

Samita Gautam

Level 5 diploma in computing

Computing project

Softwarica college of IT

Contents

[Table of figures: 2](#_Toc12916796)

[Chapter 1 introduction: 3](#_Toc12916797)

[1.1 introduction to your system: 3](#_Toc12916798)

[1.2 background to the system: 3](#_Toc12916799)

[1.2.1 Problem statement: 3](#_Toc12916800)

[1.3 overview of the proposed system: 3](#_Toc12916801)

[Chapter 2 Scope: 3](#_Toc12916802)

[2.1 Aim of the project: 3](#_Toc12916803)

[2.2 objective of the project: 3](#_Toc12916804)

[2.3 features included in it: 4](#_Toc12916805)

[2.4 overview of the scope: 4](#_Toc12916806)

[Chapter 3 development methodology: 5](#_Toc12916807)

[3.1 Methodology used in the project: 5](#_Toc12916808)

[3.2 design pattern used: 6](#_Toc12916809)

[3.3 system architecture: 7](#_Toc12916810)

[Chapter 4 scheduling: 8](#_Toc12916811)

[4.1 WBS (work breakdown structure) 8](#_Toc12916812)

[4.2 Description of Milestone: 9](#_Toc12916813)

[4.3Milestone: 11](#_Toc12916814)

[4.3 Gantt chart: 12](#_Toc12916815)

[Chapter 5 14](#_Toc12916816)

[Risk management: 14](#_Toc12916817)

[Chapter 6 16](#_Toc12916818)

[Configuration management: 16](#_Toc12916819)

[Chapter 7 17](#_Toc12916820)

[Conclusion: 17](#_Toc12916821)

# Table of figures:

[Figure 1 Water Fall Methodology 5](#_Toc12916731)

[Figure 2 MVC Design Pattern 6](#_Toc12916732)

[Figure 3 Three Tier Architecture 7](#_Toc12916733)

[Figure 4 Work break Down Structure 9](#_Toc12916734)

[Figure 5 Scheduling 12](#_Toc12916735)

[Figure 6 Gant Chart 13](#_Toc12916736)

[Figure 7 Git Directory 16](#_Toc12916737)

[Figure 8 Project Local Directory 17](#_Toc12916738)

# Chapter 1 introduction:

## 1.1 introduction to your system:

The system is a complete solution beginning with taking orders from customers, and ending with billing and tax reports. The system’s interface is carefully optimized for high speed and easy input for the customer order. Payments can be made through easy medium like cash, credit cards or cheque.

## 1.2 background to the system:

People now a days visit a lot of restaurant and also prefer having their meal there. As many people are visiting the restaurant, the restaurant are busy, also the tables are being reserved. So with the help of this system people will be able to order food online, reserve their table early, also pay their bills online along with many more features.

## 1.2.1 Problem statement:

In the world where people prefer going to restaurant to eat every meal but the restaurant manage their business manually which make their service slow such as taking orders from the customers. This kind of slow service will affect the reputation of the restaurant.

So, this system will help the restaurant to take and record of customers and also show the price of food in the restaurant in more systematic/computerized manner.

## 1.3 overview of the proposed system:

so altogether the system will be able to keep the record of all the food prices, customers order and will also be able to reserve tables for customers. Instead of keep the record and prices of food manually the system will make it easier for the restaurant manager to keep the record in this system.

# Chapter 2 Scope:

The scope of this system is for all kind of restaurant form small to large. Which will make the record of food prices, customers and reservation more managed and computerized so that no common development mistakes are reproduced.

## 2.1 Aim of the project:

the aim of this project is to help the restaurant manager to manage the restaurant more effectively and efficiently by computerizing meal orders, billing the inventory control and to process transaction and store the result data.

## 2.2 objective of the project:

Similarly, the objectives of this system other than to help the manager to manage the restaurant are:

* tracking sales of each item.
* Generating financial statement quickly and accurately.
* Providing better customer service.
* Managing the staff efficiently.
* Keeping a better communication between the kitchen staff, customer and the cashier.
* Saving cost in many sectors.

## 2.3 features included in it:

The system will be capable of doing many things other than just taking order form customers. Some of the features of the restaurant management system have been listed down below:

* Keeping the track of all the sales and also taxes:

This is one of the important features of this system which will allow the staff of a restaurant to keep a track of all the sales they have made in a day also the tax they have paid or have to pay automatically.

* Making the menu setup simpler:

Keeping a set of user-friendly menu setup in the system will make the customer more comfortable. So, this way customer can understand the whole menu and order food respectively.

* User friendly order management:

Even though customer may under the whole menu setup they may not know how to use the system to order food. So, keeping the order management user friendly will help a lot for a better system.

* Inventory control:

The restaurant staff must be able to use the system to track all the goods they have supplied or have been supplied. This system will help them to keep a track of inventory supplies.

* Report of customer data:

The system will also be able to keep a record of customer data and generate a report based on it i.e. the food most ordered by a set of people of a certain age group. And so on.

* Generating the bills for the customers:

This is one of the most basic and must have feature in a RMS. The food order by customer and generating a bill automatically for the customers.

All of the above will be done more effectively by computerizing them.

## 2.4 overview of the scope:

Even though the system may not be able to do all the things required by a restaurant. It is able to do all the basic things needed for a restaurant to improve its efficiency. All of the things are done systematically by computerizing them.

# Chapter 3 development methodology:

## 3.1 Methodology used in the project:

The project I have used waterfall methodology. Waterfall methodology is a linear sequential life cycle model. It one of the simplest methodologies to understand and use. In this methodology there are six phases, so each phase must be completed before moving on to the next phase, there must be no overlapping the phases. I have a diagram shown for a better understand of this model below:

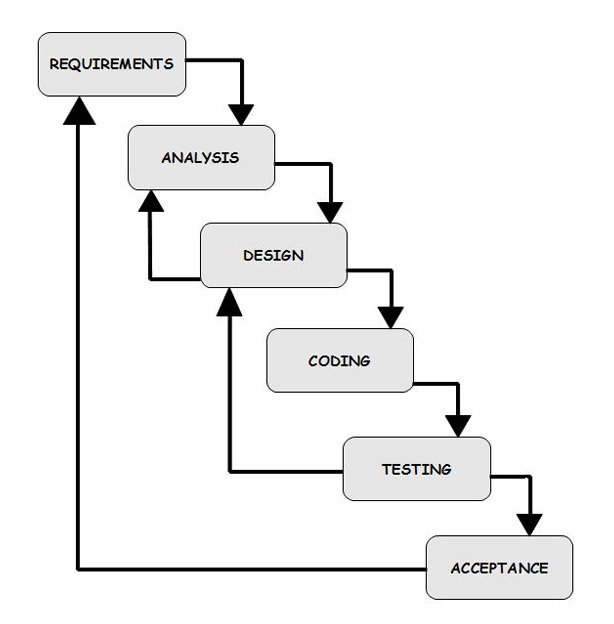


Figure Water Fall Methodology

## 3.2 design pattern used:

Similarly, the design pattern I have used in this project is MVC (Model View Controller). In this design pattern there are three parts model, view and controller. Model contains all the data that might be used in the controller part, basically it is the database part. View contains the UI where the end users are allowed to input data. And the controller part contains the data provided by the users which are manipulated and showed to the users.

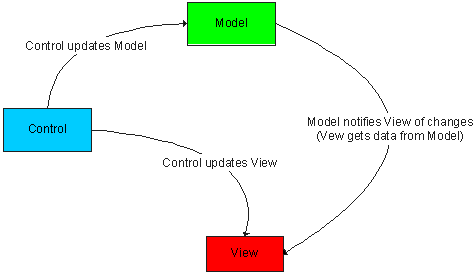


Figure MVC Design Pattern

## 3.3 system architecture:

The system architecture is a generic discipline to handle objects called system, in a way that supports reasoning about the structural properties of these objects. The system architecture is a conceptual design that helps to explain a system view, behavior and its structure. I have used a three-tier architecture for this project. Below is the diagram of three tier architecture for better understanding:

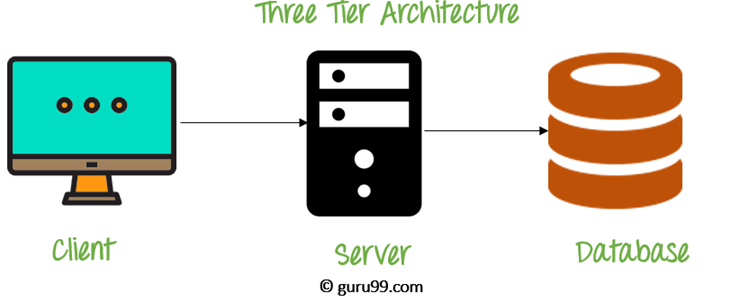


Figure Three Tier Architecture

The three-tier architecture consists of three part:

* Client tier: It is the application clients that access a JAVA EE server and located on a different machine from server.
* Server tier: It is entire application which is distributed across different computing layer.
* Database tier: This is responsible for the database for functional storage, processing, and integrity of data.

# Chapter 4 scheduling:

## 4.1 WBS (work breakdown structure)

A work breakdown structure (WBS) is a key method of getting a complex, multi steps project done. It make sure the project deliverable that organizes to work into manageable sections. “Deliverable oriented hierarchical decomposition of the work to be executed by the project team.”-The Project Management Body of knowledge. WBS can be built by creating folders and subfolders, and can even go further to divide individual tasks into subtasks.

Figure Work break Down Structure

## 4.2 Description of Milestone:

* Proposal (15 Days)

Proposal is the first step to get into the system. In purpose planning is made and can get idea about how to complete and make the system work by following steps which is needed regarding to this. With the help of analyzing risk management, WBS and configuration management we check for four days to make sure if everything is in correct order.

* Analysis (28 days)

Analysis is the second step to be followed to start the project. With requirement analysis, use case and class diagram is done and after that for 8 days analysis specification is completed.

* Design (29 days )

Design should be done before developing a project. Without designing a project cannot be started by doing structure diagram, behavior diagram, UI diagram and database diagram design of the project is made.

* Coding(20 days)

After design here comes the coding part. In this process database required to the system is built with the help of design and later be done in form of coding.

* Testing(9 days )

After coding almost all the things which is need to complete the project testing is done. After testing the system we can know where and what things are going wrong and what can be done for it by using form of testing which is black box testing and black box testing.

* Configuration Management(11 days)

It is the last process to make the system complete. Provide training to the people to make them know how to use it and utilize it in appropriate way and making sure that they are comfortable by using the system and asking for their feedback as well.

## 4.3Milestone:

|  |  |  |
| --- | --- | --- |
| Milestone | Number of days | Date |
| Proposal  Risk management  WBS  Configuration management  Submission | 15  4  3  4  4 | June 15th – July 1st  June 15th – June 19th  June 19th – June 22nd  June 22nd – June 26th  June 26th – July 1st |
| Analysis  Requirement analysis  Use case  Class diagram  Analysis specification | 28  8  6  6  8 | July 2nd – July 29th  July 2nd – 10th  July 10th – 16th  July 16th – 22th  July 22th – July 29th |
| Design  Structure diagram  Behavior diagram  UI diagram  Database diagram | 29  7  7  6  7 | July 30th – august 29th  July 30th – august 7th  August 7th – 14th  August 14th – 20th  August 20th – august 29th |
| Coding  Building database  Coding | 20  10  10 | August 30th – September 20th  August 30th – September 10th  September 10th September 20th |
| Testing  White box testing  Black box testing  Validation testing | 9  3  3  3 | September 21st – September 30th  September 21st – September 24th  September 24th – 27th  September 7th – September 30th |
| Configuration management  User interaction  reports | 11  5  6 | September 31st – October 12th  September 31st – October 5th  October 5th – October 12th |
| Total days | 109 | June 15th – October 12th |

Table table of Milestone.

## 4.3 Gantt chart:

A Gantt chart is the popular most commonly used in project management and it’s a useful ways of showing activities such as tasks or events displayed against time. On the left of the chart is a list of the activities and along the top is a suitable time scale. Each activity is represented by a bar; the position and length of the bar reflects the start date, duration and end date of the activity. This allows you to see at a glance.

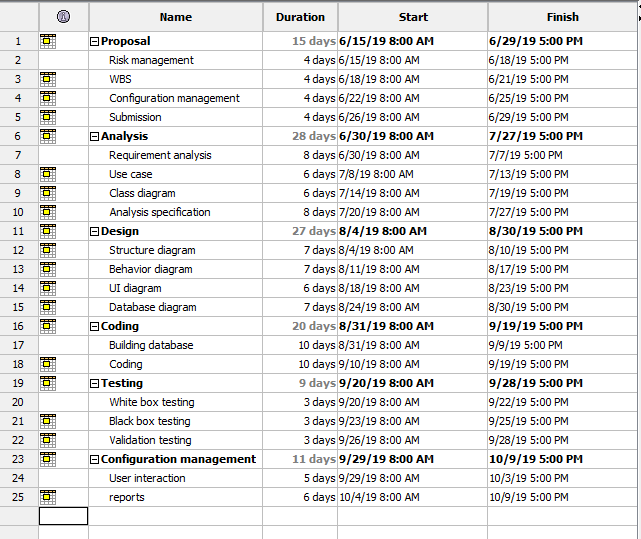


Figure Scheduling

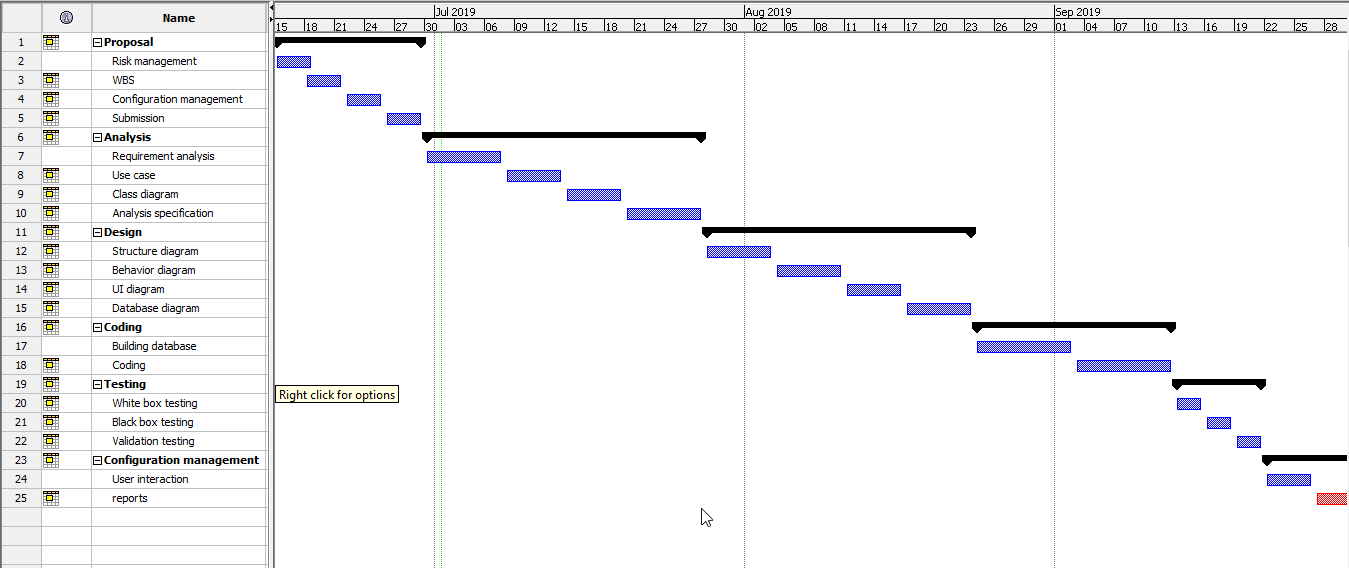


Figure Gant Chart

# Chapter 5

# Risk management:

Risk is part of all our lives. Risk management is the process of understanding, analyzing and addressing risk to make sure to achieve their objectives. It is managed proactively, optimizing success by minimizing threats and maximizing opportunities. Risk is defined at two level for the project, program and portfolios. Risk management must be closely aligned to schedule management. Cost, time and resource estimates should take risks into account.

**Impact = Likelihood \* Consequences.**

Risk likelihood values:

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

Table risk likelihood values.

risk consequences values:

|  |  |
| --- | --- |
| Consequences | Values |
| Very low | 1 |
| Low | 2 |
| Medium | 3 |
| High | 4 |
| Very high | 5 |

Table risk consequences values

Risk management:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S.N. | Risk | Likelihood | Consequences | Impact | Action |
| 1 | Limited time and financial | 2 | 5 | 10 | Should properly estimate the time and financial by making plan consulting with the organization. |
| 2 | Lack of training knowledge | 2 | 4 | 8 | Training should be provided to the developer to handle problems. |
| 3 | Malware/virus | 2 | 5 | 10 | Anti -virus software should be installed. |
| 4 | Project out of track | 1 | 4 | 4 | Analysis should be done properly. |
| 5 | User Requirement Not reach | 1 | 5 | 5 | User requirement analysis should be done. |
| 6 | Crash of Hardware | 2 | 4 | 8 | Back up should be done. |

Table risk management

# Chapter 6

# Configuration management:

Configuration management is set of characteristics which is important foundation that define a final product or deliverable. It is a governance and system engineering which includes all functional and physical specifications. The configuration management process seeks to identify and track individual configuration items (CIs), documenting functional capabilities and interdependencies. It is also an automated method for maintaining computer system and software which is called consistent state.

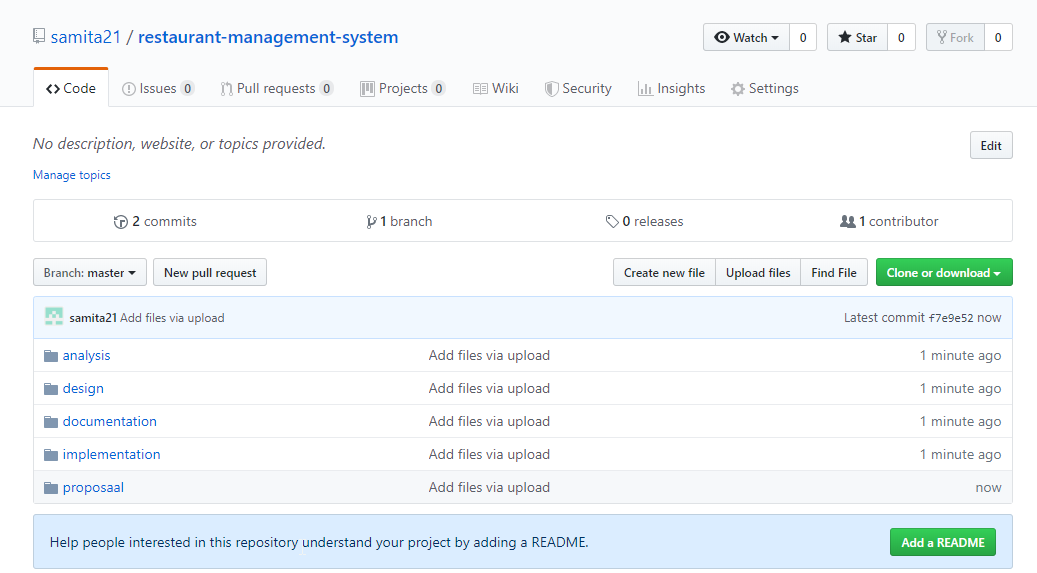


Figure Git Directory

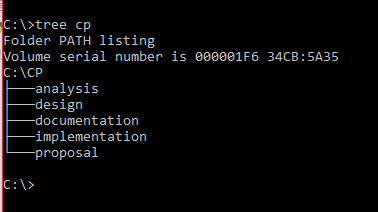


Figure Project Local Directory

# Chapter 7

# Conclusion:

In this system, which is entitled as The Restaurant Management System has been overviewed above. The introduction, justification, problem statement, its benefits and methodology which are important to finish this project have been maintained briefly. This project wouldn’t have been completed with the help of the waterfall methodology, MVC design pattern, and to complete the project time have been divided with the help of the Gantt chart. Problems and risks which have been mention in the project have also discovered the project to get over it. Configuration Management is also done to make a better software with complete efficiency.

# References

(2019), I., 2018. *Software Development Methodologies..* [Online]   
Available at: http://www.itinfo.am/eng/software-development-methodologies/  
[Accessed june 2019].

Codecademy., 2011. *MVC: Model, View, Controller.* [Online]   
Available at: https://searchsoftwarequality.techtarget.com/definition/3-tier-application  
[Accessed june 2019].

Jamsheer, K., n.d. *12 Best Software Development Methodologies with Pros & Cons..* [Online]   
Available at: https://acodez.in/12-best-software-development-methodologies-pros-cons/  
[Accessed june 2019].