Restaurant Management System

Bachelor of
Engineering in
Computer Science Engineering
(Session 2018 – 2019)

Guided By:

Name of Project Guide

Submitted By:

Deepankar Ratra

Department of Computer Science Engineering

SRM Institute of Science & Technology
(http://www.srmuniv.ac.in)

April 2018

Dissertation Approval Sheet

The dissertation entitled **"Restaurant Management System" submitted** by **Deepankar Ratra** is approved as partial fulfilment for the award of **Bachelor of Engineering in Computer Science Engineering** degree by **SRM Institute of Science and Technology**.

Internal Examiner **External Examiner**

Director
SRM Institute of Science & Technology
Delhi NCR.

Candidate Declaration

We hereby declare that the work which is being presented in this project entitled Restaurant Management System in partial fulfilment of degree of Bachelor of Engineering in Computer Science Engineering is an authentic record of our own work carried out under the supervision and guidance of Mr. Abhinandan Chivate.

We are fully responsible for the matter embodied in this project in case of any discrepancy found in the project and the project has not been submitted for the award of any other degree.

Date:
Place:

Deepankar Ratra

ACKNOWLEDGEMENT

The internship opportunity I had with **Cognizant Technology Solutions** was a great chance for learning and professional development. Therefore, I consider myself as a very lucky individual as I was provided with an opportunity to be a part of it. I am also grateful for having a chance to meet so many wonderful people and professionals who led me though this internship period.

Bearing in mind previous I am using this opportunity to express my deepest gratitude and special thanks to the **Mrs. Shilpa Mahajani** at CTS, Pune who in spite of being extraordinarily busy with her duties, took time out to hear, guide and keep me on the correct path and allowing me to carry out my project at their esteemed organization and extending during the training.

Deepankar Ratra

ABSTRACT

Cognizant is an American multinational corporation that provides IT services, including digital, technology, consulting, and operations services. It is headquartered in Teaneck, New Jersey,United States of America. Cognizant is included in the NASDAQ-100 and the S&P 500 indices. It is also one of the fastest growing Fortune 500 companies. It was founded as an in-house technology unit of DUN & BRADSTREEIN in 1994, and started serving external clients in 1996.

Cognizant had a period of fast growth during the 2000s, becoming a Fortune 500 company in 2011. In 2015, the FORTUNE Magazine named it as the world's fourth most admired IT Services company. In 2017, Cognizant was named in Fortune's Future 50 list.

Cognizant provides information technology, information security, consulting, ITO and BPO services. These include business & technology consulting, system integration, application development & maintenance, IT infrastructure services, analytics, business intelligence data , warehousing, customer relationship management, supply chain management, engineering & manufacturing solutions, enterprise resource planning, research and development, outsourcing, and testing solutions.

Cognizant has three key practice areas that span its business — Digital Business, Digital Operations, and Digital Systems & Technology.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1. Introduction

1.1 Purpose of this document

This document is aimed at:

- Providing the necessary inputs to the detailed requirements gathering phase and further on for the SDLC processes.

The purpose of this document is to systematically capture requirements for the project and the system to be developed. Functional requirements are captured in this document.

1.2 Project Overview

1.2.1 Objectives

Below are the objectives that shall be fulfilled post the execution of this project:

- Admin registration & credential authentication
- Management of staff.
- Management of Inventory.

CHAPTER 2

LITERARY SURVEY

2. Literary Survey

2.1Methodology

➢ SDLC:

In software engineering, a software development process is the process of dividing software development work into distinct phases to improve design, product management, and project management. It is also known as a **software development life cycle**. The methodology may include the pre-definition of specific deliverables and artifacts that are created and completed by a project team to develop or maintain an application.

Most modern development processes can be vaguely described as **agile**. Other methodologies include *waterfall, prototyping, iterative and incremental development, spiral development, rapid application development, and extreme programming.*

Some people consider a life-cycle "model" a more general term for a category of methodologies and a software development "process" a more specific term to refer to a specific process chosen by a specific organization.

➢ Agile:

"Agile software development" refers to a group of software development methodologies based on iterative development, where requirements and solutions evolve via collaboration between self-organizing cross-functional teams

Agile software development uses iterative development as a basis but advocates a lighter and more people-centric viewpoint than traditional approaches. Agile processes fundamentally incorporate iteration and the continuous feedback that it provides to successively refine and deliver a software system.

There are many agile methodologies, including:

- Dynamic systems development method (DSDM)
- Kanban
- Scrum

➢ Client–server model:

Client–server model is a distributed application structure that partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters, called clients.[1] Often clients and servers communicate over a computer network on separate hardware, but both client and server may reside in the same system. A server host runs one or more server programs which share their resources with clients. A client does not share any of its resources, but requests a server's content or service function. Clients therefore initiate communication sessions with servers which await incoming requests.

2.1Technology and Tools

➢ Front End:

▪ Java (HTML, CSS, JavaScript, Bootstrap)

• HTML:

❖ **Hypertext Markup Language** (**HTML**) is the standard markup language for creating web pages and web applications.
With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web.

❖ Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

• CSS:

❖ **Cascading Style Sheets** (**CSS**) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

❖ CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts.[3] This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

- **JavaScript:**

❖ **JavaScript** often abbreviated as **JS**, is a high-level, interpreted programming language that conforms to the ECMAScript specification. It is a programming language that is characterized as dynamic, weakly typed, prototype-based and multi-paradigm.

❖ Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it, and major web browsers have a dedicated JavaScript engine to execute it.

- **Bootstrap:**

❖ **Bootstrap** is a free and open-source front-end web framework. It contains HTML and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions. Unlike many earlier web frameworks, it concerns itself with front-end development only.

➢ **Middleware:**

▪ **Java (Spring MVC, Hibernate MVC, Java Servlet)**

- **Spring MVC:**

❖ The **Spring Framework** is an application framework and inversion of control container for the Java platform. The framework's core features can be used by any Java application, but there are extensions for building web applications on top of the Java EE (Enterprise Edition) platform. Although the framework does not impose any specific programming model, it has become popular in the Java community as an addition to, or even replacement for the Enterprise JavaBeans (EJB) model. The Spring Framework is open source.

- **Hibernate MVC:**

- **Hibernate ORM** (Hibernate in short) is an object-relational mapping tool for the Java programming language. It provides a framework for mapping an object-oriented domain model to a relational database. Hibernate handles object-relational impedance mismatch problems by replacing direct, persistent database accesses with high-level object handling functions.
- Hibernate is free software that is distributed under the GNU Lesser General Public License 2.1.
- Hibernate's primary feature is mapping from Java classes to database tables, and mapping from Java data types to SQL data types. Hibernate also provides data query and retrieval facilities. It generates SQL calls and relieves the developer from the manual handling and object conversion of the result set.

- **Java Servlet:**

- A Java servlet processes or stores a Java class in Java EE that conforms to the Java Servlet API,a standard for implementing Java classes that respond to requests. Servlets could in principle communicate over any client–server protocol, but they are most often used with the HTTP.
- Thus "servlet" is often used as shorthand for "HTTP servlet". Thus, a software developer may use a servlet to add dynamic content to a web server using the Java platform. The generated content is commonly HTML, but may be other data such as XML and more commonly, JSON. Servlets can maintain state in session variables across many server transactions by using HTTP cookies, or URL mapping.

➢ **Backend:** *{can run on any database}*

- **Oracle/SQL Server**

- **MySQL:**

- **MySQL** is an open source relational database management system (RDBMS). "SQL", is abbreviation for Structured Query Language.

- MySQL is free and open-source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation).

CHAPTER 3

ANALYSIS

3. Analysis

3.1 Software Requirements

- Operating System: Linux OS, Windows 7/8/10
- IDE: Eclipse IDE for Java EE Developers (Oxygen)
- Server: MySQL WorkBench Server 6.2, Tomcat 8.5
- RDBMS: MySQL
- Environment: JDK 1.6, 1.7, 1.8 for Java 6, 7, 8 configured on the workstation

3.2 Hardware Requirements

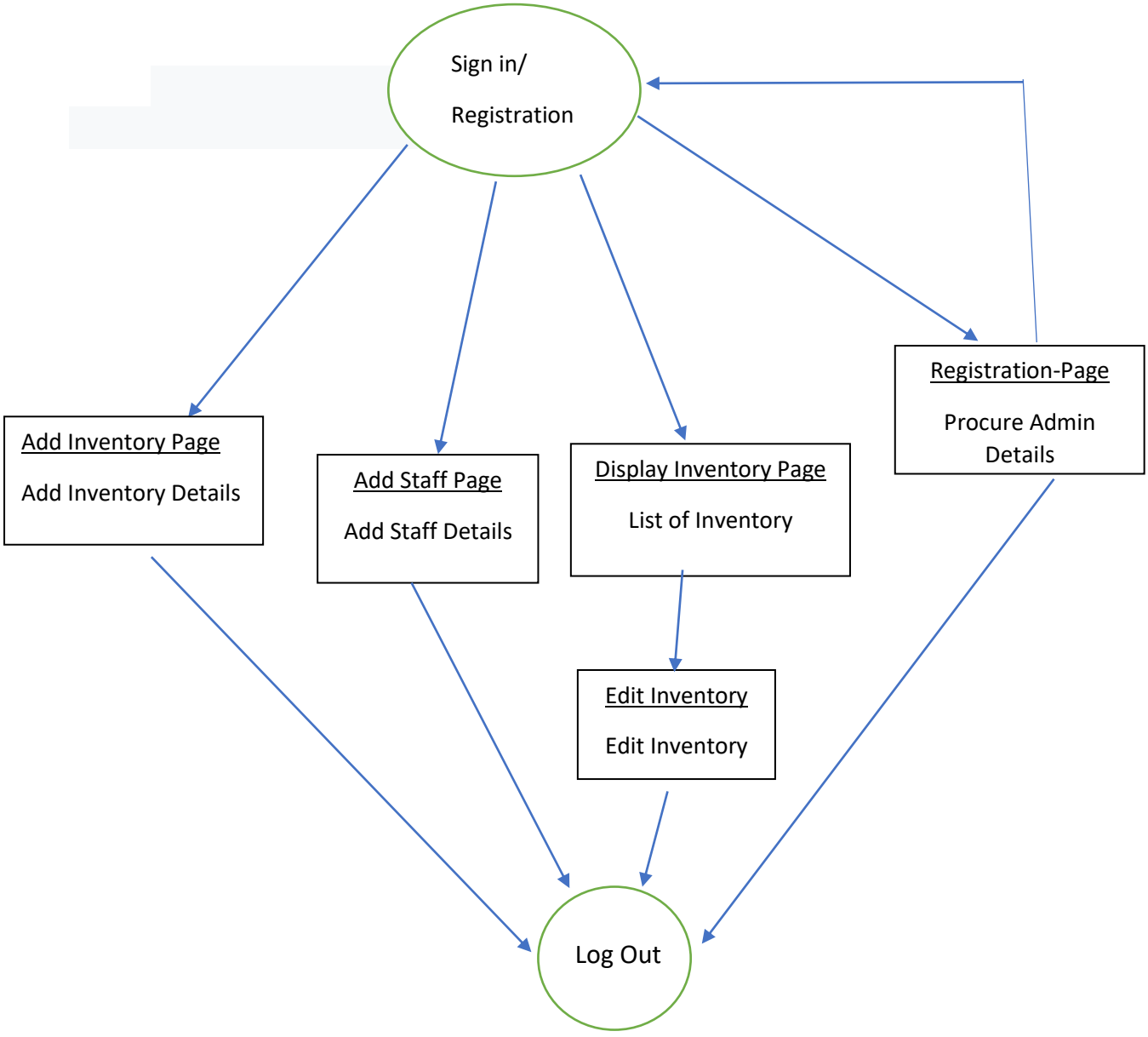
- Processor: 1.7GHz Intel Core2Duo or above
- RAM: 4 GB
- Hard Disk: 100 GB-1 TB
- Network Adaptor

CHAPTER 4

DESIGN

4. Design

4.1 Diagrams



4.2 Tables

ADMIN

Field Name	Field Type	Data Type	Mandatory	Possible Values
User ID	Text (10)	Varchar	Yes	
UserName	Text (20)	Varchar	Yes	
Password	Text (20)	Varchar	Yes	

Table 1

STAFF

Field Name	Field Type	Data Type	Mandatory	Possible Values
First Name	Text (50)	Varchar	Yes	
Last Name	Text (50)	Varchar	Yes	
Age	Numeric (2)	Integer	Yes	
Gender	Drop Down	NA	Yes	Male/Female
Address	Text (50)	Varchar	Yes	
Staff ID	Text (20)	Varchar	Yes	

Table 2

INVENTORY

Field Name	Field Type	Data Type	Mandatory	Possible Values
Inventory ID	Text (50)	Varchar	Yes	
Inventory Name	Text (10)	Varchar	Yes	
Creation Date	Text (10)	Date	Yes	
Start Date	Text (10)	Date		
End Date	Text (10)	Date	Yes	
Total Stock	Text (10)	Integer	Yes	

Table 3

CHAPTER 5

CONCLUSION

5.Conclusion

The restaurant management system guided us through the proper architecture of a management system, making us learn the structured levels of development. Being new to the software, a little problem was faced while writing the complete code from scratch however learning the new technologies in order to build the project has levelled up the making process of management system. As a limitation of project, there could be a few more functionalities added to enhance the system and similarly a better architecture can be used to make it smoother. But whatever is made, it has surely cleared our basics and taught us a lot. Concluding to it, we look forward to enhance the features of the system as well as expanding the possible opportunities that come its way.