Project Proposal

On

**<<Project Name>>**

<<Your Name>>

<<NCC ID>>

Computing Project

Level 5 Diploma in Computing

Softwarica College of IT and E-Commerce

Kathmandu , Nepal

<<Date of Submission>>

Submitted to: Kiran Rana

**Contents**

1. Introduction……………………………………………..…..2

1.1 Project Introduction…………………………………..…..2

1.2 Justification for project…………………………………..….3

1.2.1 Background of the project……………………..….3

1.2.2 Problem statement………………………..……4

1.3 Description of the project…………………….…..….6

1.3.1 Features…………………………………..….6

2. Project Scope….………………………………………….…….7

2.1 Scope and Limitation of the project……………….……7

2.2 Aims and Objectives…………………………….…...8

3. Development Methodology………………………….……9

3.1 Methodology used……………………………….…...9

3.2 Design Pattern………………………………………….10

3.3 System Architecture………..…………………………..11

4. Work Breakdown Structure (WBS) / Scheduling……………...12

4.1 Work Breakdown Structure…………………………….13

4.2 Milestones…………………………………………13

4.2 Scheduling / Gantt Chart…………………………….14

5. Risk Management……………………………………………...15

6. Configuration Management……………………………………17

7. Conclusion of the project………………………………………19

8. References…………………….………………………….…….21

**List of figures**

Figure 1 <<name of the figure>>…………………………………………...6

Figure 2 <<name of the figure>>……………………………………………10

Figure 3 <<name of the figure>>……………………………………………..

Figure 4 <<name of the figure>>……………………………………………..

Figure 5 <<name of the figure>>……………………………………………..

Figure 6 <<name of the figure>>…………………………………………page no

# **1. Introduction**

## **1.1 Project Introduction**

In government institution the terms ‘Darta-Chalani’ is one of the most necessary part of the whole institution. This act as main gateway that officially registered and provide the number system for any outgoing and incoming corresponding letters. The terms ‘Darta-Chalani’ is abbreviation for the registering and provide the number for any incoming letters and dispatching with the unique number for any outgoing letters.

The ‘Darta-Chalani Online System’ helps to maintain the records digitally for all any incoming and outgoing correspondences which will reduces the lengthy jobs of handheld register books. By this institution will be benefits in search friendly that retrieval of data more smoothly and efficiently in less time.

## **1.2 Justification for project**

### **1.2.1 Background of the project**

The project, ‘Darta-Chalani Online System’ can be used for general records of all the correspondences of an organization in digital form. The system will able to replace the handheld register books and increase the efficiency of the organization.

**Primary focus**

The main purpose of the project is to replace hand held register book and increase the efficiency of organization and making the organization more accountability.

### **1.2.2 Problem Statement**

At the present time period all the corresponding of incoming and outgoing letters are maintained clerkly in hand held lengthy registers.

It takes much longer time to get information about the past records. All the records are handled by single man and absence of him no one can access those records. The project is the alternative to the handheld registered work and getting digitalized. The project aims to minimize the use of paper work.

## **1.3 Description of the project**

**1.3.1 Features**

The features of the project are as follows:

* Maintain digital records for all the correspondences letters
* Keep the copy of those correspondences in digital format
* The more user access can grant for all level users
* Reduce the time and focused clerk for searching any records
* Reduce ambiguous of records
* Consistent records and details.

# **2. Project Scope**

## **2.1 Scope and Limitation of project**

Scope

It is a good system that can be implement in government institutions to digitize the correspondences letters in efficient way. Data can be retrieved in small time and no pages needs to be turned. Bulky pages registers are long gone with the use of the system.

Limitation

The system can only keep the digital records of the correspondence letters only. This does not forward the letter to respective officers. Only the handheld register work is digitized. Computer knowledge is must to use the system.

## **2.2 Aims and Objectives**

**Aim**

Users are able to search and sort the relative fields correspondences letters for the relative project progress purposes. Also, replace the lengthy handheld registers books.

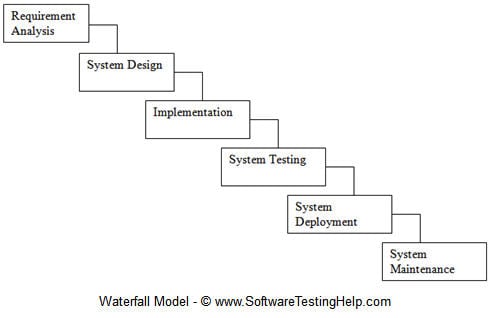
**Objectives**

* The system allows admin to dispatch and register the correspondences of officials letters.
* The system allows user to register in this system.
* The system generates the verification token through their email to verify the genuine user.
* The systems can reset the forgotten password through the registered email for verified users.
* The system allows users to view all the official records of dispatched and registered correspondence letter.
* The user can send the copy of digital records to their email.

# **3. Development Methodology**

## **3.1 Methodology used**

The ‘Darta-Chalani Online System’ will be accomplished by Waterfall Model method. Waterfall model considered the classic approach to the systems development life cycle which is very simple to understand and use. In this approach a schedule can be set with deadlines for each stage of development process. The phase in the development development process begins only if the previous phase is completed.

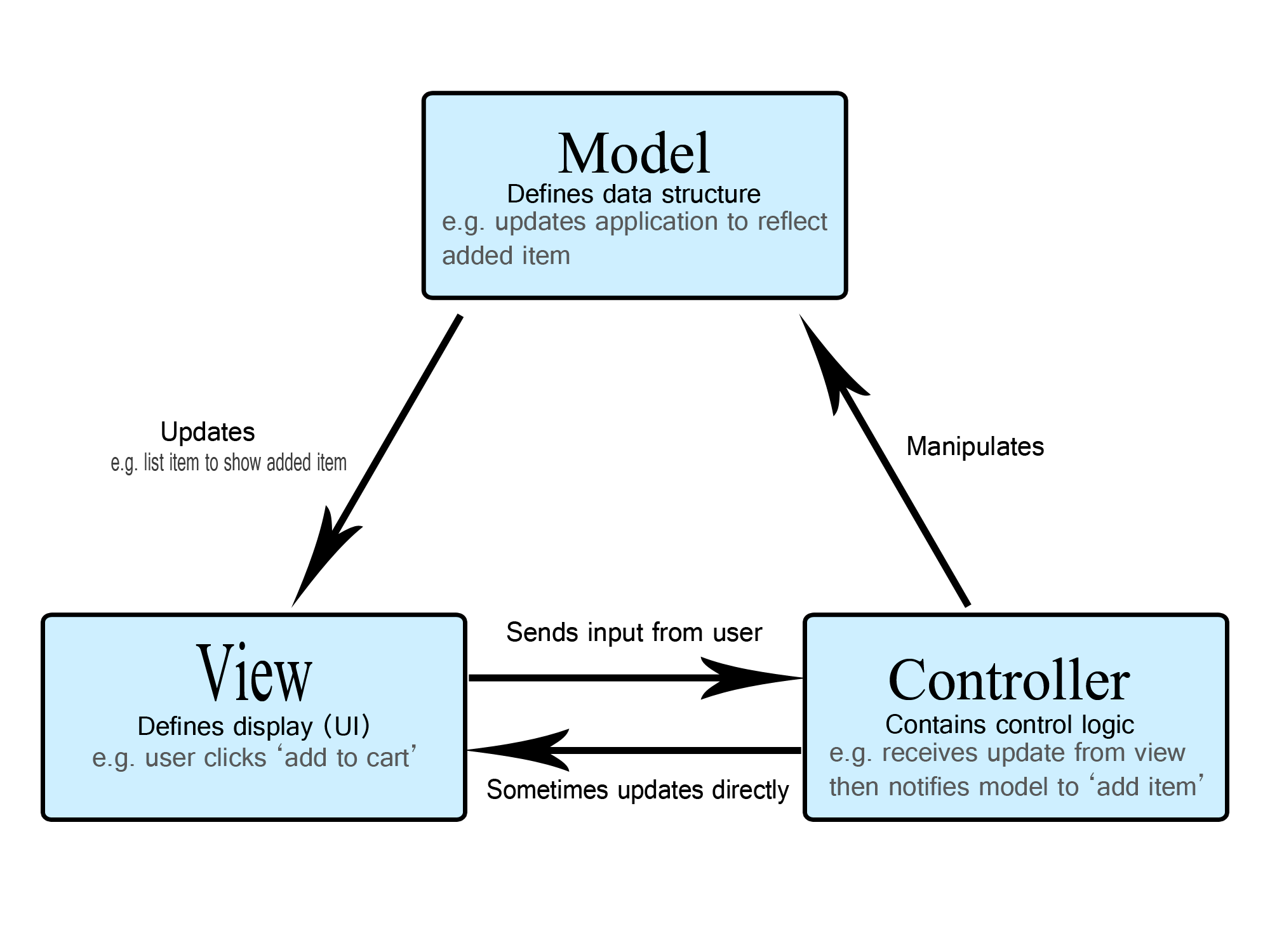


*fig. 1: Waterfall Model*

The waterfall development method had some restriction that it does not allow much reflection or revision. Once an application is in testing stage, it is very difficult to go back and change something that was not well-thought out in the concept stage.

## **3.2 Design Pattern**

The ‘Darta-Chalani Online System’ development will be done based on MVC pattern. Model View Controller (MVC) is a software architecture pattern, commonly used to implement user interfaces while designing and developing the web application. In general, it separates out the application logic into three separate parts, promoting modularity and ease of collaboration and reuse.



*fig. 2: MVC diagram*

## **3.3 System Architecture**

By using MVC pattern a three-tier architecture will be implemented which is a client-server architecture. The functional process of this architecture is logic, data access, computer data storage and user interface which is developed and maintained as independent modules on separate platforms. By this the programing for a tier can be changed or relocated without affecting other tiers.

The three tiers architecture are:

1. Presentation Tier: The presentation tier is the front end layer that displays information related to service available on a website. This user interface is often a graphical one accessible through a web browser or web-based application and communicates with other tiers by sending results to the browser and other tiers in the network.
2. Application Tier: Also called the middle tier, logic tier, business logic or logic tier, this tier is pulled from the presentation tier. It controls application functionality by performing detailed processing.
3. Data Tier: Houses database servers where information is stored and retrieved. Data in this tier is kept independent of application servers or business logic.



*fig. 3: Three-tier Architecture*

# **4. Work Breakdown Structure (WBS) / Scheduling**

## **4.1 Work Breakdown Structure**

Work Breakdown Structure (WBS) is a key project achievement that organizes the team’s work into manageable sections. The WBS identify the major functional deliverables and subdividing those deliverables into smaller systems and sub-deliverables. It also plays a vital role in smoothness and continuous development of the project.

## **4.2 Milestones**

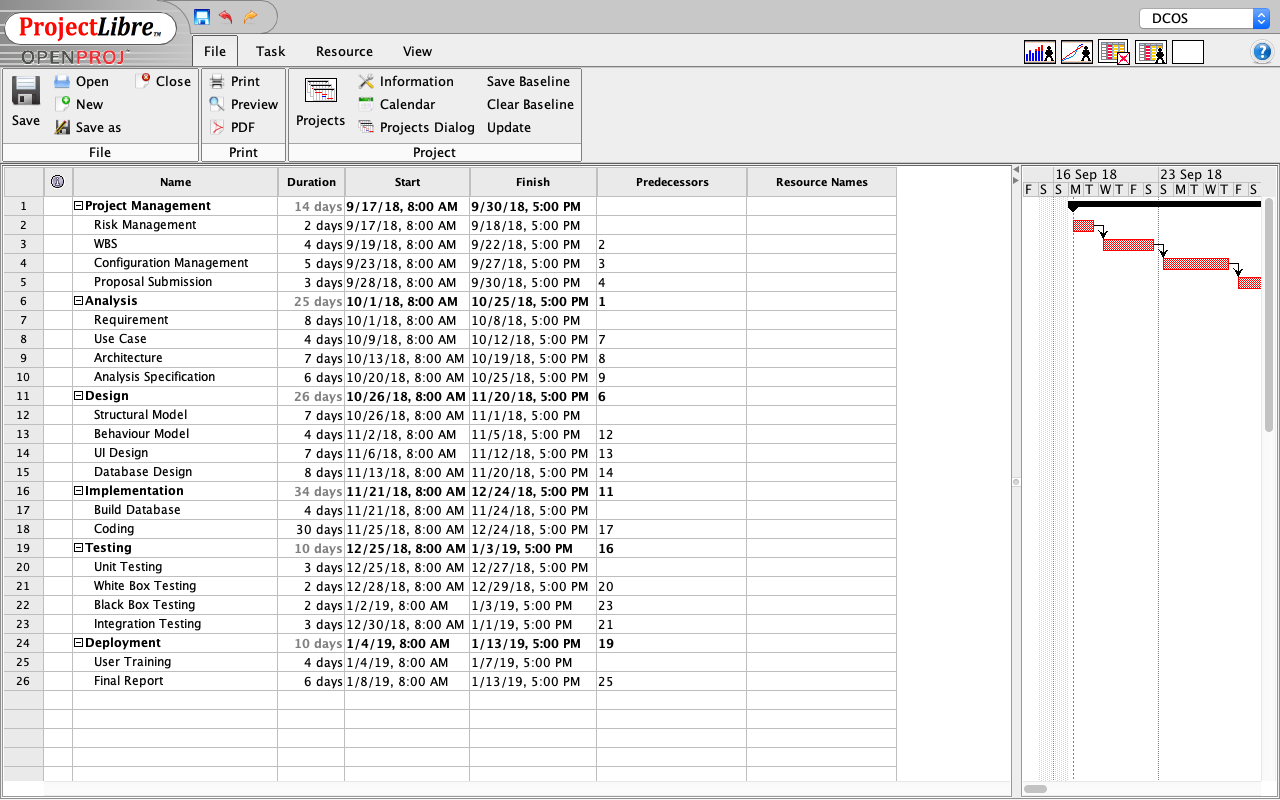
|  |  |
| --- | --- |
| **Milestones** | **Date** |
| **Project Management**  Risk Management  WBS  Configuration Management  Proposal Submission |  |
| **Analysis**  Feasibility Study  Requirement analysis  Planning  Use Case  Architecture ( Initial Class Diagram) |  |
| **Design**  Structural Diagram  Behavioral Diagram  UI Design  Database Design (ER , Data Dictionary) |  |
| **Implementation**  Building Database  Coding |  |
| **Testing**  Unit Testing  Integration Testing  Blackbox Testing  Whitebox Testing |  |
| **Deployment**  User Training  Final Report |  |

**Description of Milestones:**

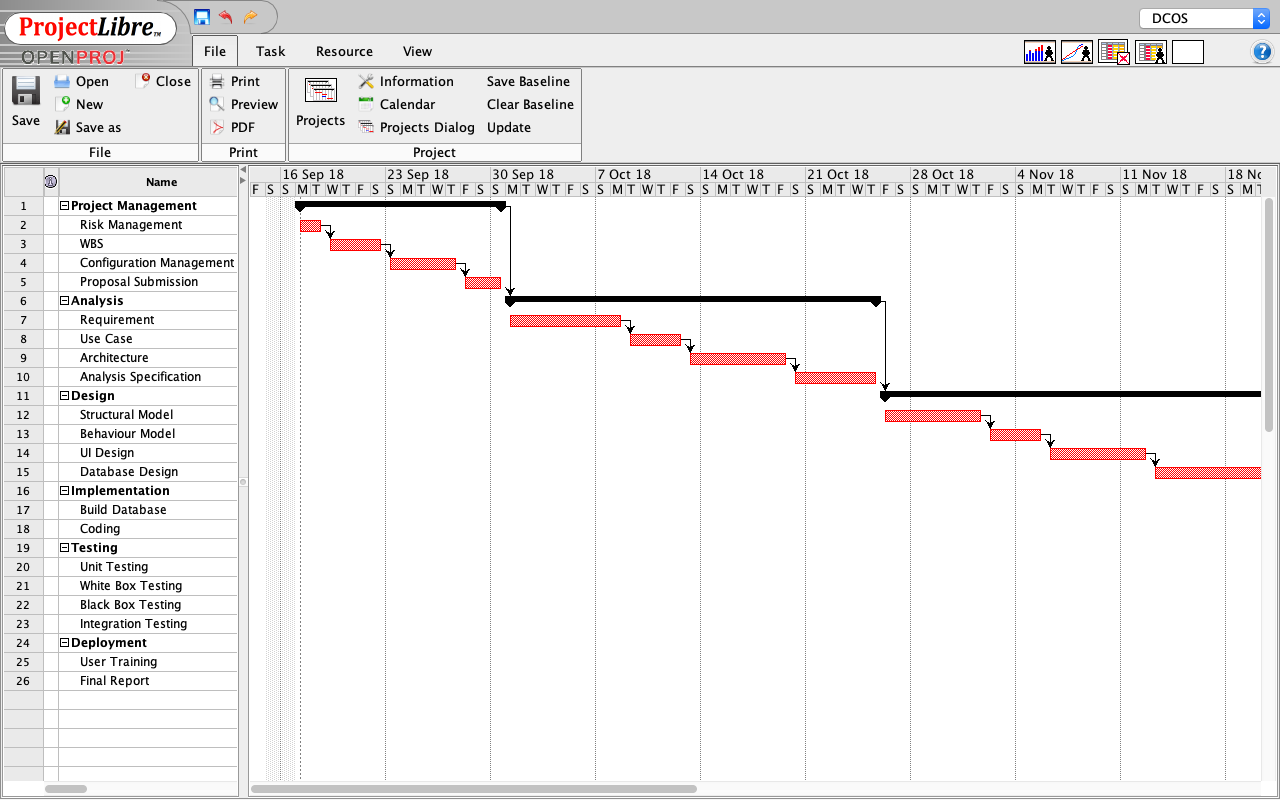
* **Project Management (14 days)**
  + Risk Management (2 days)
  + WBS (4 days)
  + Configuration Management (5 days)
  + Proposal Submission (3 days)
* **Analysis (25 days)**
  + Requirement (7 days)
  + Use Case (3 days)
  + Architecture (7 days)
  + Analysis Specification (7 days)
* **Design (30 days)**
  + Structural Model (7 days)
  + Behavioral Model (6 days)
  + UI Design (10 days)
  + Database Design (6 days)
* **Implementation (32 days)**
  + Build Database (3 days)
  + Coding (29 days)
* **Testing (10 days)**
  + Unit Testing (3 days)
  + Integration Testing (3 days)
  + White Box Testing (2 days)
  + Black Box Testing (2 days)
* **Deployment (10 days)**
  + User Training (4 days)
  + Final Report (6 days)

## **4.3 Scheduling / Gantt Chart**

The process of allocating, controlling and optimizing work, workloads, time, duration, deliverables to achieve a specific milestone is known as scheduling. By scheduling the specific task can done in specific time allocation period.



*fig. 4: Gantt Chart*

**

*fig. 5: Gantt Chart 2*

# **5. Risk Management**

Risk management can be a defined as the procedure approached to identify a possibility of suffering from loss, expectation of loss, a potential problem that may or may not occur in the future which can be analyzed and simplified after taking proper approach to the likely solutions. The risk can be controlled by analyzing the risk, level, likelihood, impact of the risk a likelihood is defined in percentage after examining what are the chances of risk to occur due to various technical conditions.

The possibility of risks to get impact in our project can be followed by shown below given procedures:

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 |

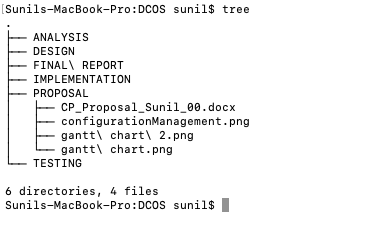
**Risk Management Table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S. No | Risks | Likelihood | Consequences | Impact | Solution |
| 1 | Incomplete information | 2 | 4 | 8 | Researches must be done thoroughly before initializing the project. |
| 2 | Hard Disk Failure | 1 | 5 | 5 | Data should be backed up into the cloud servers and other means of storage |
| 3 | Failure to meet the requirements | 2 | 5 | 10 | Planning should be done to approach the requirement. |
| 4 | Health Issues | 1 | 4 | 4 | Precautions should be taken to avoid any injuries or illness. |
| 5 | Hardware Incompatibility | 1 | 2 | 2 | Hardware components should be upgraded and maintained as per the requirement. |
| 6 | Time | 2 | 4 | 8 | Proper planning should be done without rescheduling the schedules |
| 7 | Natural Disaster | 1 | 5 | 5 | Best alternatives should be approached to minimize the effects. |

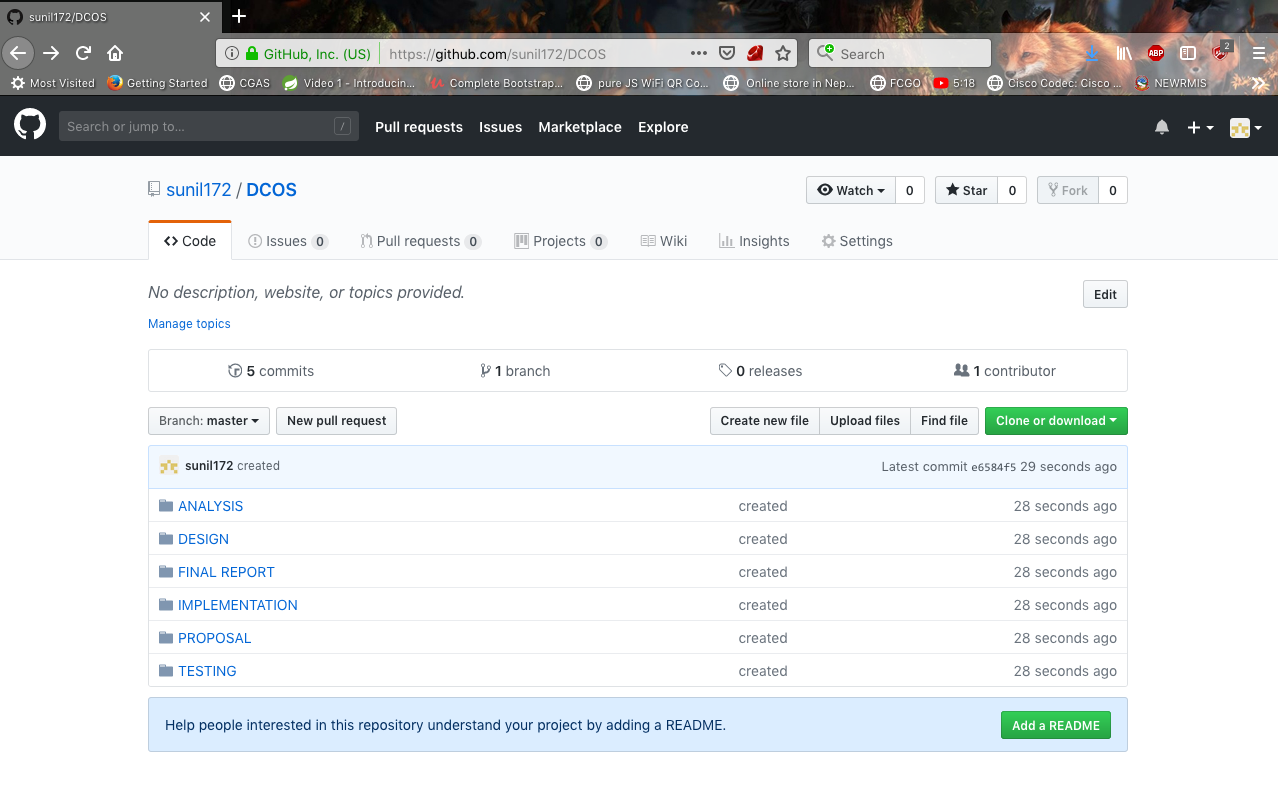
# **6. Configuration Management**

Configuration Management is a process for establishing, modification, maintaining consistency of project attributes performance, functional and physical attributes. Configuration management controls change in system by the functional relation between parts, subsystems and systems for effectively. Configure management consist of provisions for the storing, tracking, updating of all system information on a component, subsystem and system basis.

The process that records changes to a file or set of file over time which should recall specific versions latterly is controlled by Version Control. Version control allows system to compare files, identify differences, merge the changes if needed prior to committing any code. It also able to identify the track of application builds in development and production.



*fig. 6: Tree diagram from local drive*

**

*fig. 7: GitHub Repository*

# **7. Conclusion of the project**

The objectives of the project will be gained after development of this project. The project is able to handle the official correspondences letters and user management. This project will redundancy the central records and can access the desired records on their own station. It encourages the organization to enhance efficiency, integrity, tidy and smooth and more quality job handling.

# **8. References**

* Promoting E-governance at Local Level (2016), Local Development Training Academy [online] Available at: http://ldta.org.np/wp-content/uploads/2016/06/Research-report-on-Promoting-E-governance-at-Local-Level.pdf [Accessed Jan 2, 2019]
* SDLC-Waterfall Model (2018), tutorialspoint [online] Available at: https://www.tutorialspoint.com/sdlc/sdlc\_waterfall\_model.htm [Accessed Jan 2, 2019]
* Basic MVC Architecture (2018), tutorialspoint [online] Available at: https://www.tutorialspoint.com/struts\_2/basic\_mvc\_architecture.htm [Accessed Jan 2, 2019]
* Three-Tier Architecture (2018), techopedia [online] Available at: https://www.techopedia.com/definition/24649/three-tier-architecture [Accessed Jan 2, 2019]
* Work Breakdown Structure (WBS) (2018), workbreakdownstructure.com [online] Available at: https://www.workbreakdownstructure.com [Accessed Jan 2, 2019]
* Configuration Management (CM) (2018), TechTarget [online] Available at: https://searchitoperations.techtarget.com/definition/configuration-management-CM [Accessed Jan 2, 2019]