**Bike Showroom Management System**



Asmita Adhikari  
 00172902

**Computing Project (CP)**   
 Level 5 Diploma in Computing

Softwarica College of IT and E-Commerce  
Kathmandu, Nepal

Submitted to: Kiran Rana

# Introduction

## 1.1 Project Introduction

Project entitled “Bike showroom management system” is based on web-application developed for bike showroom maintaining daily activities like customer maintenances, bikes information, sales of bikes, booking of bike, bike companies details, new models of bikes could be seen by customers with their price ranges, Models of bikes from different companies. Admin could add new bikes on stock, delete sold out bikes also update the new stocks within their showroom.

## 1.2 Justification for project

### 1.2.1 Background of project

The system is developed for the users who could view bikes kept in sales and simultaneously book them. The system allows users to search the various models of bikes with their price ranges and functionality. Users could explore various kinds of bikes like sports bike, scooters, motorbikes and so on. When users book their desired bike, and request is sent to admin.

Tools that I would be using for building the projects are project libre, Mysql, VScode etc.

### 1.2.2 Problem Statement

The objective of the project is to create the management system for bike showroom is to show details about bikes in the showroom their functions with brands of bikes and in your budget. The system is web-based application which is new for the organization. Till now the showroom doesn’t have its web-based application they are following the old techniques of   
database storage in files or either in draft. Typically managing database would be difficult job, the new system might be somehow helpful for database storage, but some time system might crash, and work couldn’t be done.

## 1.3 Description of project

### 1.3.1 Features

Following are the features included in the project of bike showroom management system.

* **Login system:** Login of users are provided where user create their account and login when they visit the website.
* **Register for non-users:** Non-users could register in the website.
* **Types of bikes:** Different bikes are available in marketlike sports bikes, scotty, Mountains bikes etc.
* **Search for the bikes**: Bikes could be searched like different models of bikes.
* **Different mileage of bikes:** Different mileage of bikes are arranged in one order to make easier for view.
* **New launches of bikes:** New bike launches could be seen in website.
* **Brands of bikes from different companies:** There are different companies of bikes like Yamaha, Honda, KTM, TVS etc.
* **Bike of upcoming models:** Updated or newly coming bikes are added.
* **Feedback form users:** Customers could like or dislike the services provided by organization they could feedback into it.

**CATWOE** is used to create rigors and comprehensive root definition. It is a simple checklist used to stimulate thinking about problem and solutions. It is used to identify business achievement, problem areas and solutions.(anonymous, 2013-2015)

The six elements of CATWOE are listed below:

* **Clients:** Clients or Customer are the stakeholder who benefit or suffer during the change of system. In the first step of analysis customers are identified and understand how change in system affects them. The authorized person who signs in will be client of the system.
* **Actors:** Actors are people who are directly involved in system change. The impact of new system for them must be clarified. The stakeholder’s responsibilities are listed out. The admin of the system will be known as actor of the system.
* **Transformation:** It refers to what will happen to the data and what processes will be affected by the development of system. CATWOE analysis requires the list of inputs and the nature of change inputs undergo to give outputs. It deals with input and outputs the client’s gets during the access of the system which result robustness, accuracy, efficiency, reliability of the system. In the current scenario, customer visit liquor shop and view product then buy the liquor which is difficult for both customer and client to deal at the time and give response. By the development of new system both the customer and client could get benefit without facing problem.
* **World View:**  It refers to wider impact of transformed system. There might be positive and negative impact on the change of system on overall business matter. There might be different perspective of person regarding to solution to a problem as a world view. The problem of existing system is analyzed from various levels of stakeholders. For this system admin user would add product details and customers can view the product to order which would be delivery on cash. They can surf variety of brands, types of liquor with certain offers and prices.
* **Owners:**  They are person who owns the organization and has power to take decision for changing project and future works. In the system admin is the owner.
* **Environmental Constraints:**  It deals with political, legal, demographical, technological, social, economic environmental factors limitations and constraints. These environmental factors are kept in mind during the system development for successful project.

### Conceptual model:

The descriptive model of a system based on qualitative assumptions about its elements, their interrelationships and system boundaries. It is used to explain how the system should function and what activities are important to take place of processes. In this model 3E (Efficiency, Efficacy and Effectiveness) are measured to improve system performance.[(Anon., 2018)](#_Chapter_7:_Conclusion:)

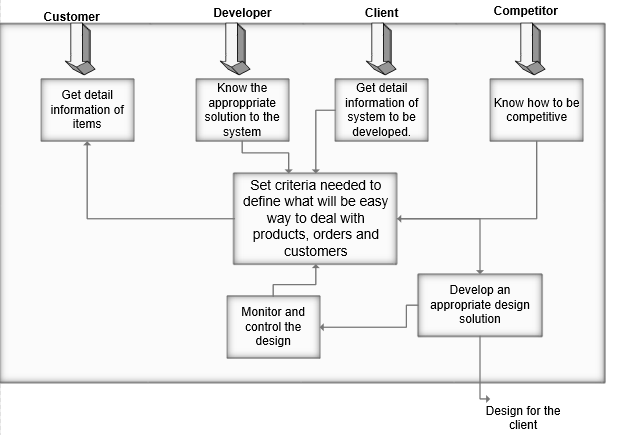


Figure 2: Conceptual model

## **Advantage of SSM:**

* Involvement and participation of user during development
* Open discussion of problems, needs and solution
* Helps to deal with various environmental factors
* Joint problem solving technique
* Various perceptions

## **Disadvantage of SSM:**

* Difficult to manage
* Does not fit for complex organization
* Lack of technical perspectives
* Difficult to get on agreements between users

# 2. Project Scope

## 2.1 Scope and limitation of project

A part of project that involves planning which involves defining and detailed list of specific goals, features, quality of a project within the untimely cost. In other word, project scope is what needs to be achieved in work done to deliver of project. (totally tech, n.d.)

**Scope of project**

The project of bike showroom management system is made for online customer who could book the desired bike. The project removes old methodology of people going to showroom and see the bike of their want.

**Limitation of project**

* Project doesn’t have online payment system customer could only book bikes.

## 2.2 Aims and objectives

Aims for the project bike showroom management system are as follows:

* Implement a web-based bike showroom management system.
* User convenient.
* Maintain data in web-page so that costumer would be up-to-date.

Objective for project bike showroom management system are as follows:

* To access web application for showroom.
* Admin awareness about system uses, how system works.
* Update all the showroom up-to-date stocks.
* Managing and creating proper documentation for further references.

## **Section 2.2: Feasibility Study:**

A feasibility study describe about viability of an idea within an emphasis on finding potential problems. This study will address where and how business will be operating. Market, Technical and Financial feasibility could be known by this study.

For this project some of the steps for feasibility study had been taken within the interaction of stakeholder which is described below:

### Interview:

The most common method to get information of an individual is interview. For the collection of data various type of interview such as structured, semi-structured and un-structured interview. It is conducted by different means as face-to-face, telephone, emails.

For the detection of problem live interview had been done with client and his customers. In this session, we had discussed about problem face from current system and what they want from new system. A certain customers where telephoned because of busy schedule we had tried to interact with daily customers and other customer to be get feed backs.

**Result:**  After this session, we get to know that there were both positive and negative about new system implementation from the customer. The problem faced by our client is also identified. For him it was difficult to interact with many people at a same time and get the product they want with in a bragging of prices. In customer point of view the new system will help them to get the product as per their wish with in all description they do not have to rush for offers too.

### Questionnaires’’:

It is another way to get information with the list of question asked to certain group of people to know their perspectives. Different sort of question is asked to people that may include the multiple choice, yes or no question to extract their view with short points.

Following list of question is asked to respondents:

1. Do you remember customers order in same time?
2. How long will you remember product details?
3. How many customer visit shop?
4. Did you get the product you want from store?
5. Do you prefer online shopping?

## **Use Case Diagram:**

It is one of behavioral diagram that describe interaction of actor within a system. It describe about the user of the system and their roles.

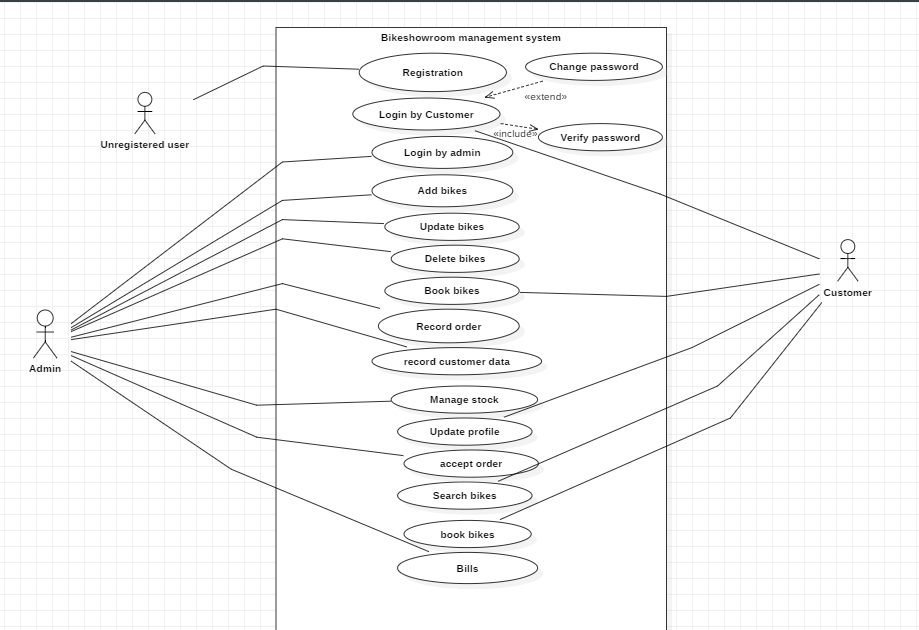
The reason to sketch use case diagram are listed below:

* To identify how actor interact system without worrying about functional requirement
* To visualize functional requirements which will be translate into choice of design and priorities for development.
* To identify external and internal factor which may influence system?

**Symbols and Notation of Use Case Diagram:**

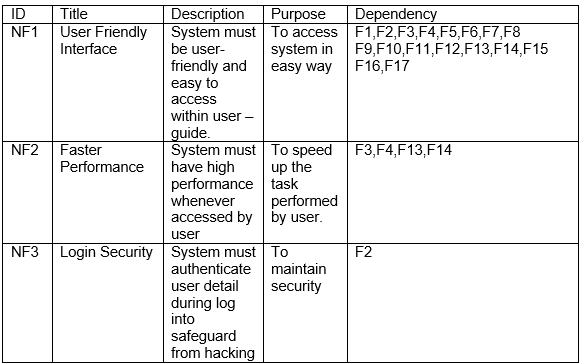
1. **System:** It is a rectangular box which contains use case.
2. **Use Case:** It is a round shape which describe role of user.
3. **Actors:**  It is a stick man which is user of system.
4. **Relationships:**  It is a simple line that shows relation between actor and use cases. There are two type of relationship i.e. include and extend.

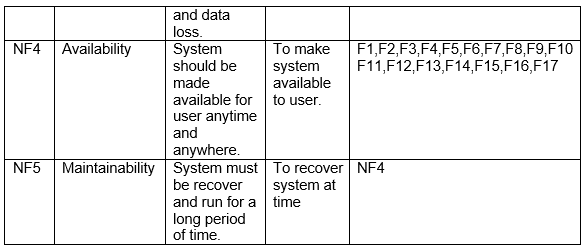
The **use case diagram** of system is shown below:



## Non-Functional Requirement:

* The non-functional requirements are listed below:
* **Security:**  The security of the data plays important role for the development of system. The secure system can be gain by authentication during login, strong password and day-to-day backups. If the personal information of the user is used in illegal way such as hacking and data misuse then it may arise huge problem.
* **Performance:** It deals with robustness of system. The execution of the functionality must be faster in use. The performance of the system is determined when various users will access same functionality at same point of time. The fault-tolerance of the system is also measured.
* **Usability:** The ease of use of system for user is Usability. Whenever user access the system there must proper understand, co-ordination and learning between user and system.
* **Availability:** The system must be made available whenever user wants to access from wherever.
* **Reliability:** System should provide extract output for the input given by the user when a task is being performed. For the protection of system from failure error must be identified and correction of strategy.





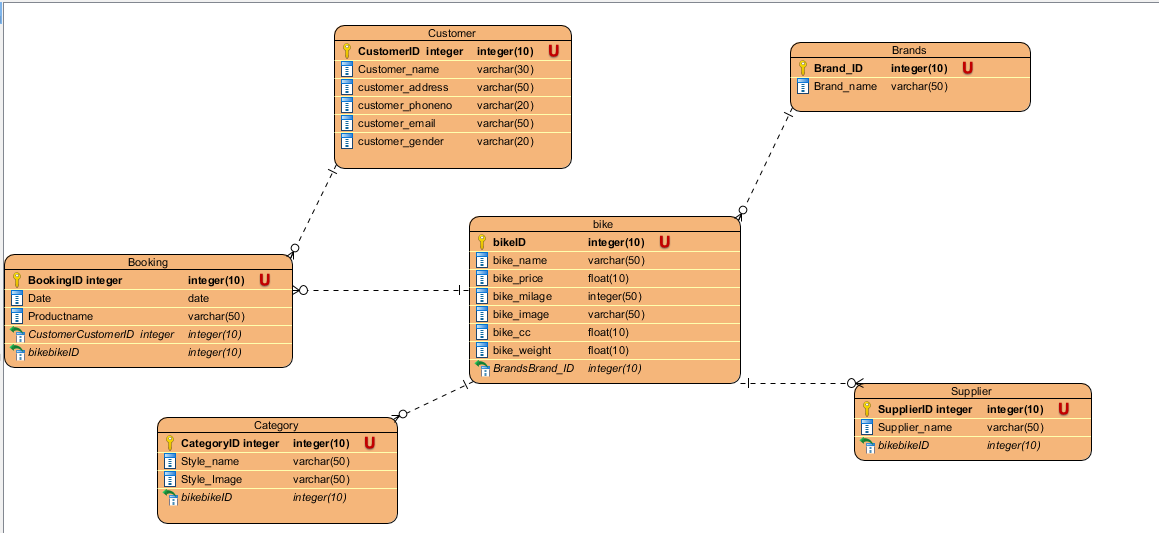
## **: Prioritization:**

Prioritization of requirement would help to deliver the largest and most immediate business benefits in earlier stage. The reason to use Moscow prioritization method is to simplify the requirement in high, medium and low importance during the development.

MoSCoW stands for Must Have, Should Have, Could Have and Wont Have. The requirement within Must have, Should have or could have will be initially try to deliver to client but the wont have prioritization will be fulfill in the future use.[(anonymous, 2018)](#_Chapter_7:_Conclusion:)

* **Must Have:** The requirement which is guaranteed to deliver on target day.
* **Should Have:** The requirement which is important but not vital
* **Could Have:** The requirement which is less important
* **Won’t Have:** The agreed requirement which will be deliver in future purpose.

|  |  |
| --- | --- |
| Title | Prioritization |
| Registration | Must have |
| Login | Must have |
| Add new Product | Must have |
| Search Product | Should have |
| View Product | Should have |
| Edit Product | Should have |
| Delete Product | Should have |
| Order Product | Could have |
| View Order | Could have |
| View User | Should have |
| Delete User | Should have |
| Send Notification | Should have |
| Add product to cart | Should have |
| View cart | Should have |
| Remove Product from Cart | Could have |
| View Profile | Should have |
| Edit Profile | Could have |



ER diagram

Justification

* To visualize database design ideas, we have a chance to identify the mistakes and design flaws, and to make correction before executing the changes in database.
* By visualizing a database schema with an ERD, we have a full picture of the entire database schema. You can easily locate entities, view their attributes and to identify the relationships they have with others.

Advantage

* It is very simple if we know relationship between entities and attributes.
* It is better visual representation.
* It is an effective communication tool for database designer.

Disadvantage

* It has limited constraints and specification.
* Information can be hidden in ER model.
* It is difficult to show data manipulation in ER model.

# Meta Data

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| User table | | | | | |
| Column Name | Type | Length | Key | Null | Description |
| Customer ID | Int | 10 | PK | No | Unique identification of customer |
| Customer Name | Varchar | 30 |  | No | Customer’s Name |
| Customer’s address | Varchar | 50 |  | No | Customer’s address |
| Customer’s email | Varchar | 50 |  | No | Customer’s email |
| Customer’s phone | Varchar | 20 |  | No | Customer’s phone |
| Customer’s gender | Varchar | 20 |  | No | Customer’s gender |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Product table | | | | | |
| Column Name | Type | Length | Key | Null | Description |
| Bike ID | Int | 10 | PK | No | Unique identification of Bike |
| Bike Name | Varchar | 30 |  | No | Bike Name |
| Bike price | Float | 10 |  | No | Bike price |
| Bike CC | Float | 10 |  | No | Bike CC |
| Bike mileage | Float | 10 |  | No | Bike mileage |
| Bike weight | Float | 10 |  | No | Bike weight |
| Bike Image | Varchar | 20 |  | No | Bike’s Image |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Booking table | | | | | |
| Column Name | Type | Length | Key | Null | Description |
| Booking ID | Int | 10 | PK | No | Unique identification of Booking |
| Date | Date |  |  | No | Booking Date |
| Product name | Varchar | 30 |  | No | Product Name |
| Customer ID | Int | 10 | FK | No | Customer’s ID |
| Bike ID | Int | 10 | FK | No | Bike ID |
| Bike Image | Varchar | 50 |  | No | Bike Image |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Category table | | | | | |
| Column Name | Type | Length | Key | Null | Description |
| Category ID | Int | 10 | PK | No | Unique identification of Category |
| Style Name | Varchar | 50 |  | No | Bike Style name |
| Bike ID | Int | 10 | FK | No | Bike ID |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Brand table | | | | | |
| Column Name | Type | Length | Key | Null | Description |
| Brand ID | Int | 10 | PK | No | Brand ID |
| Brand Name | Varchar | 50 |  | No | Bike Brand name |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Supplier table | | | | | |
| Column Name | Type | Length | Key | Null | Description |
| Supplier ID | Int | 10 | PK | No | Supplier ID |
| Supplier Name | Varchar | 50 |  | No | Bike Supplier name |
| Brand ID | Int | 10 | FK | No | Brand ID |

# Activity Diagram

Activity diagram is a behavioural design tool which shows the flow of activity in an application. An activity can be drawn using activities, swim lanes, initial and final point, conditions, signals, join and fork nodes, merge events, action flow, object flow, time events, interruptive edge etc.

I am using this design because it helps me, or any other designer understand how different functionality in my system will be working. Since I will be implementing using these designs, the implementation process will be a lot faster and easier.

Justification

* To draw the activity flow of a system.
* To describe the sequence from one activity to another.
* To describe the parallel, branched and concurrent flow of the system

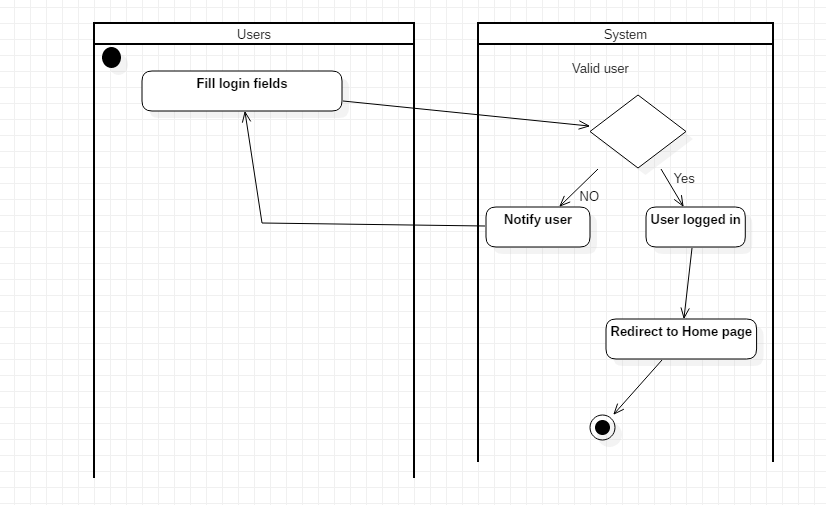
Advantage

* Since it is the most user-friendly diagram. So, generally regarded as an essential tool.
* It helps to display multiple conditions and actors within a work flow using swim lanes.
* These diagrams are normally easily comprehensive for both analysts and stakeholder.

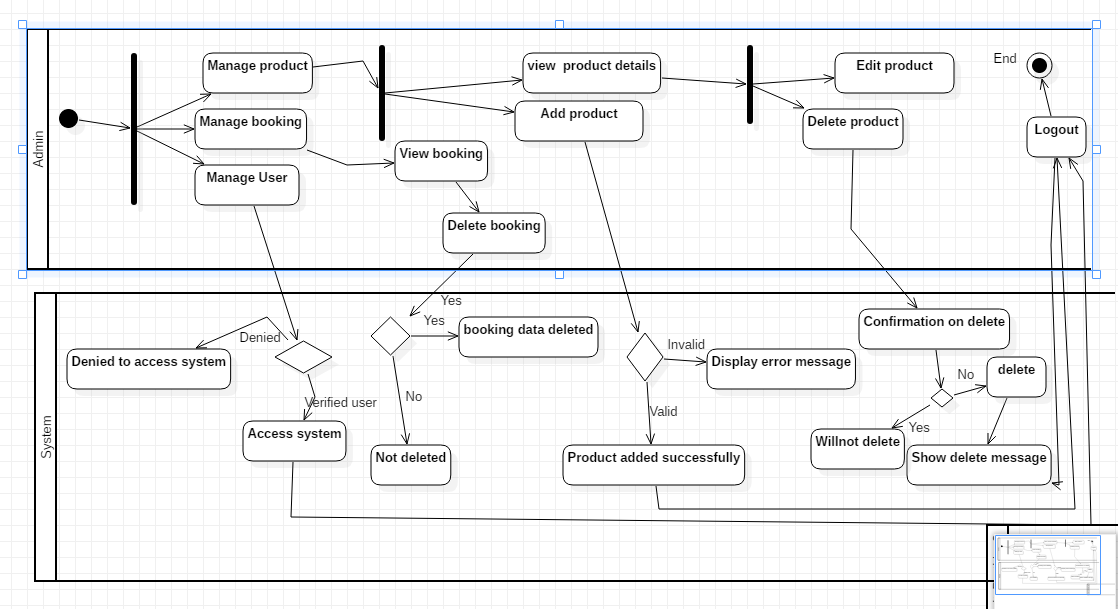
Disadvantage

* These diagrams can lead the over complex which might affect the user-friendly nature.
* These diagrams do not give the detail about how object behave or collaborate.

## Login Activity



In login activity, user and the system interact with each other to perform login. A user submits his/her login to system and the system checks, if they are valid then the validity will decide the success of login.



Class diagram

Class Diagram shows a static view of a system. Class diagram consist of collection of class, relationships, interface, constraints and collaborations. It is a diagram which can mapped directly with object-oriented languages.[(anonymous, 2018)](#_Chapter_7:_Conclusion:)

The above class diagram consists of various relationships such as composition, dependency, association. Each model consists of controller which is dependent to own classes. Product is related to product type and user model within a composition relationship in the absence of one another could not work. User model has a multiple association relationship as a one user can get much notification at a time. User and Order has a composition relationship since order is dependent to user. Sale and Cart is dependent to order class within a composition relationship.

## 

# 3. Development Methodology

## 3.1 methodology used

Software development methodology used in the project is waterfall model. Waterfall model is linear sequential method of software development which involves continuous process in development. Each process must be completed going to another phase.

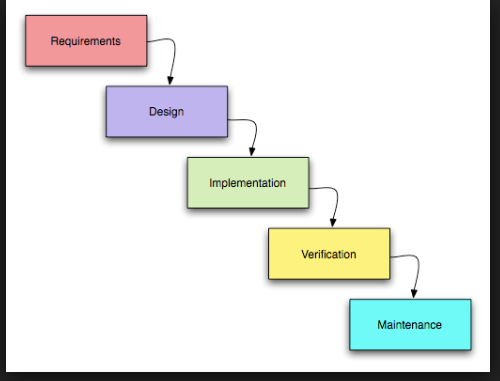


Figure 1 Waterfall model

Waterfall model used in software development methodology has five steps as shown in above figure. Each step should be completed before going to another step.

**Advantages of waterfall model:**

1. Each phase has its own distinct goals.
2. Each phase would be complete in definite deadline.
3. Easy to manage.
4. Completed in one time and phases don’t overlap.

## 3.2 Design pattern

Design pattern used in the project is MVC (Model View Controller)

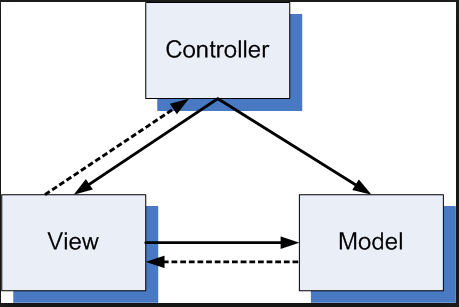


Figure 2 Design pattern using MVC

Architectural pattern used for software development MVC design pattern specifies that application contains data model, presentation model and controls information. MVC requires each separated into different objects. The design patterns mostly relate UI interaction layer of application.

**Following explains MVC:**

**Model**: It contains pure application data which doesn’t concern UI or presentation.

**View**: It present model data to user in a proper interface and allows user to manipulate data.

**Controller**: controller is between model and view. View is updated when model changes and model is updated when user manipulates the view.

## 3.3 System Architecture

3 tier-architecture is used to make the application. Software architecture containing 3 layers of logical computing. 3 tier architecture contains 3 layers.

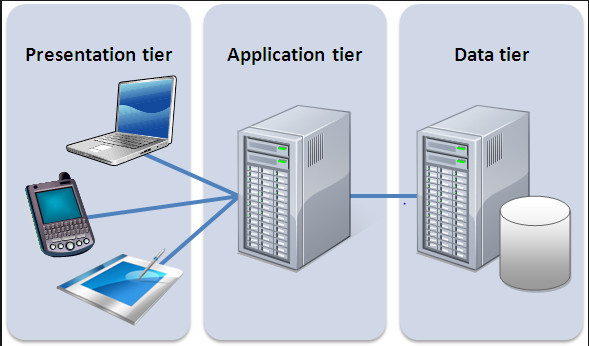


Figure 3 Three tier-architecture

1. Presentation layer: Font-end layer in three tyre-architecture consist of user interface. This layer interface is graphical accessible through web browser application.
2. Application layer: Application tier contains logical functions of an application.
3. Data layer: data layer contains data storage system in database.

**Advantages of MVC.**

1. Faster development process.
2. Friendly development platform.
3. Modification doesn’t affect entire model.

# Risk Management

Process of identifying, analysing the risk of the certain task and identifying solution to overcome the problem is known as risk management. A risk is anything which impact directly on the project’s performance, usability or budget.

Risk could be managed by following methods.

* Identifying risk
* Analysing the risk
* Evaluating the risk
* Overcoming the risk
* Monitoring and reviewing risk

***Impact = Likelihood \* Consequence***

I am going to use following risk alleviation approach to manage risks and provide proper actions.

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

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

Table 7: Risk Likelihood and Consequence Values

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risk** | **Likelihood** | **Consequence** | **Impact** | **Action** |
| Server failure | 1 | 5 | 5 | Monitor server regularly. |
| Database Failure | 1 | 4 | 4 | Back-up of data in database. |
| Power fluctuation | 1 | 2 | 2 | Reduction of amplitude voltage. |
| Server overload | 2 | 5 | 10 | Maintenance on monthly basic, cleaning the servers and databases. |
| Employee theft | 2 | 5 | 10 | Marinating trustful environment with employee. Employee dissatisfaction leads to employee theft. |
| Lack of resources | 2 | 2 | 4 | Budget increment and make sure of available resources. |
| Lack of user knowledge | 3 | 2 | 6 | Users awareness should be provided about using the system. |

Table 8: Risk Management Table

# Configuration Management

Configuration management is the process of ensuring consistency among physical and logical assets in an operational environment. configure process seeks to identify and track individual configuration items has on another system. Administration, technical and software developers could configure tools to verify the change to one configure item has on other system.

Version control is the important function of software configuration management , which is responsible for all elements in configuration database automatically assigned version of logo, and ensure that the version of the name of the uniqueness.