# Chapter 1. INTRODUCTION

# Project introduction

Bus ticket booking system. In this system people can book ticket their home or anywhere like by using their application. Nowadays people want to use technology and people do not want to stand on long queue. The system will be of great help to the community because customer can order, pay for, obtain and validate tickets reference number from any location and at any time using mobile phones.

# Justification of project

1. Background of the project

Bus ticket booking system is the desktop application which can access through laptop. In this application user can book several seats. And you can choose the seat. You also know rate of bus ticket from this application. Ticketing is the most important module in this project which uses all the tables and calculates fare for the passengers.

1. Problem statement

Exciting system require more computational time, more manual calculation, and the complexity involves in selection of features is high. The other problem is lack of security of data deficiency of data accuracy time consuming etc.

1. Description of the project
2. Features

# Features of the system

* Admin can Add, delete, update the candidate
* User friendly
* This application can be access easily from wherever the customer wants
* Customer can book a seat
* Customer can register their name from this application
* User can rate a ticket

# Overview of project

# Chapter 2: Scope of the project

1. Scope

Bus ticket booking system that manages the work load of the ticketing. It makes systematic uses of the relative data and store on the data through the help of application. Main scope to ensure the load of data and the person and brings down the solution. Taking out the tedious from the day to day job of customer and staff using the system.

1. Limitation

* The limitation is that it is desktop-based application and does not available on the online sectors.
* Does not have facilities of online payment.
* Customer should book from the counter or either from phone.

1. AIMS

The software is designed with aims and moto to manages the heavy jobs tasks from their day to day activities. Some of the aims of the application formation can be given below:

* Main is to simplicity the work.
* Scientific management of data.
* Reusability of data.
* Customer satisfaction.
* User satisfaction.
* Main but not the list profit motive for the company.

1. Objectives

Some of the objectives are given below:

* To reduce the work load.
* Smooth performance.
* Customer management.
* Continuous services.

1. Overview of the scope

The task has been divided into the four categories. In which scopes of the system is described and the aims of the proposal. Limitation describe their limitation of the application as well as the objectives perform the main object of the building of task.

Each of the relationship and their performance, function, aims, scope has on scope of project.

Chapter 3: Development Methodology

1. Water fall Model

I have used water fall model in this project. This methodology focusses on sequential stage where a one stage complete then another stage starts. There are other model like Agile m= development where multiple stages can be started without complete the previous one. This might create a much problem development process that is why I follow waterfall methodology.

For this project waterfall model has been chosen as:

* Easy to understand
* Steps of procedures
* Easy to follow the steps
* User can not jump the steps
* Sequence step need to follow before move on next step

A close up of a device

Description automatically generated

Figure1: waterfall model

1. Design pattern
2. Model view controller Pattern

MVC stands for Model view Controller Pattern. This pattern is used to separate application concerns.

Model: Model represents object. Each of the class and other represent from here.

View: view represents the visualization of data that model contains.

Controller: controller act both model and view it controller the data flow into model object and updates the view. (Aththanayake, 2010)

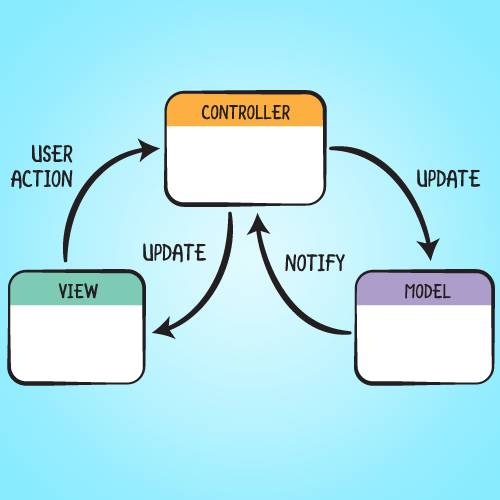


Figure2: Model view controller

1. System architecture

System Architecture is the conceptual model that defines the structure, behavior and more views of stem. I have used 2 tier system Architecture.

Our system is based on the two tier because the data is store on the same device and only between application and database.

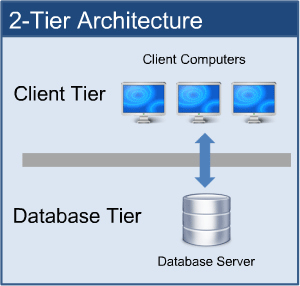


Figure 3: 2 tier system Architecture

Client Tier:

In this layer client activities are listed here through the help of application or visual user interface. Client request through the predefine user interface as well insert through it.

Database Tier:

The data that comes from the user interface UI has been store here in the form of rows and columns. Through UI user requested is selected items from their views on it.

# Chapter4: Project plan work breakdown Structure

1. Work breakdown structure (WBS)

Work breakdown is the process of decomposition of whole task into a sub task that can be easily managed. Decomposition sub task help to identify the milestones and controlling points of the project. project. Also, WBS helps to estimate all the risk, cost and time factors of project very easily.

Bus ticket booking System

Analysis

Proposal

Final Document

Testing

implementation

Design

Feasibility Study

Study

Build Database

Unit testing

User manual

Structural M

Scope

Integration Testing

Behavioral Model

Gathering Requirement

Planning

Black Box Testing

UI Design

Use case

Final report

Coding

Database design

Class diagram

Monitoring

White Box testing

1. Milestones

|  |  |
| --- | --- |
| Milestones | Date |
| Project management  Risk management  WBS  Configuration management  Proposal Submission | 2019-03-25 to 2019-03 -28  2019-03-29 to 2019-04 -01  2019-04-02 to 2019-04 -05  2019-04-06 to 2019-04 -09 |
| **Analysis**  Requirement analysis  Use Case  Class Diagram  Requirement Specification | 2019-04-10 to 2019-04 -16  2019-04-17 to 2019-04 -23  2019-04-24 to 2019-04 -30  2019-05 -1 to 2019-05-08 |
| **Design**  Structural Model  Behavioral Model  UI Design  Database Design | 2019-05 -09 to 2019-05-14  2019-05 -15 to 2019-05-21  2019-05 -22 to 2019-05-27  2019-05 -28 to 2019-06-03 |
| **Implementation**  Building Database  Coding | 2019-06 -04 to 2019-06-09  2019-06-10 to 2019-06-24 |
| **Testing**  Unit Testing  Integration Testing  Blackbox Testing  Whitebox Testing | 2019-06-25to 2019-06- 26  2019-06-27 to 2019-06-28  2019-06-29 to 2019-06-30  2019-07-01 to 2019-07-01 |
| Final Documentation  User manual  Final Report | 2019-07-02 to 2019-07-12  2019-07-02 to 2019-07-07  2019-07-08 to 2019-07-12 |
|  |  |

# Description of milestone

Project Management:

Total 16 days distribute for this task and further divided risk management (4 days), WBS (4days), configuration management (4 days) and proposal submission (4days).

Analysis:

For Analysis total 29 days allocate for this task and divided these days for requirement(7days), use case(7days), class Diagram (7 days) and requirement Specification (8days).

Design:

In this task total 26 days and further divided into structural model (6days), behavioral model(7days), Ui Design (6days) and database design (7days).

Implementation:

For this task total 26 days and divided into database build(6days) and for coding (15) days. Database decide the low use of space if used of accurate data. And for implementation the logical part took time and knowledge.

Testing:

Total 7 days for testing these days divided 2 days for unit testing, 2 days for integration testing, 2 days for black box testing and 2 days for white box testing.

Final Documentation

Total 11 days for documentation in which 6 days for user manual and 5 days for final report. It ensures that time for new system to adopt took time.

1. Scheduling: Gantt Chart

Gantt chart is graphical depiction of project schedule. It’s a type of bar chart that shows the start and finish dates of several elements of a project that include resources, milestones, tasks, and dependencies.

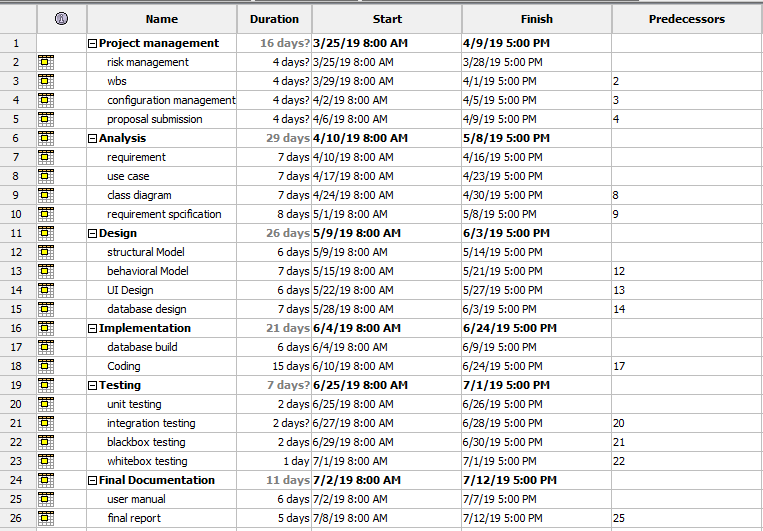


Figure 3: Schedule for Gantt chart

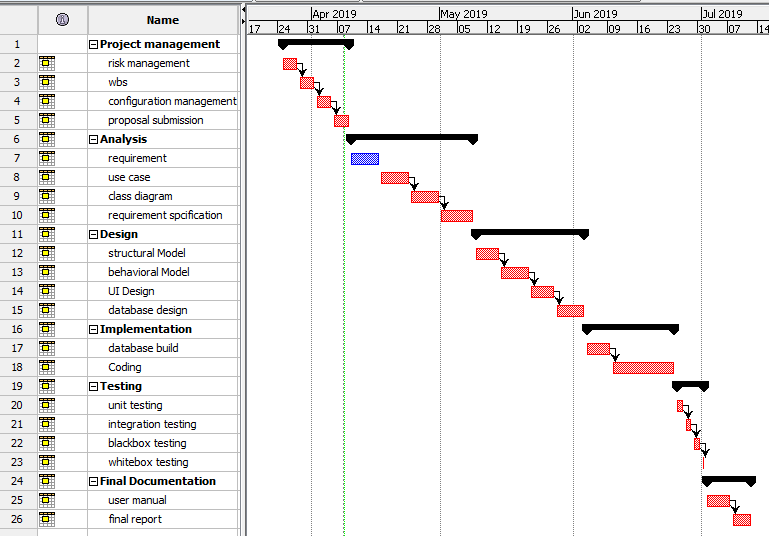


Figure 4: Gantt chart

Chapter 5: Risk Management

Risk management can be defined as the process to identify, analyze, evaluate and tackle the loss exposer and to monitor the risk control and resources to mitigate the adverse effects of loss

In some cases, risk can cause serious problem which can lead the project to failure so to reduce the impact of the risk that occurs we use of technique which calculate the impact following the likelihood and consequences of the risk

Risk Likelihood values are given below

**Impact = likelihood \* consequence**

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

Risk consequence values are given below

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

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S.N. | Risks | Likelihood | Consequence | Impact | Solution |
| 1 | Hard Drive failure | 2 | 3 | 6 | Regular monitoring on the system |
| 2 | System failure | 1 | 5 | 5 | Proper backup and regular maintenance |
| 3 | Theft | 2 | 4 | 8 | Protection from unauthorized use using password and encryption |
| 4 | Natural disaster | 1 | 5 | 5 | Safe level of place and minimize possible chances early before |
| 5 | Untrained employee | 2 | 3 | 6 |  |
| 6 |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Chapter 6: Configuration Management:

The configuration management check for the data, version controls, functions. Changes in the system quality as well. To ensure that the quality, performances, criteria that needs.

Chapter7: Conclusion