



American International University-Bangladesh

Department of Computer Science

Project Report

Course: Object Oriented Programming – 2 (C#)

Section: A

Project Name: Dine POS System

Submitted By:

Group: 4

Name	ID
Biswas, Tushar	18-36802-1
Zaman, Md. Muntanuz (Prev. Group 10)	18-37272-1

Objective and Scope:

“Dine POS System” is a desktop application which is developed with a view to place and process the orders of the customers in a restaurant. Through the application, the restaurant management can place the orders of the customers through Dine-In or Takeaway. With the help of this application, the restaurant management can also keep track of how many units of items should be served and how much should be expected of the customers. It will also allow the restaurant management to generate a bill which will be automatically calculated by the application.

Features:

- ✚ Defining roles and users (Example: Manager, Head Chef).
- ✚ Create food categories and food menu.
- ✚ Place an order.
- ✚ Modification of order.
- ✚ Generating Bills.

Technical Approach:

1. **Microsoft .Net Forms:** We used various .Net forms to help us design and develop the application. We added buttons, panels, textboxes, grid views, tables, labels and print preview.
2. **Inheritance:** We used inheritance for reusability purposes and allowed the application to have a cleaner code. All the forms derived from parent class and inherits its property and even handling methods.
3. **SQL Database for data storage:** We used SQL database to add, select, and delete data given by the customers, management, and admins. We also used the data added by the customers to calculate the total price of the products using data grid view. The unit prices of the products are added and modified by the restaurant management.
4. **Class:** We used various classes to make the code clean and well organized. It also helped us to quickly find errors and fix them.
5. **Object:** We used various objects throughout the project.
6. **Parameterized and Non-parameterized Function:** We used Parameterized Functions to
7. **Polymorphism:** we used Polymorphism to reuse the functions of a class repeatedly throughout the application code.
8. **Encapsulation:** We used Encapsulation to assign the public, private, protected, and default statuses of the class, variables, objects of the application.