**BAHIR DAR UNIVERSITY**

****

**BAHIR DAR INSTITUTE OF TECHNOLOGY**

**FACULTY OF COMPUTING**

**DEPARTIMENT OF SOFTWARE ENGINEERING**

**SOFTWARE TOOLS AND PRACTICE PROJECT**

**PROJECT TITLE: CAFÉ & RESTAURANT MANAGEMENT SYSTEM**

**GROUP MEMBER**

**NAME ID**

1. **ABRHAM ENDALAMAW 1101945**
2. **ABAYNEH BINALFEW 1101942**
3. **ADDISU ABEBAW 1103350**
4. **ADISIE FENTA 1102259**
5. **ADDISU ANILEY 1102093**
6. **AEMRO ENYEW 1103339**
7. **DANIEL BELETE 1101932**

Submitted to: Mintesinot A.

Submission date:09/04/21G.C

Contents

[ABSTRACT 3](#_Toc12672031)

[Background 3](#_Toc12672032)

[Introduction 3](#_Toc12672033)

[Objective 4](#_Toc12672034)

[Specific objective 4](#_Toc12672036)

[Scope of the project 5](#_Toc12672037)

[Functional Requirement 5](#_Toc12672038)

[Usecase Description 6](#_Toc12672039)

[System use case 8](#_Toc12672040)

[Activity Diagram 10](#_Toc12672041)

[State chart diagram 14](#_Toc12672042)

[Main Class 16](#_Toc12672043)

[Customer class 18](#_Toc12672044)

[Food Class 21](#_Toc12672045)

[Drink class 24](#_Toc12672046)

[Order class 26](#_Toc12672047)

[Staff Class 28](#_Toc12672048)

[Receptionist 32](#_Toc12672049)

[chef class 33](#_Toc12672050)

# **ABSTRACT**

The project “CAFÉ & RESTAURANT MANAGEMENT SYSTEM” is designed for Café & Restaurant management. The Café & Restaurant management system is an application, which involves administrational function related to various details like booking room, service list, stuff list, billing list etc. Realizing the need for quick retrieval and easy management of data, the system was studied, analyzed and computerized.

Owing to the number of drawbacks evident in the existing system, an automated solution is proposed. The Proposed System aims to remove most of the drawbacks found extensively in the existing system. The proposed system being developed as a replacement for the existing system is a console based and no interactions with the database. The proposed system has certain Modules. These are customer, Administrator, order, test, service and Room.

# **Background**

Café & Restaurant management practice is evolving to provide service through computerized system, with an increasing emphasis on technology focused service as compared to being primarily manual focused. Service is not based on a single transaction or episode, but rather includes multiple interactions with a customer for several times to make order and book room. Each time customer want to access Café & Restaurant service, he must be there to enjoy the service. But customer in different part of the city may not have enough time to go there and enjoy the service and many Café & Restaurants has been losing a lot of income because of following manual based Café & Restaurant management system

# **Introduction**

This document is mainly focused on Café & Restaurant management system for any company who want to make their Café & Restaurant computerized and It also describes the function and performance of online Café & Restaurant management system.

Our system is a System with ability to record, display, store, and exchange customer specific information in a manner that optimizes workflow within Café & Restaurant.

The information management systems used by Café & Restaurant manager’s professionals must support the delivery of service to the customer, including the dispensing of food and drink in accordance with Café & Restaurant standards. System ability to record, display, store, and exchange service related information.

## **Objective**

### General objective: This Café & Restaurant management system provide sufficient and relevant information to the user and provide effective working condition for the employees

# Specific objective

* Requirement gathering
* In this phase, we tried our best to collect requirement from different stakeholders such as Customer, Café & Restaurant manager, chef, receptionist and waiter by giving special attention to functional requirement.
* Requirement analysis
* At this phase our prime goal is checking both the relevance and traceability of requirement which help us to develop system that best satisfy our customer need and to provide priority to the requirement in development life cycle.
* Documentation
* After analyzing requirement, we formulate a clear Document to help us predict cost of development and to use as reference when we test and maintain our system.
* Description of use case
* Participated actor
* Postcondition
* Precondition
* Basic course of action
* Alternative course of action
* Test case for the system
* Design
* System Usecase
* Sequence diagram
* State chart diagram
* Class diagram
* Implementation
* Generating code from class diagram using case tool. for this project, we used Argo Ump to generate code from class diagram to speed up our development
* Cloning of the code from online repository. We have cloned some part of code from GitHub to make generated code perform effectively for what it intended for
* Testing
* We Provide unit test case for each method in the project and implement them by using Junit test to check effectiveness of our project

### **Scope of the project**

The scope of the project is restricted on the following elaborated service

* Display available service with their quality and price
* Reserve room to the customer
* Generate bill for food and room
* Add to and Delete order from order list
* Check room availability

# **Functional Requirement**

* The system display all services provided by the Café & Restaurant to the customer
* A customer can view available service such as, booking to the Café & Restaurant, paying bill, order food and order drink
* Manager can manage staff related information
* Receptionists can generate bill, check room availability and assign room to the customer
* Waiter can take customer order and prioritize the order
* A customer view detail information related to the service
* A customer fill bill form and provide payment by cash
* A chef take order form waiter and prepare food according to the received order
* The system provide staff related information to authorized user

### **Usecase Description**

Usecase name: view\_Services

Actor: Customer

Precondition: customer Login to the system

Basic course of action

|  |  |  |  |
| --- | --- | --- | --- |
| no | Action | no | Response |
| 1 | Login |  |  |
| 2 | search service |  |  |
| 3 | select service | 4 | display service list |
| 5 | select detail | 6 | display detail of the selected service |
| 7 | view detail |  |  |
| 8 | Exit |  |  |

Alternate course of action : NO

Postcondition : a customer can view the detail information related to the provided services

Usecase name: Order

Precondition: login to the system

Actor: customer , waiter, chef

**Basic course of action**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Action | no | System Response |
| 1 | customer select view service | 2 | display available service |
| 3 | customer view service |  |  |
| 4 | customer select service from the list | 5 | Display detail of selected service |
| 6 | customer select type of service | 7 | Display price |
| 8 | customer pay the price |  |  |
| 9 | customer submit price |  |  |
|  |  | 10 | Evaluate form |
|  |  | 11 | generate order to waiter |
| 12 | Waiter give order number to orders |  |  |
| 13 | waiter submit order to chef |  |  |
| 14 | chef receive order |  |  |
| 15 | chef prepare meal according to the order |  |  |
| 16 | chef transfer order to waiter |  |  |
| 17 | waiter deliver order to customer |  |  |
| 18 | customer enjoy the meal |  |  |
| 19 | customer generate feedback |  |  |
| 20 | customer submit feedback |  |  |
|  |  | 21 | Accept feed back |
| 22 | Exit |  |  |

***System Use Case***



### **Sequence Diagram**



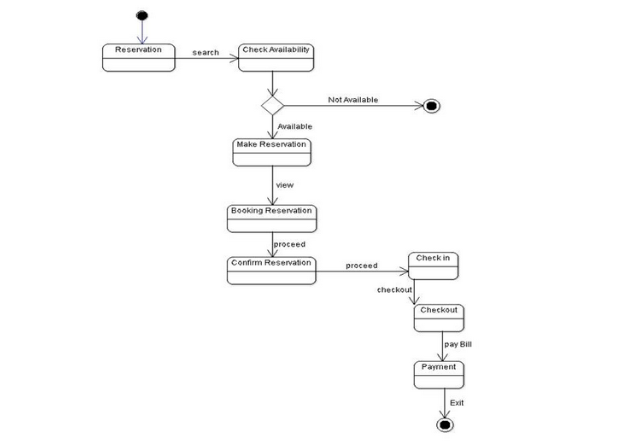




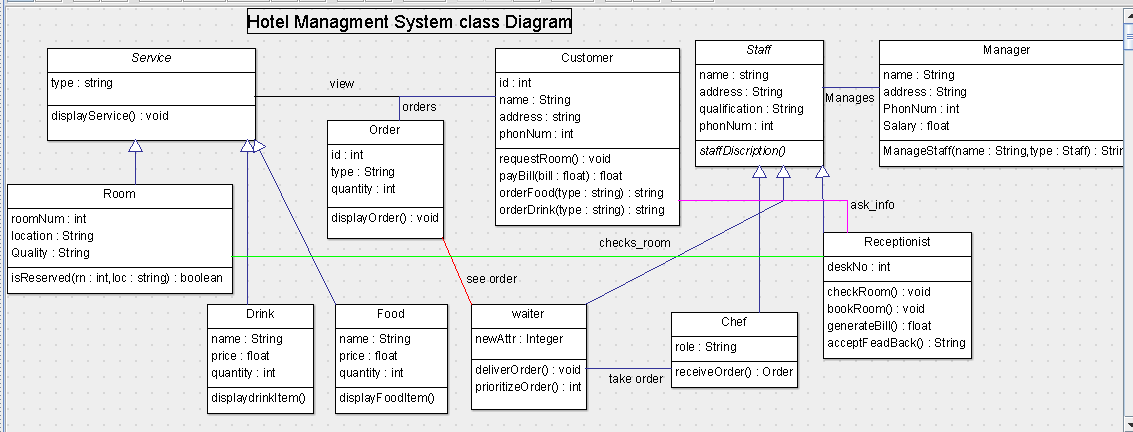


### **State chart diagram**





#### **Class diagram**



#### **Java implementation for the class design**

# Main Class

package Café & Restaurantmanagmentgroupproject;

import java.util.Scanner;

public class Café & Restaurantmanagmentgroupproject {

Drink drink;

Food food;

Room room;

public Café & RestaurantManagmentGroupProject() {

System.out.println("Well Come to Our Café & Restaurant!");

Wellcome();

}

public void Wellcome(){

String str1;

System.out.println("Who You Are!?");

System.out.print("I am:");

Scanner in=new Scanner(System.in);

str1=in.nextLine();

if(str1.equalsIgnoreCase("admin")){

Staff.checkPassword();

System.out.println("Enter Service Information");

int choice;

do{

System.out.println("Choose Service\n"+

"enter 1 to store foodinfo\n"+

"enter 2 to store drinkinfo\n"+

"enter 3 to store roominfo\n"

+ "enter 4 to view service\n"

+ "enter 5 to exit");

Scanner scan=new Scanner(System.in);

choice=in.nextInt();

switch(choice){

case 1:

Food.foodAdder();

break;

case 2:

Drink.DrinkAdder();

break;

case 3:

Room.roomAdder();

break;

case 4:CustomerInfo();

break;

case 5:System.exit(0);

break;

}

}while(choice!=4);

}

if(str1.equalsIgnoreCase("customer")){

CustomerInfo();

}

}

public void CustomerInfo(){

new Customer().Display();

}

public static void main(String[] args) {

Café & RestaurantManagmentGroupProject Café & Restaurant=new Café & RestaurantManagmentGroupProject();

}

}

# Customer class

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Café & Restaurantmanagmentgroupproject;

import java.util.Scanner;

import java.util.Vector;

public class Customer {

public int id;

public String name;

public String address;

public int phonNum;

public Customer() {

}

public Customer(int id, String name, String address, int phonNum) {

this.id = id;

this.name = name;

this.address = address;

this.phonNum = phonNum;

}

public void requestRoom() {

}

public float payBill(float bill) {

return 0.0f;

}

public String orderFood(String type) {

return null;

}

public String orderDrink(String type) {

return null;

}

public void Display(){

int num;

System.out.println("Enter 1 for Display Service\n"+

"Enter 2 for Order Service");

System.out.println("Enter Choice");

Scanner in=new Scanner(System.in);

num=in.nextInt();

if(num==1){

System.out.println("Service Information:\n");

System.out.println("Existing Service Items");

new Food().displayFoodItem();

/\*System.out.println( "Food:"+Food.name);

System.out.println( "Price:"+Food.price);

System.out.println( " Drink:"+Drink.name);

System.out.println( " price:"+Drink.price);

System.out.println( "RoomNumber:"+Room.roomNum);

System.out.println( "RoomQuality:"+Room.Quality);

System.out.println( "RoomLocation:"+Room.location);\*/

System.out.println("Newely entered Services\n");

Food.foodListDisplayer();

Drink.drinkListDisplayer();

Room.roomListDisplayer();

}

if(num==2){

new Order().OrderService();

}

}

}

# Food Class

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Café & Restaurantmanagmentgroupproject;

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

public class Food extends Service {

private static List<String>list=new ArrayList<>();

public static String name;

public static float price;

public static int quantity;

public Food() {

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public float getPrice() {

return price;

}

public void setPrice(float price) {

this.price = price;

}

public int getQuantity() {

return quantity;

}

public void setQuantity(int quantity) {

this.quantity = quantity;

}

public Food(String name, float price) {

this.name = name;

this.price = price;

}

public void displayFoodItem() {

System.out.println("1.shiro\n"

+ "price=10\n"

+ "2.pasta\n"

+ "price=15\n");

}

@Override

public String displayService() {

return getName()+""+getPrice();

}

public static void foodAdder(){

System.out.println("Do u Want to add food Item?");

Scanner scann=new Scanner(System.in);

int extent=scann.nextInt();

while(extent!=0){

System.out.println("Enter Food Name");

name=scann.next();

System.out.println("Enter Food Price");

price=scann.nextFloat();

list.add(name);

list.add(price+"");

System.out.println("Do u wanna to add some more?");

extent=scann.nextInt();

}

}

public static void foodListDisplayer(){

System.out.println("Available foods are:\n");

for(int i=0;i<list.size();i++){

System.out.println(list.get(i)+"\n");

}

}

}

# Drink class

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Café & Restaurantmanagmentgroupproject;

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

public class Drink extends Service {

public static String name;

public static float price;

public static String sampleDrink="Koka";

public static int kokaPrice=10;

private static List<String>list=new ArrayList<>();

public Drink() {

}

public Drink(String name, float price) {

this.name = name;

this.price = price;

}

public void setName(String name) {

this.name = name;

}

public String getName() {

return name;

}

public float getPrice() {

return price;

}

public void setPrice(float price) {

this.price = price;

}

public String displaydrinkItem() {

System.out.println("1.fanta\n"

+ "price=10\n"

+ "2.Koka\n"

+ "price=10\n");

return "name:"+name+"\n"+"price:"+price;

}

@Override

public String displayService() {

return toString();

}

public String toString(){

return getName()+getPrice();

}

public static void DrinkAdder(){

System.out.println("Do u Want to add Drink Item?");

Scanner scann=new Scanner(System.in);

int extent=scann.nextInt();

while(extent!=0){

System.out.println("Enter Drink Name");

name=scann.next();

System.out.println("Enter Drink Price");

price=scann.nextFloat();

list.add(name);

list.add(price+"");

System.out.println("Do u wanna to add some more?");

extent=scann.nextInt();

}

}

public static void drinkListDisplayer(){

System.out.println("Available Drinks are:\n");

for(int i=0;i<list.size();i++){

System.out.print(list.get(i)+"\n");

}

}

}

# Order class

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Café & Restaurantmanagmentgroupproject;

import java.util.Scanner;

import java.util.Vector;

import org.junit.Test;

import static org.junit.Assert.\*;

public class Order {

public static int id=2;

public static String type;

public static int quantity;

public int price;

public Vector make;

public Vector provide;

public Vector seeorder;

public Vector myCustomer;

public Order() {

}

public Order( String type, int quantity,int price) {

this.type = type;

this.quantity = quantity;

this.price=price;

}

public void OrderService(){

Scanner in=new Scanner(System.in);

System.out.println("Enter Your Order Info");

String orderName;

int amount,price=0;

System.out.println("Enter Order name");

orderName=in.next();

System.out.println("Enter Order quantity");

amount=in.nextInt();

System.out.println("Enter price");

price=in.nextInt();

if(orderName==Drink.sampleDrink&&price<Drink.kokaPrice)

testOrderDrink();

Order order=new Order(orderName,amount,price);

System.out.println("Order Success\n");

System.out.println(order.displayOrder());

}

public void testOrderDrink(){

assertEquals(price, Drink.kokaPrice);

}

public String displayOrder() {

return "OrderId="+id+"\nQuantity= "+quantity+"\nName="+type+""

+ "\nPrice="+price;

}

}

# Staff Class

package Café & Restaurantmanagmentgroupproject;

import java.util.Scanner;

import java.util.Vector;

import org.junit.Test;

import static org.junit.Assert.\*;

public abstract class Staff {

public String name;

public String address;

public String qualification;

public int phonNum;

static String testCase="admin123";

public Staff(String name, String address, String qualification, int phonNum) {

this.name = name;

this.address = address;

this.qualification = qualification;

this.phonNum = phonNum;

}

public abstract void staffDiscription();

public static void checkPassword(){

System.out.println("enter Password");

Scanner scan=new Scanner(System.in);

String password=scan.nextLine();

assertEquals(password, testCase);

}

}

#### Room class

package Café & Restaurantmanagmentgroupproject;

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

import java.util.Vector;

public class Room extends Service {

private static List<String>list=new ArrayList<>();

public static int roomNum;

public static String Quality;

public Vector my;

public Vector checks\_room;

public static String location;

public Room() {

}

public Room(int roomNum, String location, String Quality) {

this.roomNum = roomNum;

this.location = location;

this.Quality = Quality;

}

public int getRoomNum() {

return roomNum;

}

public void setRoomNum(int roomNum) {

this.roomNum = roomNum;

}

public String getLocation() {

return location;

}

public void setLocation(String location) {

this.location = location;

}

public String getQuality() {

return Quality;

}

public void setQuality(String Quality) {

this.Quality = Quality;

}

public boolean isReserved(int rn, String loc) {

return false;

}

public String displayService() {

return null;

}

public static void roomAdder(){

System.out.println("Do u Want to add room Item?");

Scanner scann=new Scanner(System.in);

int extent=scann.nextInt();

while(extent!=0){

System.out.println("Enter room Location");

location=scann.next();

System.out.println("Enter room number");

roomNum=scann.nextInt();

System.out.println("Enter room Location");

Quality=scann.next();

list.add(location);

list.add(roomNum+"");

list.add(Quality);

System.out.println("Do u wanna to add some more?");

extent=scann.nextInt();

}

}

public static void roomListDisplayer(){

System.out.println("Available room are:\n");

for(int i=0;i<list.size();i++){

System.out.println(list.get(i)+"\n");

}

}

}

# Receptionist

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Café & Restaurantmanagmentgroupproject;

import java.util.Vector;

public class Receptionist extends Staff {

public int deskNo;

public Receptionist(int deskNo, String name, String address, String qualification, int phonNum) {

super(name, address, qualification, phonNum);

this.deskNo = deskNo;

}

public void checkRoom() {

}

public void bookRoom() {

}

public float generateBill() {

return 0.0f;

}

public String acceptFeadBack() {

return null;

}

@Override

public void staffDiscription(){

System.out.println("DeskNum:"+deskNo);

}

}

# chef class

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Café & Restaurantmanagmentgroupproject;

public class Chef extends Staff {

public String role;

public Chef(String name, String address, String qualification, int phonNum) {

super(name, address, qualification, phonNum);

}

public Order receiveOrder() {

return null;

}

public void staffDiscription(){

}

}

#### waiter class

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Café & Restaurantmanagmentgroupproject;

import java.util.Vector;

public class waiter extends Staff {

public Integer newAttr;

public Vector takeorder;

public Vector seeorder;

public Vector returnorder;

public waiter(Integer newAttr, Vector takeorder, Vector seeorder, Vector returnorder,

String name, String address, String qualification, int phonNum) {

super(name, address, qualification, phonNum);

this.newAttr = newAttr;

this.takeorder = takeorder;

this.seeorder = seeorder;

this.returnorder = returnorder;

}

public void deliverOrder() {

}

public int prioritizeOrder() {

return 0;

}

@Override

public void staffDiscription(){

}

}

//service class

package Café & Restaurantmanagmentgroupproject;

import java.util.Vector;

public abstract class Service {

public abstract String displayService() ;

}

### **Test case**

public static void checkPassword(){

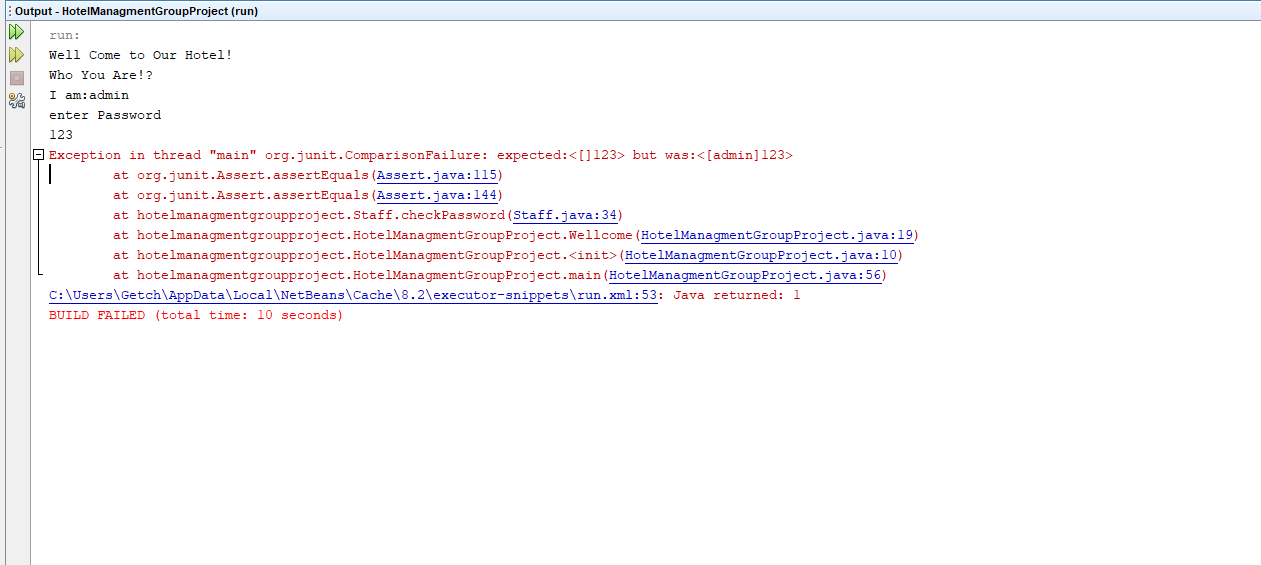
System.out.println("enter Password");

Scanner scan=new Scanner(System.in);

String password=scan.nextLine();

assertEquals(password, testCase);

}



#### Steps followed to add additional code from online repository

1. Go to web browser and search for GitHub
2. Log on to your online repository by providing your user name and password
3. Got to search button and type what you want to get and press search button
4. After choosing result that fit your requirement copy the link
5. Open your gitbush write the following command git clone paste the link and press enter

# **Tools used in system development**

* **Microsoft visio 2016 for draw, sequence diagram and use case diagram.**
* **Edraw max for draw state chart diagram**
* **Argo UML for class diagram and generate source code from UML class diagram**
* **NetBeans 8.2 for testing and debugging code**
* **Microsoft Word 2016 for documentation**
* **Git bush to clone code from online repository**

REFERENCE

http://[www.tutorialspoint.com](http://www.tutorialspoint.com)

http://en.[www.wikipedia.org](http://www.wikipedia.org)

<http://ant.apache.org>

https://github.com