SANGFood

System Design

1.0

11.12.2016

Ayşe Naz Aydemir
Gamze Saraç
Nazlı Akdağ
Selen Kozanoğlu

Prepared for
SE301 Software Engineering

IŞIK UNIVERSITY
COMPUTER
SCIENCE AND
ENGINEERING

SANGFood

SANGFood

Table of Contents

SANGFood

SYSTEM DESIGN DOCUMENT

1. Introduction

SANGFood is a web site for users who wants to make an order for the restaurants which they want. This project allows users easily find a restaurant and make order for it. In this project, users can access find specific food or restaurants and restaurants' food/menu. Also, users can keep their past orders and can follow them.

1.1. Purpose of the System

The purpose of this project is to create a web site for online making food order from restaurant. But, users must register and login the system to make an order. After login the system, they can make search about what they want to order and which restaurant they want order from, also they can cancel or edit their order. Further, this project have users profile page and users can enter their profile page and look past orders. Also, they can discover new restaurants by category or city without logging the system and access their menu/food. In addition, restaurant owners can arrange their restaurant page.

1.2. Design Goals

The goal of the project is to satisfy the functional and nonfunctional 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 order from restaurants effectively and can look their past orders on their profile page.

1.3. Definitions, Acronyms, and Abbreviations

JDBC: Java Database Connectivity.

JSP: JavaServer Pages (JSP) is a server-side programming technology that enables the creation of dynamic, platform-independent method for building Web-based applications.

SDD: System Design Document.

MVC: Model View Controller

1.4. References

-Requirements Analysis Document (20.11.2016)

2. Current Software Architecture

Describe the architecture of the system being replaced**. If there is no previous system**, this section can be replaced by **a survey of current architectures for similar systems**. The purpose of this section is to make explicit the background information that system architects used, their assumptions, and common issues the new system will address.

3. Proposed Software Architecture

3.1. Overview

In SANGFood, the our system has 12 subsystems like Customer Interface, Manager Interface, Admin Interface, Restaurant Owner Interface, are the subsystems for the 'Presentation Layer' which they are in Interface Subsystem. Subsystems for the 'Business Layer' are, Login Subsystem, System Management Subsystem, Restaurants Subsystem, Registration Subsystem, Update Info Subsystem, Ordering Subsystem, and Customer Info Subsystem. Finally, subsystem for the 'Data Layer' is Data Access.

Interface Subsystem

- **Customer Interface:** Provides services for users for common interfaces, it contains Login Form, Edit Profile Form, Password Change Form, and etc. That is to say, Customer Interface provides services to display common forms such as login, password change to all users.

- **Restaurant Owner Interface:** Provides services for restaurant owners for adding new food in their restaurants or any changing to their restaurant like editing address or changing menu.

- **Manager Interface:** Provides services for manager to Accept Restaurants form and send information to admins about new restaurant.

- **Admin Interface:** Provides services to display admins' forms such as Create Username and Password for new restaurants, Add Restaurant Form, Form and so on.

Login Subsystem

Provides services for users (customer, restaurant owner,manager and admin) to login.

Update Info Subsystem

Provides services for users (customer, restaurant owner) to update their information and to change password.

Ordering Subsystem

Provides services for customers to make order.

Registration Subsystem

Provides services for user (customer) to register to the system.

System Management Subsystem

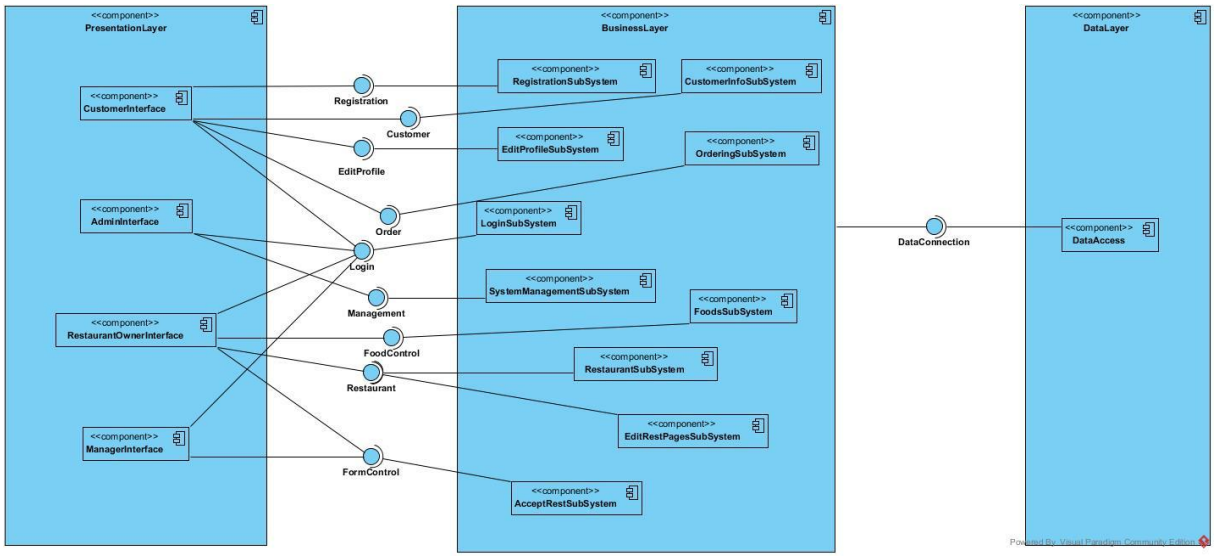
Provides services for admin to add restaurant, accept/reject restaurant, and delete restaurant.

Data Access Subsystem

Contains all our persistent objects, this part could be called Model of MVC.

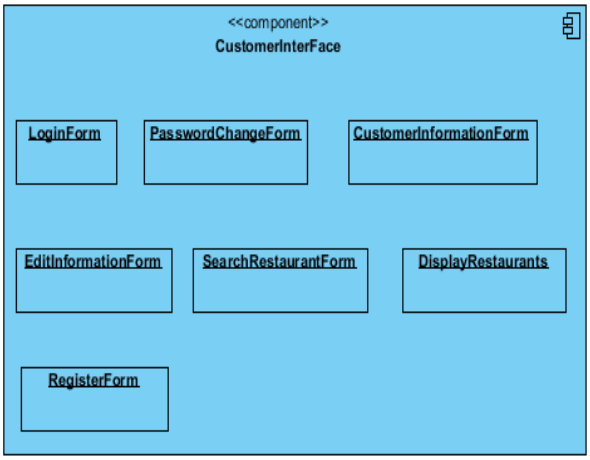
SANGFood

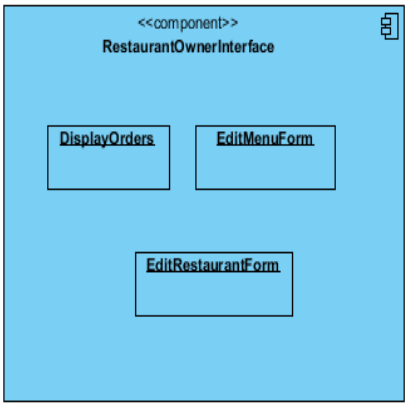
3.2. System Decomposition

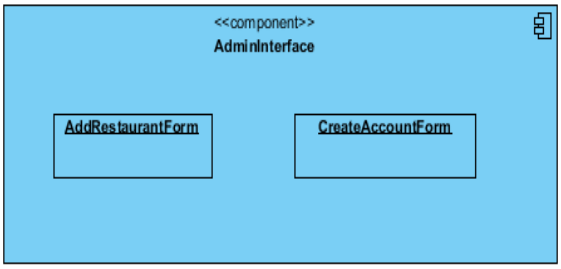


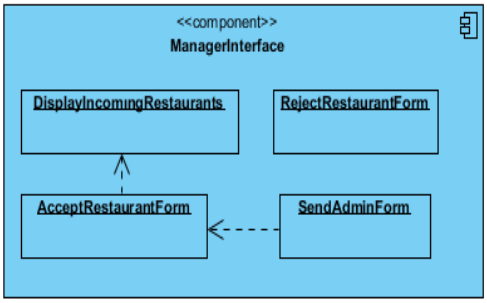
SANGFood system decomposition as shown as the figure. System subsystems are ; Customer Interface, Admin Interface, Manager Interface, RestaurantOwner Interface, Login SubSystem, Registration SubSystem, Ordering SubSystem, EditProfile SubSystem for customers, Foods SubSystem, Restaurant SubSystem, Editrestaurantpage SubSystem, Accept SubSystem, Data Access. Customer Interface contains Registration,CustomerInfo, Edit Profile, Order, Login. Admin Interface contains Login and SystemManagement. RestaurantOwner Interface contains Login, Food Control, Form Control, Restaurant. Manager Interface contains Login and Form Control.. Login Subsystem provides services for users (customer, restaurantowner, manager and admin) to login, it contains Login Subsystem object. EditProfile Subsystem provides services for user (customer) to update their information and to change password, it contains. Order Subsystem provides services for user (customer) to food ordering. Food Subsystem provides services for user (ResturantOwner) to checks the order that the customer sends. Restaurant Subsystem provides services for user (ResturantOwner) to contains restaurant. EditRestPage Subsystem provides services for user (ResturantOwner) to contains restaurant. AcceptRest Subsystem provides services for users (ResturantOwner and manager) to contains formcontrol. Data Access Subsystem; contains all our persistent objects.

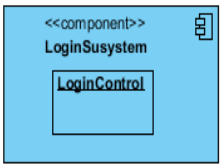
SANGFood
Subsystem Interaction:

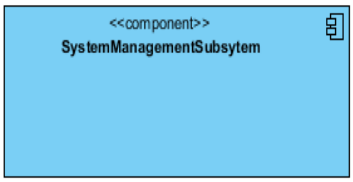


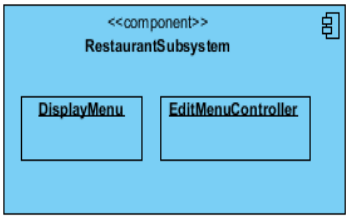


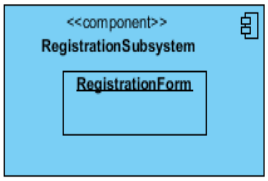


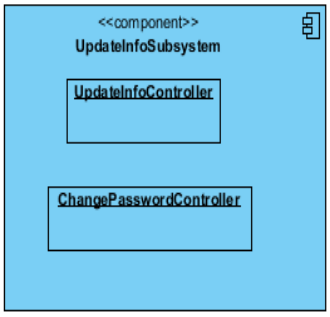


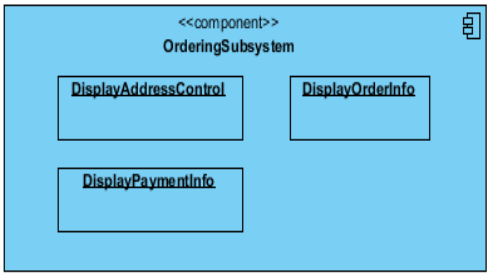


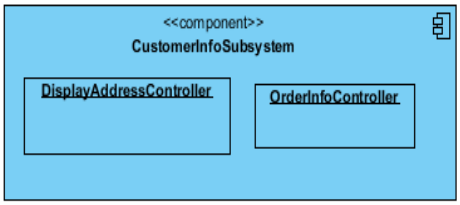




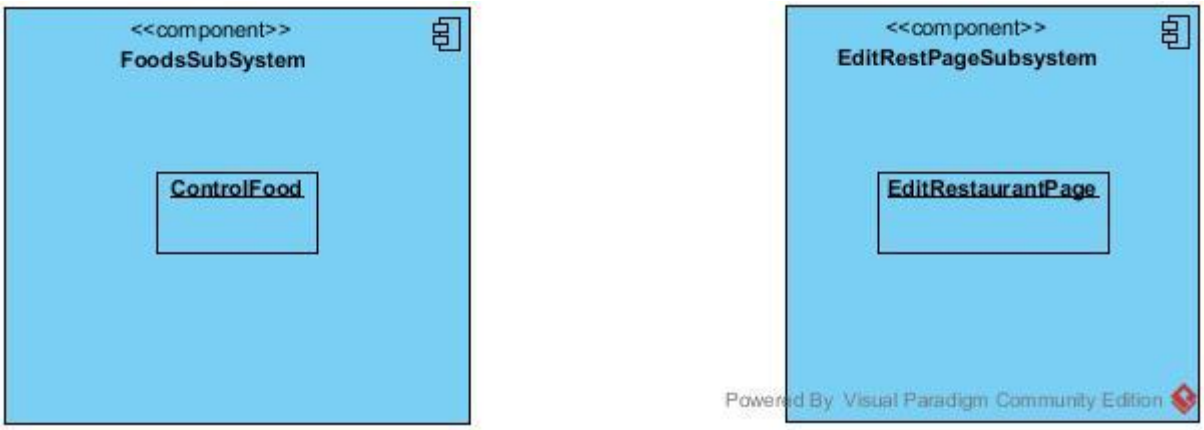


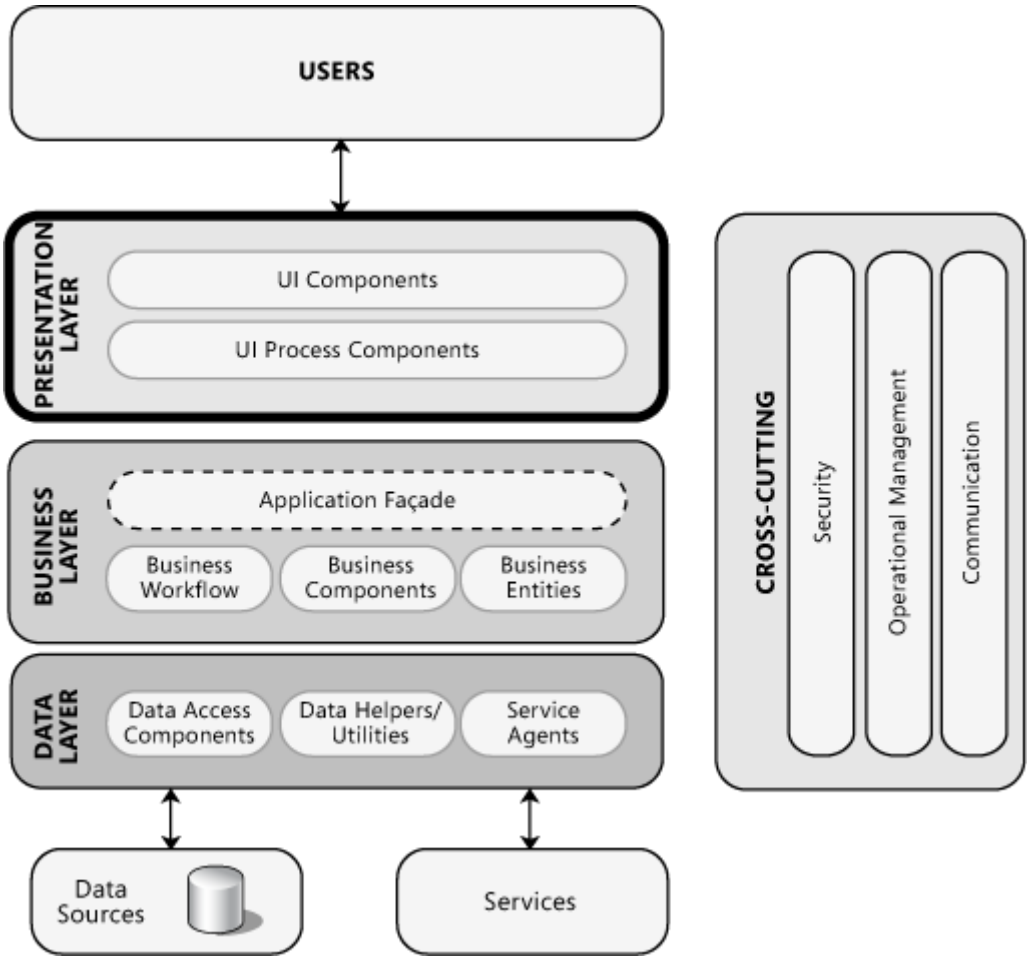






SANGFood





Figure’ references shown as References part.(2)

SANGFood

3.3. Hardware Software Mapping

SANGFood is web-based online food ordering system from restaurants for users. SANGFood is connecting to web server when user tries to visit and login in the system. SANGFood system connects to web host with http protocol. Multiple users can access web server simultaneously. Web host have web server. Web server is written with Java. The system connects to database by using Javax libraries. Users that customer, restaurant owner, Admin and Manager which just make an order , use the SANGFood system. These actors communicate with SANGFood system with http-https protocol web browser in their personal computers. Furthermore SANGFood database server using MySQL for communicating.

Database Server

- MySQL

SANGFood

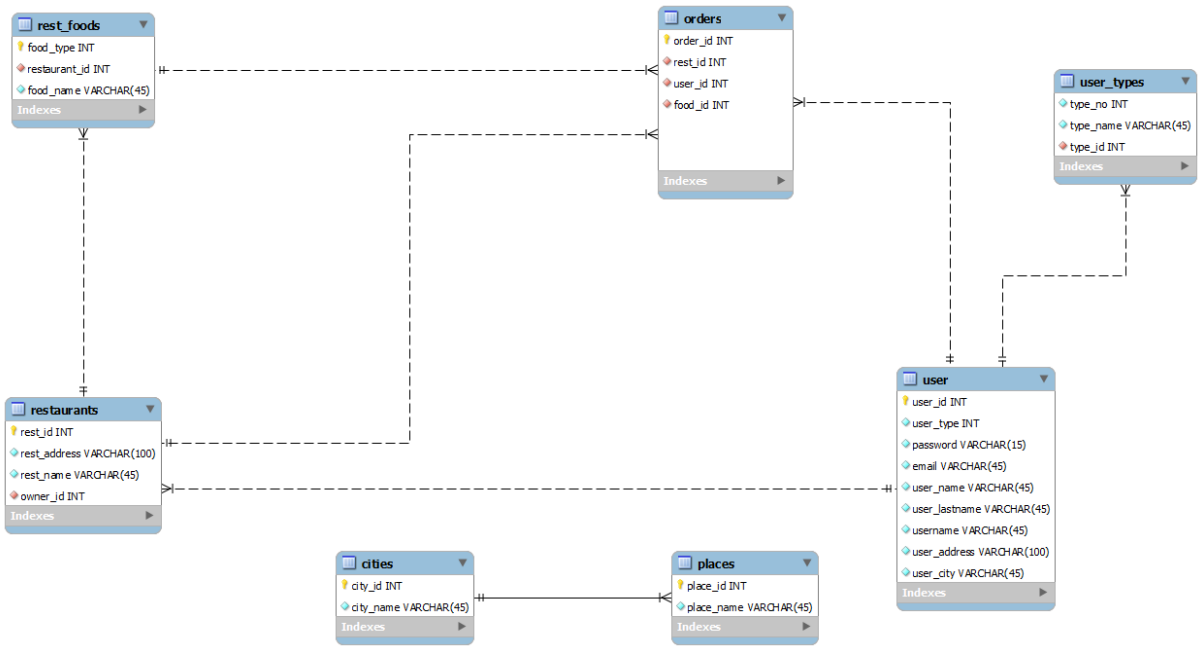
- Web Server
- Apache Tomcat

User

- Web Browser

3.4. Persistent Data Management

All data are to be stored. Based on our application database for user's needs. In our database of user information, order, available storage and comments of the food details. Tables was created according to the needs such as update, add, delete such system. Tables can be changed according to the application stage.



user table:

User table has user id as primary key, user type for admin or restaurant owner etc. password and username also users name , surname, email and addresses.

SANGFood

user types table:

User types table has type no , type name and type id.

restaurant table:

Restaurant table has rest id as primary key, restaurant address, restaurant name and also restaurants owner.

orders table:

Orders table has 3 different foreign key from other tables, they are restaurant id , user id and food id and also order id for primary key.

restfoods table:

Restfoods has foodtype as primary key, restaurant id as foreign key and food name.

cities table:

Users cities hold in another table.

places table:

Users places hold in another table.

3.5. Access Control and Security

Authentication and Security:

SANGFood system is designed to the user to make food orders. For this reason, systems need some personal information of the user such as the user's name ,surname and phone number and payment info. However, this user's personal information should not be seen by someone else.

While we were designing and programming the system, we decided to keep this saved article everyone on their phone, not the remote. This provides security to the project.

Authorization:

The system has database connection for first login. According to login user types system decides which actor will login the site by looking the actor type from database table. System has four sub database classes for actors which are database Restaurant Owner, Customer, Manager and Admin. Each class has specific attributes and methods for actor.. This actor login system is controlled with user_type field in database

Objects / Actors	Customer	RestaurantOwner	Admin	Manager	Restaurants	Orders	Menu	DbConnection
Customer	ChangePassword() ChangeInfo()	-	-	-	ListRestaurant() searchRestaurantsByPlace() searchRestaurantsByCity()	getFood() getDrinks() getMenu() AddMenu() AddDrink() AddFood()	getMenu() view()	Login() Logout() Register()
Restaurant Owner		EditRestauranProfile() ChangePassword() EditMenu() viewOrders()	-	-	EditRestaurant()	View()	setMenu() AddMenu()	Login() Logout() Register()
Admin			ChangePassword() AddRestaurant()		AddRestaurant()	-	-	Login() Logout()
Manager				AcceptRestaurant() RejectRestaurant() Edit() ChangePassword() DeleteRestaurant()	AcceptRestaurant() RejectRestraurant()		-	Login() Logout()

The Access control Matrix for the Restaurant Owner System is as follows:

3.6. Global Software Control

Our system has MVC (Model - View - Controller) software architecture. SANGFood is thread safety but also multithreaded program either because our system must provide many users at the same time to order food. And also some functions of our system should be synchronized for providing less problems during debugging and testing but especially while website is working for real customers. But still threads became with many problems.

3.7. Boundary Conditions

There are several exceptions in our system. The most important thing is the user has to have the internet connection to reach the site and make an order. If there is no connection, the user can't load the main page. At this condition, system displays a message shows an exception about connectionless. Also, when the system is closed by system admin any user that currently connection into the system will be disconnected and no user can connection the system until it is opened again.

Startup: go to system URL and login

Shut Down: click logout and close browser

Error Conditions:

- **Logging in:**
  - Username or password field can are blank.
  - Username is not a 5 digit decimal number.
  - Password is not 5 characters long.
  - Password and username don't match.
  - Username is wrong or does not exist.
  - The welcome screen does not appear after logging in.

- **User settings:**
  - User is unable to change certain settings or changes don't reflect.
  - Between the time of editing and updating, the system crashes.

4. Subsystem Services

In SANGFood, the our system has 12 subsystems like Customer Interface, Manager Interface, Admin Interface, Restaurant Owner Interface, are the subsystems for the ‘Presentation Layer’ which they are in Interface Subsystem. Subsystems for the ‘Business Layer’ are, Login Subsystem, System Management Subsystem, Restaurants Subsystem, Registration Subsystem, Update Info Subsystem, Ordering Subsystem, and Customer Info Subsystem. Finally, subsystem for the ‘Data Layer’ is Data Access.

Interface Subsystem

- **Customer Interface:** Provides services for users for common interfaces, it contains Login Form, Edit Profile Form, Password Change Form, and etc. That is to say, Customer Interface provides services to display common forms such as login, password change to all users.

- **Restaurant Owner Interface:** Provides services for restaurant owners for adding new food in their restaurants or any changing to their restaurant like editing address or changing menu.

- **Manager Interface:** Provides services for manager to Accept Restaurants form and send information to admins about new restaurant.

- **Admin Interface:** Provides services to display admins’ forms such as Create Username and Password for new restaurants, Add Restaurant Form, Form and so on.

SANGFood

Login Subsystem

Provides services for users (customer, restaurant owner,manager and admin) to login.

Update Info Subsystem(Edit Profile SubSystem)

Provides services for users (customer, restaurant owner) to update their information and to change password.

Customer Info Subsystem

Customers information.

Ordering Subsystem

Provides services for customers to make order.

Registration Subsystem

Provides services for user (customer) to register to the system.

System Management Subsystem

Provides services for admin to add restaurant, accept/reject restaurant, and delete restaurant.

Data Access Subsystem

Contains all our persistent objects, this part could be called Model of MVC.

Food Subsystem

Control foods and check them for stock or still available.

SANGFood

EditRest Page Subsystem

Provides resturant owners to edit their restaurant information.

5. References

1. Bruegge B. & Dutoit A.H.. (2010). *Object-Oriented Software Engineering Using UML, Patterns, and Java*, Prentice Hall, 3rd ed.
2. https://msdn.microsoft.com/en-us/library/ee658081.aspx

Gantt Chart: