April 9, 2019

Sanim Khadgi

00175034/ L5DC

Submitted to : Niman Maharjan

**Retail inventory**

& point of sale system

Contents

[1 INTRODUCTION 3](#_Toc5669553)

[1.1 Project Introduction 3](#_Toc5669554)

[1.2 Justification for Project 3](#_Toc5669555)

[1.2.1 Bacground of the project 3](#_Toc5669556)

[1.2.2 Problem Statement 3](#_Toc5669557)

[1.3 Description of the project 3](#_Toc5669558)

[1.3.1 Features of the system 3](#_Toc5669559)

[2 Project Scope 4](#_Toc5669560)

[2.1 Scope and Limitaion of the Project 4](#_Toc5669561)

[2.2 Aims and Objectives 4](#_Toc5669562)

[3 Development methodology 5](#_Toc5669563)

[3.1 Methodology Used 5](#_Toc5669564)

[3.2 Design Pattern 5](#_Toc5669565)

[3.3 System Architecture 6](#_Toc5669566)

[4 Scheduling 7](#_Toc5669567)

[4.1 Work Breakdown Structure 7](#_Toc5669568)

[4.2 Milestone 8](#_Toc5669569)

[4.3 Scheduling 9](#_Toc5669570)

[4.3.1 Time Estimated Table 9](#_Toc5669571)

[4.3.2 Grantt chart 9](#_Toc5669572)

[5 Risk Management 10](#_Toc5669573)

[6 Configuration Management 11](#_Toc5669574)

[7 Conclusion 12](#_Toc5669575)

[8 References 12](#_Toc5669576)

[Figure 1Waterfall Model 5](#_Toc5669601)

[Figure 2 MVC design pattern 6](#_Toc5669602)

[Figure 3 System Architecture 7](#_Toc5669603)

[Figure 4 Time estimation 9](#_Toc5669604)

[Figure 5 Grantt Chart 9](#_Toc5669605)

[Figure 6 Directory of the project 11](#_Toc5669606)

[Figure 7 Repository 12](#_Toc5669607)

# **INTRODUCTION**

## Project Introduction

In todays scenario, the world we live in is getting smarter each day. The way we do and precive things have transformed into more practical and easier route. From trading goods as a method for payment to using paper currency and now turning paper currency into digital wallets. To assist todays improving transaction procedure this system will keep track of user payment and will accept both cash and digital payment method from their own devices. The retail shop in this theme accpets inventory, analyzes daily profit and provide a reciept to the customer to validate purchase of a product.

## Justification for Project

### Bacground of the project

The core task of this project is to create a seamless enviroment between customer and retailer. The retailer can preview every purchase history and collect information to compare for future references. The customer are given an invoice of the purchase with valid signatures.

The retail unit will have multiple user. Incase of absence of one employee additional employee can be added and begin the process. The retailer will act as a system admin for the project with top level control. As this project is aimed to be lightwieght and attractive it will be developed using Personal Home Page(PHP).

### Problem Statement

Using paper based journals and ledgers for storing transaction is uncertain. Records will not be properly stored and retrived at a given time period. Flippping through all the transaction of a journal is not the most time effiecient or an effective process. The chances of informatioin being manipulated and stolen is very high. Handing out handwritten invoices can include human errors as well. Identifying which employee performed the transaction is also an issue in case of a fatal error.

To solve these and various other problems in the business my software will assist the organization. It will create a seamless interaction of stock information. A printed out or digital invoice will be generated with each purchase transaction. To store all the information a new database system will be installed in the retail department.

## Description of the project

### Features of the system

The system will elevate the experience of users through these fundamentat elements included in the system.

1. Record invertory details

After a customer purchases an item the time,paid amount, cashier name will be stored in database securely. Incase of returned product or damaged product the item will be replaced after validating records.

1. Paper or digital Invoices

Customers are handed out printed out invoices will all the transaction and a copy of the transaction is store in database for future comparisons.

1. Hirerachical control

Owner can preview employee presence, amount of transaction made in a paticular day with top level control.

1. Self explanatory UI

UI will include skeuomorphic design to the icons help user to interact easily rather than search for particular menu without hints or suggestions.

1. Calculate transaction details per session

All the transaction done by an employee in a session will be logged and calculated then stored in the database.

1. Pay through QR code

Users will be able to scan and pay for their purchases scanning our QR code. It redirects customers to a digital payment gateway.

# Project Scope

## Scope and Limitaion of the Project

This project is determined to focus on developing a seamless data storing and retriving experience to the involved stakeholders. The internal delivariables of the project include data protection and data efficieny for the retail shop and as for the external delivaribles, the customers will have a hassle free experince paying and finding out details about their transactions. The payment is done either with paper currency or by digital wallet.

However the project has some minor limitations. As the project is server based, failure of certain server hardware can halt the operation. Employees may have issues understanding components from the rapid shift of paper journals and ledgers to digital system.

## Aims and Objectives

Overall my agenda of the project is to transform from the current paper based system to a more modern and efficient system. Creation of a data hirerachy and safe data storage is a must. Shift the payment method into digital methods to decrease chaces of possible losses. Implementing a QR code gateway for payment would be a personal challenge to explore and conquest.

Obejctives of the project are as follow

* Development of a database with hirerachial flow.
* Preview transaction details at a glance.
* Track record with search functionality.
* Construct a skeumorphic UI design.

# Development methodology

## Methodology Used

To develop the program I have preferred Waterfall model. As it is a linear approach to development life cycle. The sequential nature of waterfall model provides checkpoint during the development process. Goals are set between each milestones and are besy suited in condition of having fixed resources with a timline. (Rouse,2019)

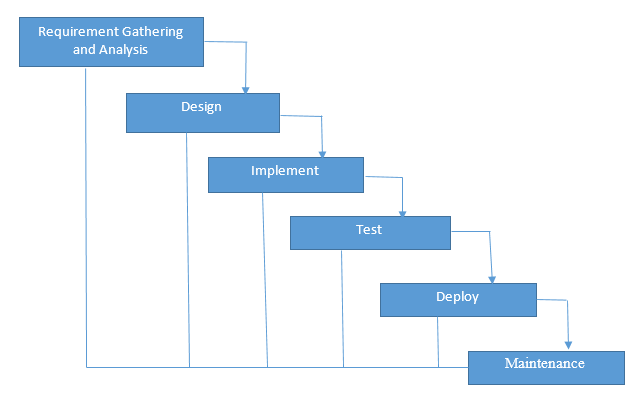


Figure 1Waterfall Model

## Design Pattern

A design pattern is a recursive solution to resolve software engineering problem. Design patterns are used in almost every program.It is not a program in itself but a stencil to develop over. Swift development can be achieved when using a prototype and efficiency of code can be drastically increased.(Techopedia,2019)

For the project I have preffered to use Model View Controller (MVC) pattern as for the development of the project. As it inspires development of a modular and dynamic system and allows me to do switfly add,edit,delete and futhermore functionality.

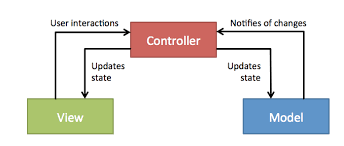


Figure 2 MVC design pattern

**Model**:

Model is the name given to the permanent storage of the data used in the overall design. It must allow access for the data to be viewed, or collected and written to, and is the bridge between the View component and the Controller component in the overall pattern.

**View**:

View is where data, requested from the Model, is viewed and its final output is determined. Traditionally in web apps built using MVC, the View is the part of the system where the HTML is generated and displayed. The View also ignites reactions from the user, who then goes on to interact with the Controller.

**Controller**:

Controller handles data that the user inputs or submits, and update the Model accordingly. The Controller’s life blood is the user; without user interactions, the Controller has no purpose. It is the only part of the pattern the user should be interacting with.

## System Architecture

System architecture activities is to define a comprehensive solution based on principles, concepts, and properties logically related and consistent with each other. The solution architecture has features, properties, and characteristics satisfying, as far as possible, the problem or opportunity expressed by a set of system requirements (traceable to mission/business and stakeholder requirements).(Willer,2019).

I selected 3 tier system architecture as It provide great flexibility during development. This integration flexibility also makes it ideal for embedding analytics software into pre-existing applications and is often used by embedded analytics vendors for this reason.

**3Tier System includes**

**Presentaion Tier:** The presentation tier is the front end layer in the 3-tier system and consists of the user interface. This user interface is often a graphical one accessible through a web browser or web-based application and which displays content and information useful to an end user**.**

**Application Layer:** The application tier contains the functional business logic which drives an application’s core capabilities.

**Data Layer:** The data tier comprises of the database/data storage system and data access layer.

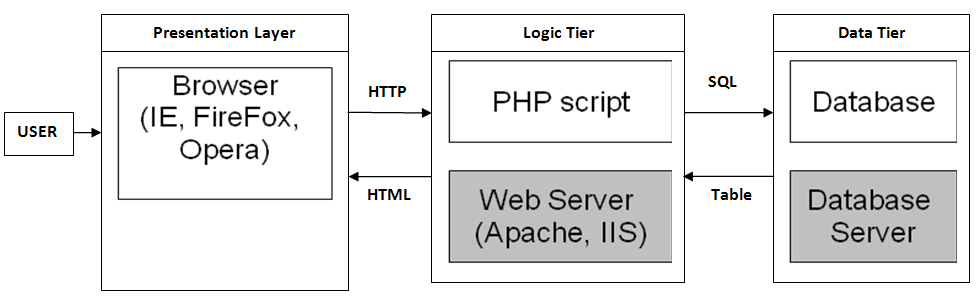


Figure 3 System Architecture

# Scheduling

## Work Breakdown Structure

A work breakdown structure (WBS) is a chart in which the critical work elements, called tasks, of a project are illustrated to portray their relationships to each other and to the project as a whole. The graphical nature of the WBS can help a project manager predict outcomes based on various scenarios, which can ensure that optimum decisions are made about whether or not to adopt suggested procedures or changes.(Chapman,2019).

**Retail invetory and point of sale system**

Implementation

Testing

Final Document

Design

Analysis

Proposal

Feedback

Validation

Assembling Requirement

Database Development

UI Design

Confuguration Management

Verification

Database structure

Planning

Risk Management

Code

Adapting skeumorphism design

Criteria analysis

WBS

## Milestone

I have created milestones in order to mark project tassk. By providing milestones it distributes the project lifecyle to view.

|  |  |
| --- | --- |
| **Milestones** | **Date** |
| **Proposal** | **(2019-03-25 to 2019-04-09) Total =16 Days** |
| Configuration Management  Risk Management  WBS | 03/25 to 03/29 |
| 03/29 to 04/02 |
| 04/02 to 04/08 |
| **Analysis** | **(2019-04-10 to 2019-05-08) Total=28 Days** |
| Assembling Requirement  Planning  Criteria Analysis | 04/10 to 04/14 |
| 04/14 to 04/29 |
| 04/29 to 05/08 |
| **Design** | **(2019-05-09 to 2019-06-03) Total=25 Days** |
| UI Design  Database Structure  Adapting skeumorphism design | 05/10 to 05/20 |
| 05/20 to05/28 |
| 05/28 to 06/03 |
| **Implemntation** | **(2019-06-04 to2019-06-24)T Total=20 Days** |
| Code  Database Development | 06/04 to 06/15 |
| 06/15 to 06/24 |
| **Testing** | **(2019-06-25 to 2019-07-01)Total=6 Days** |
| Validation  Verification | 06/25 to 06/28 |
| 06/28 to 07-01 |
| **Final Documentation** | **(2019-07-02 to 2019-07-12)Total=11 Days** |
| Feedback  Complete submission | 07/02 to 07/04 |
| 07/04 to 07/12 |

## Scheduling

### Time Estimated Table

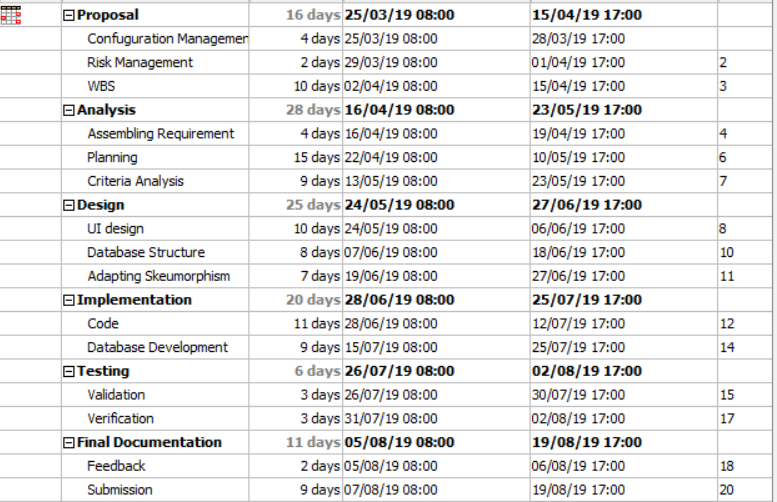


Figure 4 Time estimation

### Grantt chart

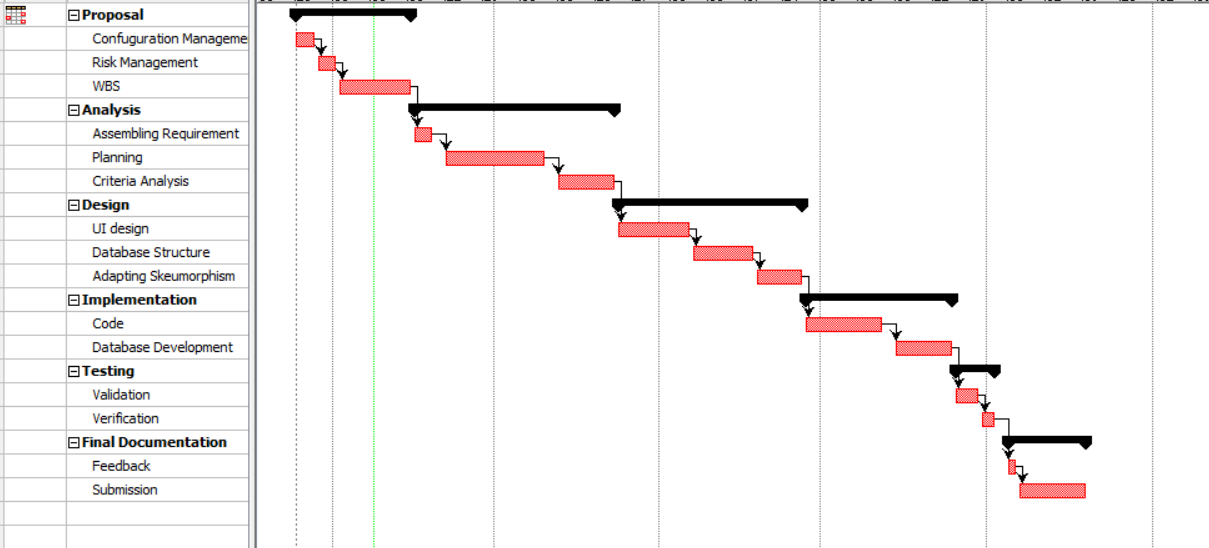


Figure 5 Grantt Chart

# Risk Management

Identifying possible threats and mitgating those threats is general description of risk management. It is usually caused due to lack of control, time and information. To futher reduce the impact of the risk that can prevail risk management is allotted to reduce the effects on the software.

Risk management is done through:

* Identifying the risk
* Reducing the impact of risk
* Reducing the probablitiy or likelihood of risk
* Risk monitoring

Here, Risk has been calculated using the formula

**(Impact = Likelihood \* Consequences)** where, Risk Likelihood value is given as:

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

And Risk Consequences value is given as:

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

The risk that may consequence my project are identified below with their impact and possible solutions:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **s.no** | **Risks** | **Likelihood** | **Consequences** | **Impact** | **Solution** |
| 1. | Failure of the system | 2 | 5 | 10 | Timely maintainance and installation of backup power can reduce the chances. |
| 2. | Design Faliure | 3 | 3 | 9 | Clear communication and clear requirements should be delivered |
| 3. | Requirement Inflation | 1 | 5 | 5 | The feautres should be clearly identified at the beginning of project development. |
| 4. | Lack of resources | 2 | 3 | 6 | Identifying the budget and human resources will help in management of resouces accurately |
| 5. | Obsolance | 2 | 4 | 8 | As the need shift toward next phase the product may become obsolete unavoidably. |
| 6. | Time estimation | 1 | 4 | 4 | Calculation of specific pre assigned task should be noted to create a project checkpoint. |

# Configuration Management

Configuration management declares the items in the system, records changes throught the development lifecycle , reports the changes and verifies for the software completeness. Without the inclusion of a configuration management system the source code woukd become fragmented and unorganized. Development process halts without proper knowledge of the key parts of the application.

I have used git bash as my choice for the conifguration tool to allow me the keep proper track of my project development.

Git id: 48986073 available at : https://github.com/SanimKhadgi

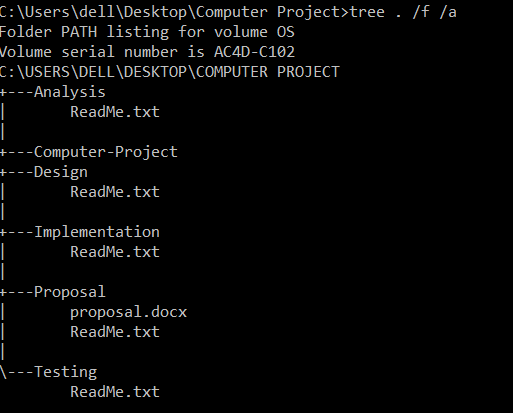


Figure 6 Directory of the project

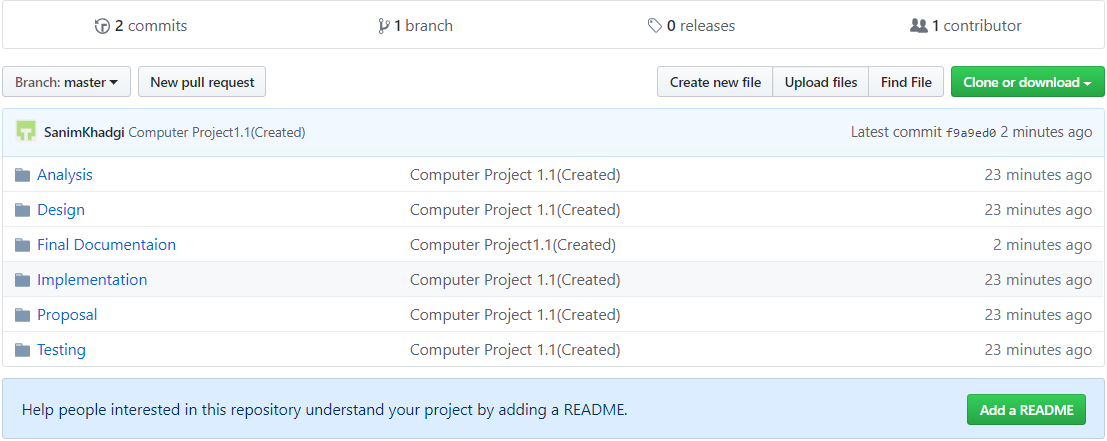


Figure 7 Repository

# Conclusion

Therby, the main goal of the project has been to create a seamless and interacting method of inventory recording and paying for the inventory with a modern redesign. The program will handle multiple actions and will have a hirerachial level of data accesibilty to ensure maximum security.

# References

Design pattern

System architecture

Work breakdown structure