2019

Nischal Tripathi

Batch 23 L5DC

7/1/2019

Train ticket booking system



NCC ID: 00176300

Sudeep Lal Bajimaya

Submitted To

Submitted By

Contents

[1. Overview 4](#_Toc12830010)

[1.1 Introduction 4](#_Toc12830011)

[1.2 Background of the system 4](#_Toc12830012)

[1.2.1 Problem statement 4](#_Toc12830013)

[1.3 Justification of proposed system 4](#_Toc12830014)

[1.4 Overview of the proposed system 4](#_Toc12830015)

[2. Scope 5](#_Toc12830016)

[2.1 Aims 5](#_Toc12830017)

[2.2 Objectives 5](#_Toc12830018)

[2.3 Features 5](#_Toc12830019)

[2.4 Overview of scope 6](#_Toc12830020)

[3. Development methodology 6](#_Toc12830021)

[3.1 Methodologies to be used 6](#_Toc12830022)

[3.2 Design pattern 7](#_Toc12830023)

[3.3 System Architecture 8](#_Toc12830025)

[4. Project Plan 9](#_Toc12830026)

[4.1 Work breakdown structural 9](#_Toc12830027)

[4.2 Milestone 10](#_Toc12830028)

[4.3 Scheduling 12](#_Toc12830029)

[5. Other Project Activities: 14](#_Toc12830030)

[5.1 Risk Management 14](#_Toc12830031)

[5.2 Configuration management 16](#_Toc12830032)

[6. Conclusion 18](#_Toc12830033)

[7. Bibliography 19](#_Toc12830034)

List of figures

[Figure 1: Waterfall Approach 5](#_Toc12828715)

[Figure 2: MVC pattern 6](#_Toc12828716)

[Figure 3: 3 tier 7](#_Toc12828717)

[Figure 4: WBS Figure 8](#_Toc12828718)

[Figure 5: Time scheduling. 11](#_Toc12828719)

[Figure 6: Gantt chart 12](#_Toc12828720)

[Figure 7: Tree structural of my files in local drive of PC. 15](#_Toc12828721)

[Figure 8: GitHub file structure. 16](#_Toc12828722)

# 1. Overview

## 1.1 Introduction

This world is also known as the world of technologies. In this era or technologies we can do anything from the internet. Similarly, people face many difficulties while travelling for ticket booking process because they have to stand in line and while they get their turn in the reception the ticket might get sold out so I decided to create a train ticket booking system for the easiness of the customer.

Train Ticket Booking System is an online service which will help the customer to have a great ticket booking experience for train and also they can register themselves to book, change or delete there ticket bookings also they can pay the ticket fare through their payment card and can get their booking receipt too. Similarly in the staff side they will also get great experience in editing rescheduling and confirming the bookings from the online system itself.

## 1.2 Background of the system

This system is mostly used to make the users life easier and it does it so by helping the peoples to book the online ticket whenever they like and also allows the users to edit, delete or reschedule their ticket booking. While in the past this system was not online it was only computer based system but I have decided to create it online by using PHP so that everyone can book their own ticket from anywhere.

### 1.2.1 Problem statement

Some of the problems which were in the past in train ticket booking system were as follows,

* People have to book the tickets after waiting in the long line.
* Due to paper based booking the booking system as unmanaged.
* Passenger data wasn’t secure.
* Employees had to work hard regarding the booking of the ticket.

## 1.3 Justification of proposed system

Thus this is the technical era and everyone has their own gadgets which supports internet like smart phones, tablet, laptop etc. in which the users can run this project from. Likewise this project is made using PHP. I will make the ticket booking experience more unique and will also help in business growth.

## 1.4 Overview of the proposed system

Train ticket booking system is a web based system which can be used from anywhere at any time. This system is made by using PHP. It can be associable from laptops, PC, mobile phone and other gadgets which have web browser in it. It is a fast way and easy way of booking ticket with great experience of graphics with functionality. This system can be used from age group of 5 to 60.

# 2. Scope

## 2.1 Aims

* To design a web based train ticket booking system which helps to make the reservation and the payment through the system.
* To develop a system which maintains the customer’s data security.
* To develop a system which will provide the customers great user experience while using the system.

## 2.2 Objectives

* Customer can check the train details and its availability.
* Customer can register / login to the system.
* Customer can make changes to their bookings.
* Customer can book tickets along with their travel class.
* Customer can cancel the bookings.
* Customer can give ratings as well as their feedbacks regarding the service.
* Customer can pay online through their card.
* Admin can reschedule the time of the train.
* Admin can edit, Delete the user’s records.

## 2.3 Features

* Customer:
* Registration and login
* Edit, delete or book the ticket
* Online payment
* Update their profile information
* View the train information
* COMPARE TRAIN FAIRS
* FAQ
* Admin
* Login panel
* Edit/delete records of customer as well as train
* View users information
* Confirmation of booking

## 2.4 Overview of scope

The system which I am creating is fully web based made from core PHP. This system can be used from anywhere at any time if they are connected to the internet. Also, to make in online at the internet we have to use some webhosting sites and host the system.

# 3. Development methodology

## 3.1 Methodologies to be used

Software Development methodology is a framework that is used for structural, plan and control the process of developing an information system. For this project I have used Waterfall model. Water fall model is a traditional system development life cycle (SDLC) his method involves a complete set of steps that a team follows. The fundamental idea is to divide the development process into a series of the phases or stages, each of which finished before next one starts. [(tutorialspoint.com, n.d.)](#_Bibliography)

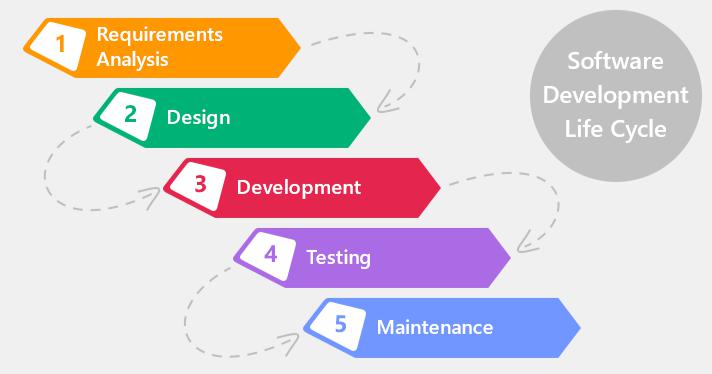


Figure 1: Waterfall Approach

Advantages of waterfall model:

* It’s simple and very easy to understand.
* It is easy to manage.
* It works well for smaller projects too because due to the requirements are easier to understand.
* In this approach phrases are processed and completed at a time.

Disadvantages of waterfall model:

* High amount of risk with uncertainty.
* Poor model for long project.
* Not suitable for complex Object oriented projects.
* Going back from ongoing phrase for some changes will be very difficult.

## 3.2 Design pattern

Design pattern is a repeatable solution to commonly occurring problem in software development. Similarly, there are many design pattern but for this project I have used Model View Controller (MVC) pattern. MVC design pattern specifies that an application consist of a data model, presentation information, and control information. MVC pattern is for architect patter not for complete application. This is really effective and efficient design pattern also this is most used design pattern so due to this I will be using it for my system.

## C:\Users\Hactivist\Desktop\model-view-controller-light-blue.png

Figure 2: MVC pattern

## 3.3 System Architecture

System architecture that I will use in this system will be 3-tier because this is the project of client to server system so the user don’t have to use the database due to this reason I am using this method. 3-tier architecture also improves the horizontal stability, performance and availability. Below down are the three tiers. [(Rouse, 2019)](#_Bibliography)

1) Presentation tier: It is a tier which is built with HTML, CSS and JavaScript and is deployed to the computer by web browsers.

2) Application tier: Application tier also known as logic tier is written in java which contains the business logic which supports the application core functionalities.

3) Data tier: It consist of database and programs for managing read and write access to a database. The database system for managing read/write access are MYSQL, Microsoft SQL Server and MangoDB.

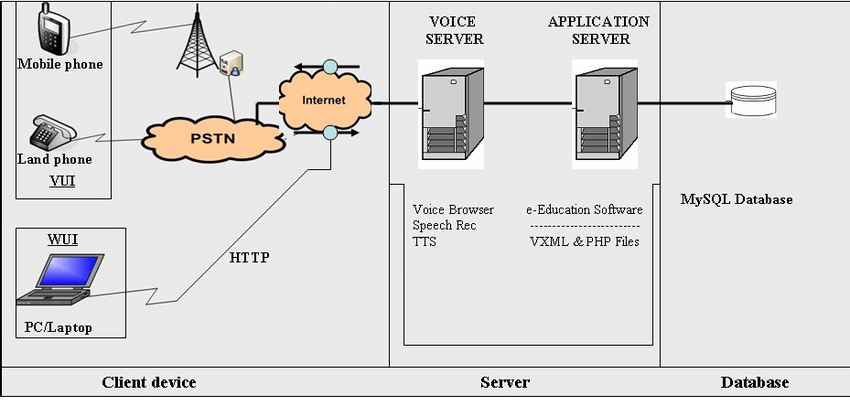


Figure 3: 3 tier

Since, to develop this system I have used PHP language. It is a web based language and also the mentioned figure shows that PHP scripts are executed through web browser. Also, the databases are saved in database server using SQL.

# 4. Project Plan

## 4.1 Work breakdown structural

It is a method of getting complex, multi-step projects done. I t is a process of dividing the large projects to complete the project in fast way. While projects are divided in small chunks it will help to know the information in more depth. Dividing the projects will help in proper planning. [(workbreakdownstructure, n.d.)](#_Bibliography)

Figure 4: WBS Figure

## 4.2 Milestone

|  |  |  |
| --- | --- | --- |
| **Task number** | **Task Description** | **Deadline** |
| **1.** | **Proposal** | **1st July** |
|  |  |  |
| **2.** | **Analysis** | **29th July** |
| i. | Planning | 3rd July |
| ii. | Requirement gathering | 8th July |
| iii. | Feasibility study | 12th July |
| iv. | Analysis method | 18th July |
| v. | System Requirement Specification (SRS) | 23rd July |
| vi. | Use case diagram | 29th July |
|  |  |  |
| **3.** | **Design** | **29th Aug** |
| i. | Structural model | 7thth Aug |
| ii. | Behavioural model | 12th Aug |
| iii. | UI design | 23rd Aug |
| iv. | Database design | 29th Aug |
|  |  |  |
| **4.** | **Coding** | **20th Sept** |
| i. | Frontend | 11th Sept |
| ii. | Backend | 20th Sept |
|  |  |  |
| **5.** | **Testing** | **30th Sept** |
| i. | Unit testing | 25th Sept |
| ii. | Blackbox testing | 30th Sept |
|  |  |  |
| **6.** | **Documentation** | **12th Oct** |
| i. | User manual | 7th Oct |
| ii. | Final documentation | 12th Oct |
|  |  |  |
|  | **Final deadline** | **12th Oct, 2019** |

As shown in the figure I will be completing my project by 12th October. I will also finish all the subtasks accordingly their deadlines.

Description of milestone,

* Proposal (5 days)
* Analysis (20 days)
* Planning (2 days)
* Requirement gathering (3 days)
* Feasibility study (4 days)
* Analysis method (4 days)
* SRS (3 days)
* Use case diagram (4 days)
* Design (23 days)
* Structural model (7 days)
* Behavioural model (3 days)
* UI design (9 days)
* Database design (4 days)
* Coding (16 days)
* Frontend (9 days)
* Backend (7 days)
* Testing (6 days)
* Unit testing (3 days)
* Blackbox testing (3 days)
* Documentation (9 days)
* User manual (5 days)
* Final documentation (4 days)

## 4.3 Scheduling

Scheduling is the way of allocating the time of the tasks so that the whole project can be completed in the estimated time. For scheduling I have used project libre which is a project management tool used to schedule the project’s tasks and their sub tasks it helps in maintaining the efficiency in completing the project.

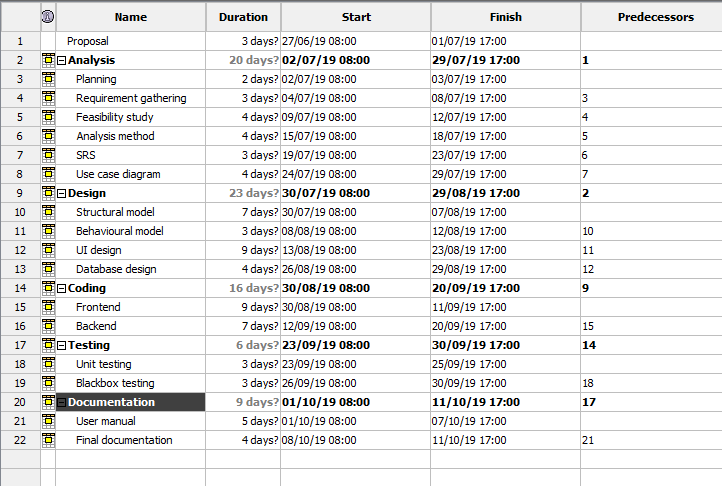


Figure 5: Time scheduling.

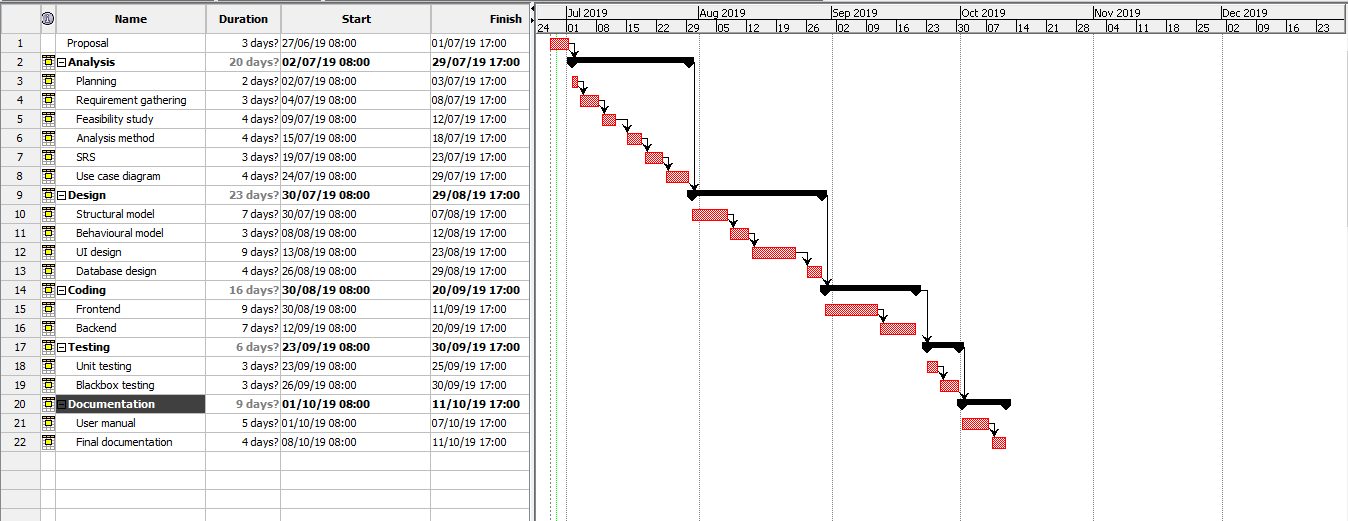


Figure 6: Gantt chart

# 5. Other Project Activities:

## 5.1 Risk Management

Developing system is one of the most difficult part of information technology and as everything developing a system also has its own risks. So, risk management is an approach of identifying risk, assessing risk as well as taking step to reduce risks to an acceptable level. The risk management approaches requires certain tools, team. The risk management plan also describes how the risk management will be structuralized and will be performed on the project.

For calculating process we will use two tables,

1. Likelihood table

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

1. Consequences table

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

Now for the calculation process we calculate the impacts of risks by multiplying likelihood to consequences.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SN | Risks | Likelihood | Consequences | Impact | Actions |
|  | Requirements changes | 1 | 5 | 5 | Should discuss about the requirements with clients as well aa with team before developing |
|  | Malwares | 2 | 5 | 10 | Should use antivirus software and prevent downloading from unknown sites. |
|  | Hard disk crash | 1 | 5 | 5 | Should keep backups. |
|  | Unauthorized access | 2 | 3 | 6 | Password security and data encryption must be done. |
|  | Natural Disaster | 1 | 4 | 4 | A policy regarding the recovery of project. |
|  | Time limitation | 2 | 5 | 10 | Should make schedules for tasks and follow the schedule. |

Here are the impacts which has score more than 6 and their precautions,

1. Malware: Malware also known as virus are the common issues which every system or software faces. It affects the system by corrupting the files and its databases. Some of the preventive measures are as follows,
2. Installing antivirus software’s.
3. Downloading from genuine sites.
4. Backup files.
5. Preventing opening unusual links.
6. Time limitation: This is a really serious issues in system development because if it occurs both client and the developer will lose money as well as time. So some of the preventive measures are as follows,
7. Conducting meeting regarding the time and the project.
8. Following the time protocols.

## 5.2 Configuration management

Configuration management is the task of tracking the changes in a system. Configuration management also helps companies to manage, organize and control the changes in the codes, files or other documents during SDLC.

Therefore, for this project also I have created various files which will be the backup files which would be updated in every 2 to 3 days before the submission and it is uploaded in GitHub. We keep the backup to ensure that if our system gets corrupted we can re-use the system from the backup.

Some of the reasons behind using configuration management are,

* It maintains the coordinate within team.
* It helps in managing project.
* Updates can be seen by any individuals in the team.

For software configuration management I have created the backups in both my local drive as well as in GitHub.

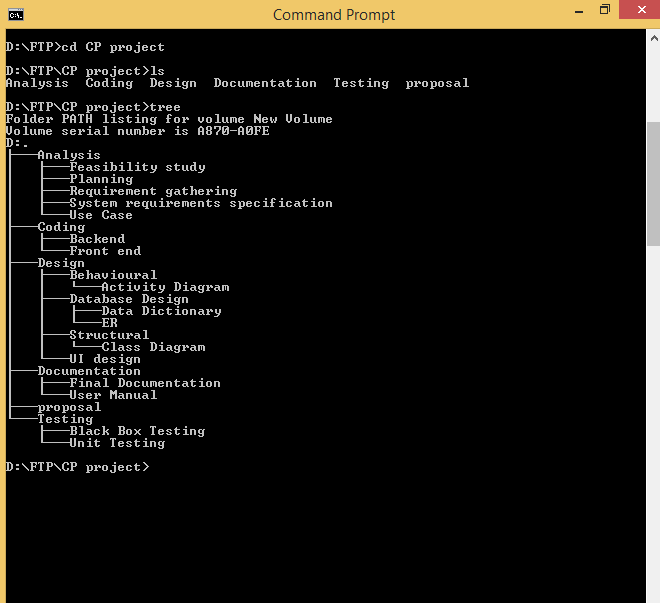


Figure 7: Tree structural of my files in local drive of PC.

I have created a file named CP where all the sub files of the projects are included in the local drive and for online backup I have created an ID in GitHub where my user name is nischal12 on GitHub I have saved all the files of the local drives. Below down is the screen shot of my GitHub files[. (GitHub, 2016)](#_Bibliography)

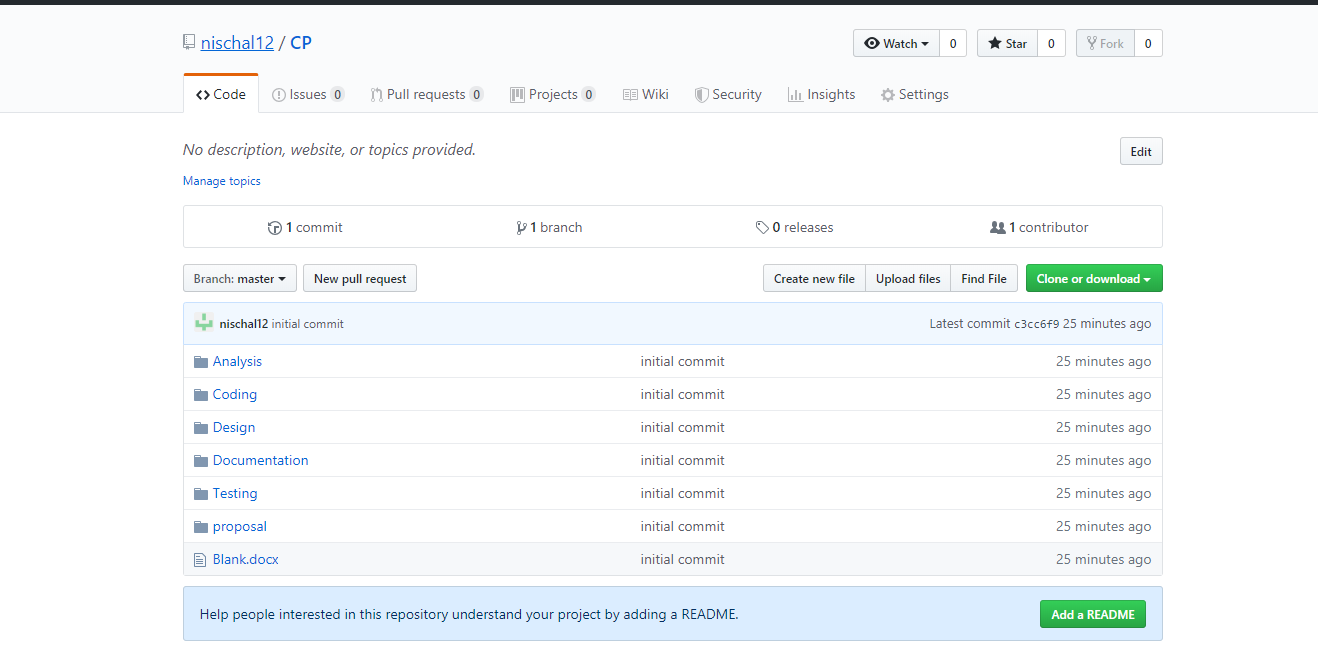


Figure 8: GitHub file structure.

# Conclusion

This project is all about developing a system which will make train ticket booking system easy than the existing one. Likewise WBS has helped us in keeping the project as in the proper way and it also breaks the works which will help us to complete the project according to the allocated time. To maintain the efficiency we have also set up the configuration management. The risks which can occur are also shown in this documents itself.

# Bibliography

GitHub. (2016, 4 7). *GitHub Guides logo*. Retrieved from https://guides.github.com/activities/hello-world/: https://guides.github.com/activities/hello-world/

Rouse, M. (2019, 2). *3-tier-application*. Retrieved from /definition/3-tier-application: https://searchsoftwarequality.techtarget.com/definition/3-tier-application

tutorialspoint.com. (n.d.). *sdlc\_overview.htm*. Retrieved from /sdlc/sdlc\_overview.htm: https://www.tutorialspoint.com/sdlc/sdlc\_overview.htm

workbreakdownstructure. (n.d.). Retrieved from workbreakdownstructure.com - Work Breakdown Structure information.: https://www.workbreakdownstructure.com/