, Testing, and Documentation. Each of the tasks contains subtasks which will be done during the main task duration.

Proposal consist of an introduction of the project and knowing the scope and problem domain. Another subtask of the proposal is the planning and management of the project. It includes risk management and configuration management. Knowing the scope and problem domain takes longer time so time is distributed accordingly.

Analysis consist of brainstorming, acquiring the requirements, feasibility study, analysis of methodology, creating SRS, use case diagram and initial class diagram. Time separation are equally balanced for analysis as every individual subtask are important.

Design consist of structural, behavioral and database model. Behavioral model have higher priority as it deals with the business logic related to the liquor store. The system will not any future update facility so the system process should be in the level of perfection. At last the review of design will be made to ensure the design is according to analyzed information.

Implementation consist of front end and back end coding. Both tasks are equally distributed time.

Testing consist of unit testing and black box testing. Unit testing is giving priority as the project will be developed using object-oriented modeling methodology.

At last documentation of the project with user manual and presentation of the project to show the results and how the project is developed.

## 4.3 Gantt Chart

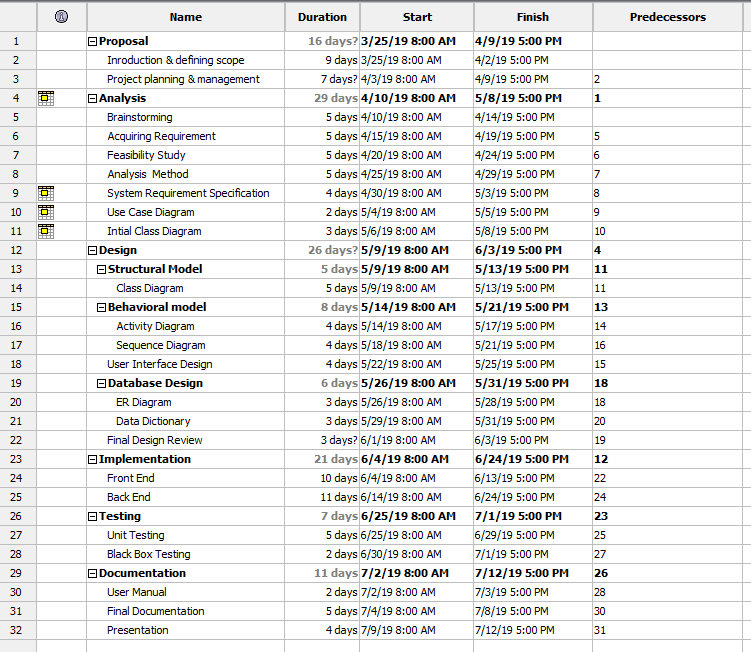


Figure 5: Gantt chart 1

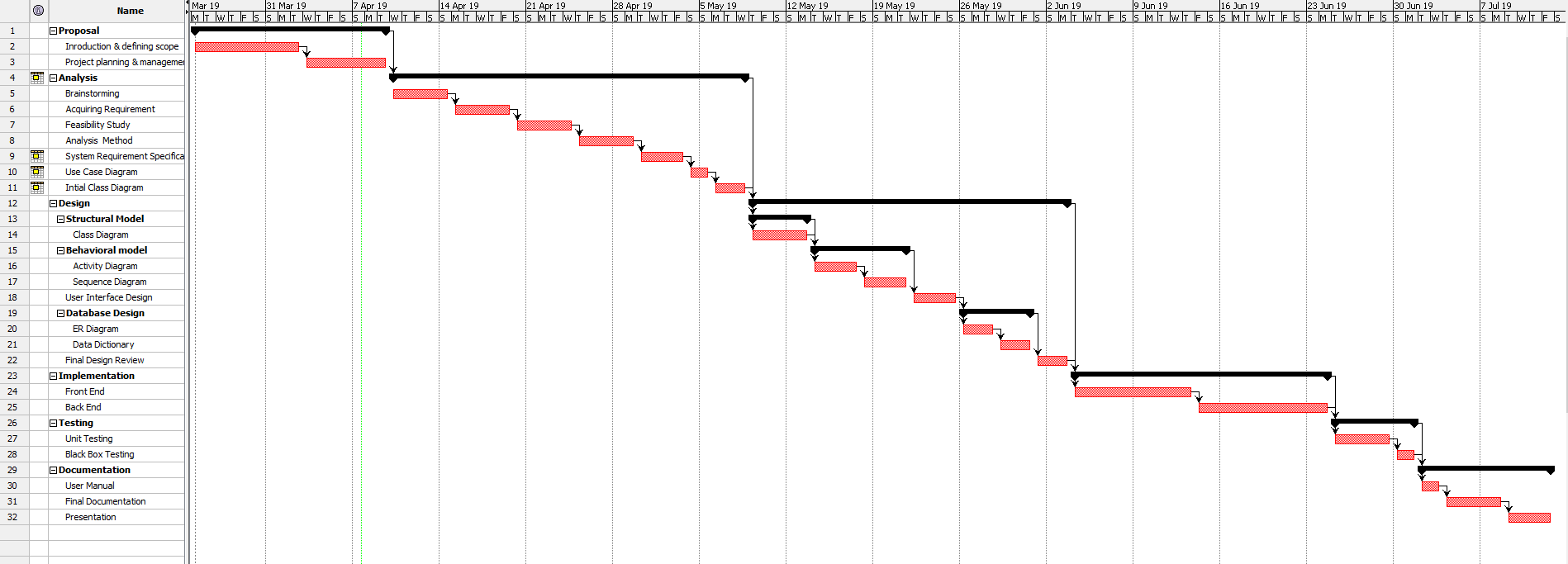
****

Figure 6: Gantt chart 2

# Chapter 5: Risk Management

The process to identify and control the threats and risks that can harm the project with the appropriate solution is called risk management. Every project has its own risk and flaws. Therefore in order to minimize the threats and risks risk management is important. The steps of risk managements are as follows:

**Step 1: Identify the risk**

Every possible risk are recognized and described.

**Step 2: Analyze the risk**

The likelihood and consequences of identified risk are determined.

**Step 3: Evaluate the risk**

The impact is determined and ranked with the help of likelihood and consequences.

**Step 4: Treat the risk**

The solution is created to minimize the risk and their casualties.

**List of Risks**

* Hard disk crash
* Data theft
* Load shedding
* Software crash
* Security Loopholes
* Change in government policies
* Natural disaster
* Infeasible Design

**Value of Likelihood**

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

Table 2: Likelihood Table

**Value of Consequences**

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

Table 3: Consequences Table

**Impact**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S.N. | Risk | Likelihood | Consequences | Impact | Action |
| 1 | Hard disk crash | 1 | 5 | 5 | Maintaining weekly or monthly backup. |
| 2 | Data theft | 1 | 5 | 5 | High security for the hardware location. |
| 3 | Load shedding | 3 | 4 | 12 | Installing backup power supply (i.e. UPS) |
| 4 | Software crash | 2 | 3 | 6 | Maintaining backup of data and installation files (to reinstall if needed). |
| 5 | Security Loopholes | 2 | 4 | 8 | Updating operating system and security tools. |
| 6 | Change in government policies | 1 | 5 | 5 | Monitoring the changes in the policies and implementing the changes if needed. |
| 7 | Natural disaster | 1 | 5 | 5 | Keeping backup in a different location or a device such as portable hard drive. |
| 8 | Infeasible Design | 1 | 5 | 5 | Proper analyzation of requirement and design accordingly |

Table 4: Impact table

# Chapter 6: Configuration Management

Software Configuration management is the discipline which is consist of processes and techniques to manage the changes made to the software project. It helps to identify the individual elements and configuration, tracks the changes and version selection and control. It is an efficient method to track all the folders and files to reduce redundant work and prevent any future anomalies and manage effectively.

There are different stages of configuration management. They are as follows:

Step 1: Build Internal Structure

Step 2: Configuration Identification

Step 3: Configuration Control

Step 4: Configuration status accounting

Step 5: Configuration verification and audit

For this project, I have configured my files according to the work breakdown structure. I have created a directory and subdirectory for each different topics and models where files are kept accordingly. There is also a backup directory that will contain the backup of all the directories and files. The backup directory is updated every time any changes are made in the main files or directories.

I am also planning to use GitHub for project management. It is a web-based hosting service that allows up to back up all our files.

GitHub clone link: https://github.com/Ajar-maharjan/CP.git

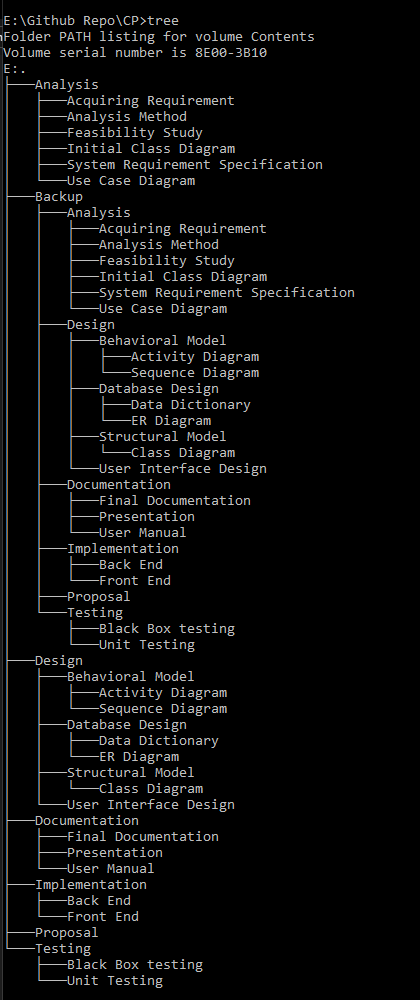
****

Figure 7: Configuration Management

# Chapter 7: Conclusion

Therefore, the project I purpose shall provide a huge upgrade for manually recording stock and sales information to automate the process. It will make the daily process easier, faster, convenient and time and manpower efficient for managing the sales and stock levels. The project will prevent wastage of useful labor and help to convert unproductive time to something more useful by removing unnecessary errors.

The project will have about 110 days for completion. I will be using a waterfall methodology for the development of the project. The whole project is divided into different parts and models. Analyze, design, implement and testing and documentation will be major concerns for the project. Time estimation and milestones, risk management and configuration management were done to make the development process more efficient. Hence I would like to conclude this proposal and wait for the approval of the project to start as soon as possible.

# Chapter 8: Reference and bibliography

Sheffield.ac.uk. (2019). [online] Available at: https://www.sheffield.ac.uk/polopoly\_fs/1.440722!/file/HowtoWriteaProblemStatement.pdf [Accessed 1 Apr. 2019].

Bizfluent.com. (2019). [online] Available at: https://bizfluent.com/info-8624995-scope-limitations-inventory-systems.html [Accessed 3 Apr. 2019].

Your Article Library. (2019). Inventory Control: it’s Objectives, Advantages and Limitations. [online] Available at: http://www.yourarticlelibrary.com/inventory-control/inventory-control-its-objectives-advantages-and-limitations/27944 [Accessed 3 Apr. 2019].

www.tutorialspoint.com. (2019). SDLC Waterfall Model. [online] Available at: https://www.tutorialspoint.com/sdlc/sdlc\_waterfall\_model.htm [Accessed 5 Apr. 2019].

C-sharpcorner.com. (2019). Advantage and Disadvantage of ASP.NET MVC. [online] Available at: https://www.c-sharpcorner.com/blogs/advantage-and-disadvantage-of-asp-net-mvc1 [Accessed 6 Apr. 2019].

Techopedia.com. (2019). What is Software Configuration Management (SCM)? - Definition from Techopedia. [online] Available at: https://www.techopedia.com/definition/24583/software-configuration-management-scm [Accessed 7 Apr. 2019].