SYSTEM DESIGN DOCUMENT

# 1. Introduction

RestaurantFinder is a web site for users who wants to make a easily reservation for the restaurants which they want. This project allows users easily find a restaurant and make reservation for it. In this project, users can access restaurants’ informations and they can look theirs address,telephone,pictures and descriptions.Also, users can keep their past reservations and can follow them.

## Purpose of the System

The purpose of this project is to create a web site for making reservation for restaurants. But, users must register and login the system to make reservation. After login the system, they can make reservation, also they can cancel or edit their reservation. Further, this project have users profile page and users can enter their profile page and look past reservations. Also, they can discover new restaurants by category or city without logging the system and access their telephones, photos, addresses and descriptions. In addition, restaurant owners can arrange their restaurant page.

## 1.2. Design Goals

The goal of the project is to satisfy the functional and non functional requirements as specified in the requirements analysis document. This project provide manage restaurants by categories and by cities. Users can search on the web site by a category name or a city name. After they login the system, they can make reservation to restaurants effectively and can look their past reservations on their profile page.

## Definitions, Acronyms, and Abbreviations

UI: User Interface

PHP: Hypertext Preprocessor

SDD: System Design Document

SDD is used to define interfaces between teams of developers and serve as a reference when architecture-level decisions need to be revisited.

Sections of the SDD are described:

* Current Software Architecture

Describe the architecture of the system being replaced

* Overview

Present a bird’s-eye view of the software architecture and briefly describes the assignment of functionality to each subsystem.

## System Decomposition

Describe the decomposition into subsystems and the responsibilities of each.

## Hardware Software Mapping

Describe how subsystems are assigned to hardware and off-the-shelf components

## Persistent Data Management

Describe the persistent data stored by the system and the data management infrastructure required for it.

## Access Control and Security

Describe the user model of the system in terms of an access matrix.

## Global Software Control

Describe how the global software control is implemented.

## Boundary Conditions

Describe the start-up, shutdown, and error behavior of the system.

# Subsystem Services

Describe the services provided by each subsystem.

**1.4. References**

## 3.2. System Decomposition

-USER INTERFACE SUBSYSTEM

User Interface Subsystem is responsible for all interaction between the users.  The UI is decomposed into a set of PHP pages loaded and displayed in a web-browser.  All entry of data, modification of data, and deletion of data will be handled directly through the UI.

-MAIN SUB-SYSTEM

Main Subsystem is responsible for interactions between the Restaurant Finder Controller class and the Cities, User, and Find Restaurant classes. The Main Subsystem is also responsible for the interactions between the Cities, User, Find Restaurant, and Restaurant Finder Controller classes to the Database Subsystem.

-DATABASE SUB-SYSTEM

Database Subsystem is responsible for all insertion, reading, updating and deletion of all data by all classes within the Main, Restaurant, Reservation, Form, Admin Panel Subsystems.  No data will be accessed in any way except through and by the Database Subsystem.  Each class from within the Subsystems described above are responsible for inserting, updating and modifying their own data by utilizing the corresponding methods belonging to the Database Class within the Database Subsystem.

-RESTAURANT SUB-SYSTEM

Restaurant Subsystem is responsible for all interactions involving restaurants.  All interactions with any restaurant will be handled through and by the restaurant list class. The Restaurant Subsystem is also responsible for the interactions between the Restaurant List and Restaurants classes to the Database Subsystem.  Both the Restaurant List and Restaurants class will each be responsible for utilizing the appropriate and necessary methods within the Database class to insert, read, update and delete all corresponding data from within each of the classes themselves.

-RESERVASYON SUB-SYSTEM

Reservation Subsystem is responsible for all interactions involving reservation.  All interactions with any reservation will be handled through and by the Reservation List class. The Reservation Subsystem is also responsible for the interactions between the Reservation List and Make Reservation, Cancel Reservation, Edit Reservation classes to the Database Subsystem.

-FORM SUB-SYSTEM

Form Subsystem is responsible for all interactions involving reservation.  All interactions with any form will be handled through and by the Form List class. The Form Subsystem is also responsible for the interactions between the Form List and Approve Request Form, Reject Request Form classes to the Database Subsystem.

-ADMIN PANEL SUB-SYSTEM

Admin Panel Subsystem is responsible for all interactions involving admin.  All interactions with any form will be handled through and by the Admin Operations class. The Admin Panel Subsystem is also responsible for the interactions between the Admin and Admin Operations classes to the Database Subsystem.

# 4. Subsystem Services

-USER INTERFACE SUBSYSTEM

* Takes user input
* Provides access login page, register page,search restaurant page,restaurant detail page

-MAIN SUB-SYSTEM

- Receives user queries through the UI subsystem

* Sends system responses to the UI subsystem
* Handles cities.
* Provides find restaurant.

-DATABASE SUB-SYSTEM

* Receive update requests from different subsystems
* Sends object information to different requesting controllers

-RESTAURANT SUB-SYSTEM

* Provides restaurants address,telephone,description information
* Provides restaurants city and category information

-RESERVASYON SUB-SYSTEM

* Provides make reservation
* Provides cancel reservation
* Provides edit reservation

-FORM SUB-SYSTEM

* Provides approve request form
* Provides reject request form

-ADMIN PANEL SUB-SYSTEM

* Provides admin operaions

**5. References**