

# **DESIGN & SPRINT 2**

## **INSTRUCTIONS**

# Design Document Template

1. **Cover sheet/Title page**
2. **Conceptual System Design** <pg num>
  - a) Report formats <pg num>
  - b) Screen layouts/screen shots <pg num>
3. **Technical Design, high level** <pg num>
  - a) Detailed class diagrams <pg num>
  - b) Supporting text specification <pg num>
  - c) Database table descriptions <pg num>
  - d) Technical support specification <pg num>

# Design Document Specifications

## 2. Conceptual System Design

### a) Report formats

Insert samples of all reports generated by or available through your system e.g., register receipt, order slip, daily business report/summary. The reports should include appropriate sample data, not xxx as place holders.

Examples of these screen could be but not limited to are an order slip, payment page, daily sales page, inventory, average turnaround etc. Also encouraged to add the Food queue screen that would be helpful for a manager or cook to view the time ordered, time elapsed and the status of the order (like pending, completed etc.)

### b) Screen layouts

Insert current depictions of the screens that the system will produce for the client. **This section of the document should be guiding the user through the application**, with the requirements and use cases being addressed by each screen clearly identified on the bottom of the page – underneath the screen.

This should include a set of complete screens starting from login page, clock in/out page, things the end user can do etc. until successful log out.

This should be shown for each individual role in the restaurant.

# Design Document Specifications

## 3. Technical Design

### a) Detailed Class diagrams

Insert an expanded/updated version of the class diagrams from the requirements phase. The UML class diagrams **should now include the variables, methods and properties along with appropriate visibility modifiers (public/private/protected)**.

This should be an extension of your Class Diagram.

#### Enemy

- name : string

- strength : int

- power : float

+ Enemy()

+ Attack(List<Transforms>) : void

+ Move() : void

+ Range(Vector3) : float

# Design Document Specifications

## 3. Technical Design

### b) Supporting Text specification

Create the documentation for the classes.

Mention what each method is intended to do.

The following is an extract from the java string class.

**public final class String extends Object.**

*You may choose one of the following methods to complete this task.*

#### Login Page Class:

**LoginPage()** - Initializes the attributes to default values.

**login()** - Logs user in under account found in database.

**signUp()** - Adds new user's information into the database.

**employeeLogin()** - Logs employee into their account if employeeStatus is true.

**managerLogin()** - Logs manager into their account if managerStatus is true.

| Constructor                        | Description   |  |
|------------------------------------|---|--|
| String()                           | Initializes a newly created String object so that it represents an empty character sequence |  |
| Field summary                      |   |  |
| Modifier and Type                  | Field   | Description  |
| static final<br>Comparator<String> | CASE_INSENSITIVE_ORDER  | A Comparator that orders String objects as by compareToIgnoreCase. |
| Method summary                     |   |  |
| Modifier and Type                  | Method  | Description  |
| char                               | charAt(int index)   | Returns the char value at the specified index.                     |



*Use either of these options*



# Design Document Specifications

## 3. Technical Design

### c) Database table descriptions

Use your ER diagram to create the tables for the database. The tables should include primary key, foreign key(s), variable name, data type and a description. This will include, for each table, **descriptions of what the table means and how it's functionally used (in the UI, etc.)**, descriptions of what each attribute means (if it isn't obvious), explanations of the relationships (foreign keys) from this table to others. Look for the example below:

| Customer Table | Attributes     | Data type | Attribute type | Dependency (PK/FK) |
|----------------|----------------|-----------|----------------|--------------------|
|                | Customer Phone | Int       | Simple         | Primary key        |
|                | Customer Name  | String    | Compound       | -                  |
|                | ...            |           |                |                    |

#### Description for Customer Table

*The Customer table stores Customer info like phone number, name etc. Once a Customer create an account, his/her data is automatically stored here. This table is related to the orders and payment table with its Primary key.*

# Design Document Specifications

## 3. Technical Design

### d) Technical Support specification

This should address the following

- the training approach
- training materials
- installation support
- onsite or remote assistance
- updates, troubleshooting, help desk hours, etc.

Explain how you plan to give each to your customer. Your approach on each...

# Sprint 2

## What you need to do?

- In this sprint you will be creating and delivering the restaurant menu component of the system. Your system should contain the complete restaurant menu for order-taking purposes.
- The user (Customer) needs to be able to sign up, log in and then have GUI access to the various menu items and be able to quickly enter the desired orders to the cart. This menu needs to have various sizes of pizzas, the typical types of toppings, various crust options, and beverages. The customer should be able to order things like a medium, thin crust pizza, with pepperoni and extra cheese etc.
- This component/prototype can be developed for running on a stand-alone desktop/laptop computer or with web-based elements. The customer records are to be stored in a file structure that will simulate the data base.

# Sprint 2

## Required files:

Use either java GUI or C# GUI. JavaFx, Swing, JFrame, windows forms, Unity are acceptable.

The source code should be documented with either Javadoc style comments for java code or triple slash (///) style comments for C# code

The deliverable should include:

1. a zipped folder (with all components/files related to this sprint)
2. an executable file (.exe or .jar)
3. a video of the running project, showing all aspects of the program **(vital)**
4. a read me document with any special remarks and username/password combinations that already exist.

# Sprint 2

## Requirements:

- Completing the **Customer component** should be adequate.
- The **menu and cart display should display the pricing** of the various options.
- There should be a **display screen for the completed order** showing the items ordered, the prices and totals.

The detailed menu options is available in the next slide/page. The pricing and additional/extra choices can be decided by the team.

The contact details of the restaurant should be available for the Customers as well. It is included in slide/page 12.

# **MENU ITEMS**

**Crusts:** Thin, Regular, Pan

**Sizes:** Small, Medium, Large, Extra large

**Sauce:** 2 options (Marinara and Alfredo)

**Toppings:** 5 Veg and 5 Meat options (Cheese, Onion, Pineapple, Jalapeno, Olives, Mushroom, Pepperoni, Chicken, Italian sausage, Bacon, Beef)

**Beverages:** 3 options (Coke, Sprite, Fanta)

**Dessert:** 3 options (Chocolate chunk cookie, Cinnamon roll, Molten lava cake)

# RESTAURANT DETAILS

## **Name of the pizza restaurant:**

Mom's and Pop's Pizzeria

## **Contact Info:**

680 Arntson Rd, suite 161

Marietta, ga 30060

770 555-1212

MomAndPopPizzeria.com

## **Operating hours:**

Mon-Thur. 9am-11pm

Fri-Sat. 11am-12am