

A PROJECT REPORT
for
Restaurant Management System

Submitted by:

Nazmus Sakib Siam (Intake: 42, ID: 18193103049)

Md. Al-amin Hossain Bijoy (Intake: 42, ID: 18193103054)

Pallab Sarkar (Intake: 42, ID: 18193103052)

In partial fulfillment of the requirements for the degree of

BACHELOR OF SCIENCE
IN
COMPUTER SCIENCE AND ENGINEERING
at



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
BANGLADESH UNIVERSITY OF BUSINESS AND TECHNOLOGY
(BUBT)

DHAKA- 1216

JUNE, 2020

Supervised by

M. M. Fazle Rabbi

Assistant Professor

Department of Computer Science and Engineering (CSE)

Bangladesh University of Business and Technology (BUBT)

Mirpur-2, Dhaka-1216, Bangladesh

ABSTRACT

This report documents the process of designing, developing and testing a software system to be used in a restaurant; usually given the name restaurant management system. The restaurant management system is there to help communication between all teams within a restaurant by minimizing the probability of human errors. The concept is also based on online food ordering and home delivery system like foodpanda, Uber Eats, pathao etc. It reduces the time management of the people because when customers want to complete their breakfast, lunch, dinner in house, they can easily order food from this software. They don't need to go to the restaurant if they want. There is also a good facility in this software to cancel the ordering food. It is easy for people to use this software for ordering foods. There will be different items available in the restaurant; many customers will be coming at different time for having food, they will be eating different items available in the restaurant. In this system, the customers can access the menu page with the listed items and prices according to the desired time. In some hotels and restaurants there are many steps to order food like enquiry the customer's name, customer's id, customer's address etc. In this software it will generate bill for customer after proper enquiry. This software is able to store the records such as, ordered foods, bills, order date, order time etc. of the customers. It will generate report for customer as well as for employees. Restaurant owner has to maintain records of each sales and purchase to keep track of the availability of stock of any materials which is used for cooking. The main motive of this software is to satisfy our customer needs and save their valuable times. The system will save time and will be easy to use where compared to manual work which will be done in prepare. Growing technology changes the needs for the people, to satisfy that we can develop a software called "Restaurant Management System".

DECLARATION

I hereby declare that the project entitled “Restaurant Management System” submitted for the degree of Bachelor of Science Engineering in Computer Science and Engineering in the faculty of Computer Science and Engineering of Bangladesh University of Business and Technology (BUBT), is our original work and that it contains no material which has been accepted for the award to the candidates of any other degree or diploma, except where due reference is made in the next of the project to the best of our knowledge, it contains no materials previously published or written by any other person except where due reference is made in this project.

Nazmus Sakib Siam
ID: 18193103049
Intake: 42
Section: 02

Md. Al-amin Hossain Bijoy
ID: 18193103054
Intake: 42
Section: 02

Pallab Sarkar
ID: 18193103052
Intake: 42
Section: 02

CERTIFICATION

This project “Restaurant Management System” report submitted by Nazmus Sakib Siam, Md. Al-amin Hossain Bijoy and Pallab Sarkar students of Department of Computer Science and Engineering, Bangladesh University of Business and Technology(BUBT), under the supervision of M. M. Fazle Rabbi, Assistant Professor, Department of Computer Science and Engineering has been accepted as satisfactory for the partial requirements for the degree of Bachelor of Science Engineering in Computer Science and Engineering.

(M. M. Fazle Rabbi)
Assistant Professor & Project Supervisor
Department of CSE

(Dr. Kamruddin Md. Nur)
Associate Professor & Chairman (Acting)
Department of CSE

DEDICATION

Dedicated to our parents for all their love and inspiration.

ACKNOWLEDGEMENTS

Any attempt at any level cannot be satisfactorily completed without the support and guidance of learned people. We would like to express our immense gratitude to all BUBT teachers on computer programming for their constant support and motivation that has encouraged us to come up with this project.

We are profoundly grateful to **M. M. Fazle Rabbi** his expert guidance and continuous encouragement throughout to see that this project rights its target since its commencement to its completion.

We would like to express deepest appreciation towards **Proctor** of Bangladesh University of Business & Technology **Wing Commander Md Momenul Islam Retd.** And **Dr. Kamruddin Md. Nur**, Chairman (Acting) of Computer Science & Engineering whose invaluable guidance supported us in completing this project.

APPROVAL

This Thesis “**Restaurant Management System**” Submitted by **Nazmus Sakib Siam, Md. Al-amin Hossain Bijoy, Pallab Sarkar** ID NO: **18193103049, 18193103054** and **18193103052** Department of Computer Science and Engineering (CSE), Bangladesh University of Business and Technology (BUBT) under the supervision of M. M. Fazle Rabbi, Assistant Professor, Department of Computer Science and Engineering has been accepted as satisfactory for the partial fulfillment of the requirement for the degree of Bachelor of Science (B.Sc. Eng.) in Computer Science and Engineering and approved as to its style and contents.

Supervisor:

M. M. Fazle Rabbi

Assistant Professor

Department of Computer Science and Engineering (CSE)

Bangladesh University of Business and Technology (BUBT)

Mirpur-2, Dhaka-1216, Bangladesh

Chairman:

Dr. Kamruddin Md. Nur

Associate Professor & Chairman (Acting)

Department of Computer Science and Engineering (CSE)

© Copyright by Nazmus Sakib Siam (18193103049), Md. Al-amin Hossain Bijoy (18193103054), and Pallab Sarkar (18193103052).

All Right Reserved

ABBREVIATIONS AND NOMENCLATURES

C	C programming Language
API	Application Programming Interface
CS	Client/Server
DBMS	Database Management System (Notepad)

TABLE OF CONTENTS

Abstract	iii.
Declaration	iv.
Certification	v.
Dedication	v.
Acknowledgement	vi.
Approval	vii.
Copyright	viii.
Abbreviations	viii.
Chapter 1: Introduction	1-7
1.1 Introduction	1
1.2 Existing model	2
1.3 Problem Statement	2
1.4 Motivation	3
1.5 Objective of the project	4
1.6 Contribution	4
1.7 Organization of Report	5
1.8 Conclusion	6
Chapter 2: Existing System	8-21
2.1 Introduction	8
2.2 Existing System	8
2.2.1 Foodpanda	9
2.2.2 HungryNaki	9
2.3 Supporting Literature	10
2.3.1 Language: C.....	10
2.3.2 Code::Blocks.....	11
2.3.3 Borland Turbo C	13
2.3.4 Microsoft Visual Studio	15
2.4 Used Diagram	20
2.5 Analysis of Existing System	21
2.6 Conclusion	21
Chapter 3: Proposed Model	23-38
3.1 Introduction	23
3.2 Feasibility Study	23

3.2.1 Technical Feasibility	24
3.2.2 Operational Feasibility	25
3.2.3 Economical feasibility	26
3.3 Requirement Analysis	26
3.3.1 Non-Functional requirement	27
3.3.2 Functional requirement	29
3.4 System Design	30
3.4.1 Diagram	31
3.4.2 Function	35
3.5 Implementation	36
3.5.1 Front End	36
3.5.2 Back End	37
3.6 Conclusion	38
Chapter 4: Experimental Result	39-44
4.1 Introduction	39
4.2 Result Analysis	40
4.2.1 Sales Report	40
4.2.2 Food List	41
4.2.3 Feedback	42
4.3 Application	43
4.4 Conclusion	44
Chapter 5: User Manual	45-51
5.1 Introduction	45
5.2 System Requirements	45
5.2.1 Hardware Requirements	45
5.2.2 Software Requirements	46
5.3 User Interface	47
5.3.1 Home Page	47
5.3.2 Login	47
5.3.3 Order/Menu	49
5.3.4 Confirm Order Option	49
5.3.5 Payment Option	50
5.3.6 Feedback	51
5.4 Conclusion	51
Chapter 6: Conclusion and Further Work	52-53
6.1 Conclusion	52
6.2 Limitations	53

6.3 Future Works 53

References 54

Appendix 55

Chapter- 1: Introduction

1.1 Introduction

“Restaurant Management System” is a new generation of restaurant management software. It is a complete solution, beginning with taking orders from customers, ending with billing and tax reports. The user interface is carefully optimized for high speed input of a customer order and the prevention of common mistakes. Payments can be accepted by cash, credit cards, checks, Bkash and Rocket. For home delivery, the payment method can be only accepted by cash (cash on delivery), Bkash and Rocket. For managers, there is a rich set of reports that shows a complete picture of restaurant operations and life cycles, consumption, most active employees, payment methods, stock and automatic tax calculations. By standardizing the entire restaurant management process, the software radically improves serving speed. It’s easy to install and easy to use [1].

This system has been proposed to be implemented to replace the manual system. The main aim of this project is computerization of all processes which happens in the restaurant. It is a database system for creating a selective retrieved of information, for subsequent analysis, manipulation and application [2].

The traditional system is a restaurant paper menu and ordering system is replaced with an electronic medium i.e. a digital tablet. Due to a digitalized system, the risk of manual errors is eliminated, thus eliminating the communication barrier. The tablet displays all the information the customer needs to know about the order he/she has placed.

This self-service fast food restaurant will be equipped with a user-friendly touch screen, a credit/debit card reader, and software for completing the process at the backend. For this system there will be a system administrator who will have the rights to enter the menu with their current prevailing prices. He/she can enter anytime in the system by a secured system password to change the menu contents by adding or deleting an item or changing its price.

Now when the customer enters the restaurant, he will place his order with the help of the touch screen using the intuitive graphical user interface, right from the selection of language till the

payment confirmation. He will select from the food options according to his choice and the system will display the payment amount he has to make once he has finished with his order [3].

For delivering food, the delivery team will always be ready to deliver customer's order and when an order will be placed, after cooking the food the delivery boy will go to the customer's destination with the ordered food including the bill. This is very easy and safe way to deliver food.

1.2 Existing Model

In the existing model customers have to come down to the restaurant in person to order foods listed by the restaurant. He/she can see all the foods in the menu option. They can choose foods from the menu which are available in the restaurant. After choosing the foods, the customers have to pay the bill using any method such as, cash, card, checks, Bkash and Rocket. After proper enquiry the foods will be served including the bill by the waiter.

For home delivery system, at first the customers have to select foods, then complete the payment method (cash on delivery, Bkash, Rocket or online) and confirm the order using the software. Then the customer can be able to see the activity of his/her order and track can the delivery boy. A customer can cancel order before preparing the food.

1.3 Problem Statement

Nowadays, many restaurants manage their business by manual system especially taking customers' orders. Restaurant's waiters take the customers' orders by manual system with using paper. This is a problem for restaurant's waiters and there is the probability of losing and duplicating customers' information. Additionally, it would affect to reputation restaurant in operate management of ordering.

Besides, in a restaurant waiters' information is also stored by manual system kept using paper and this is difficult for restaurant administrator to find waiters' information and there is the probability of missing the paper and is difficult to arrange the schedule. Sometimes, waiters' information and customers' information are important to restaurant administrator for reference in the future.

Furthermore, the restaurant administration needs proper management in the menu section. This is very important for a restaurant. The list of foods in a menu should be well designed and displayed. Besides this section is for customers. So, the menu that the restaurant is providing should be well prepared and satisfactory.

As a result, the current system (manual system) is not effective and efficient to use anymore because the current system cannot save, manage and monitor a restaurant's waiters' information, menu information, customer ordering information and generate reports well.

1.4 Motivation

The motivation for designing this software came because my friends are involved in the fast food business. They were continuing their business according to the manual system and I personally do not like waiting for long in the restaurant or to have to call to place an order especially during the peak lunch or dinner hours. My friends also wanted to start home delivery system. So, I discussed this matter with them and they were also agreed with me about new restaurant management system. As a student of CSE department I have already learnt about the C and C++ programming languages and I know how powerful and dynamic these languages are when it comes to web designing and applications. So, I use C programming language for this software with including many data type, structure and file. This language is also useful for future software updates.

The aim of our project is to provide digital facilities in restaurant, save customers' time and online food ordering system. There are some other reasons we motivated to do this project.

- Customers can get best facilities in a restaurant.
- Customers can enjoy their food in a better environment.
- Customers can purchase food by sitting in home.
- Customers can get their ordered food at home by cash on delivery system.
- Very easy to order.
- This system is very profitable.
- Customers can order their foods in 24 hours.
- Customers can purchase food by using credit card, mobile banking and cash on delivery system.

- Transaction system is very safe.
- Customers can save their many times by using this system.

1.5 Objectives of the Project

The main objectives of this software are the design, documentation and implementation of a Restaurant Management System. While the subsidiary objectives are:

- To determine how this restaurant management system has facilitated increase productivity, decrease paperwork, and ability to analyze trouble spots.
- To determine how the system will increase the level of services quality and customer satisfaction
- To determine how the system will help the restaurants to have the ability to build competitive and strategic advantages by better understanding the needs and wants of the guests, hence building repeat business.
- To determine how the system can lead organization towards better decision making and building a competitive advantage over its competitors.
- To determine how restaurant management system will improve the operating efficiencies, provide restaurant and support center management with timely access to financial and operating data and reduce administrative time and expense.

1.6 Contribution

We have implemented our project in various kinds of function. In our system customers find the desire food from the menu. The one of the most important features of our system is online food ordering system. A customer can choose food by using this software just clicking on the “**Menu**” option. And another most important implemented part are “**User**” and “**Login**” system for administration and chef. By this module they can communicate themselves about the activities. Our system also contains the menu option so that customers can choose their food frequently. A customer has to add his name for ordering food and it is compulsory.

In our system there has very effective admin panel so that the admins can manage the whole system very easily. In our admin panel there are many functions are contained. One of the most important function is receiving feedback from customers so that we can improve our facilities. An admin can see the quantity of food or how much foods are available in our stock. Moreover, admin can see the availability status of empty, low and available materials. Another most important feature we have implemented in our system storing all the details of customers bill, order foods, payment method and feedback.

Here is our contribution of this project:

- Customer will get best deals.
- Customers can easily order their foods including best payment methods.
- This project helps for purchasing foods online.
- Customers can choose their foods by categories/menu.
- Customers can choose their products along with their budget.
- This project helps the customers to purchase food in very easy way.
- Customers can get their food by home delivery.
- Customers can pay for the foods by cash on delivery or other online method.
- Customers can contact with us for any kinds of inquiry.
- Customers have the checkout option for order foods.
- Customers can cancel order at any time
- Customers can give rating and throw their feedback of food items.

1.7 Organization of Report

Chapter 2 explains existing system, existing or supporting literature and analysis of existing system. In existing system, we will discuss about the history of restaurant management and how restaurant management has started in Bangladesh. Moreover, in this part we will also discuss about the many types of existing restaurants and functionalities of the existing restaurants. In supporting literature, we will describe about the all types of tools that we have used in our system. From analysis of existing system, we will come to know pros and cons of restaurant management system.

Chapter 3 consists of the feasibility study, requirement analysis, system design and implementation. In feasibility study part we will discuss technical feasibility, economic feasibility, operational feasibility, behavioral feasibility and legal feasibility. In this chapter next part is requirement analysis. Here, we will discuss about the functional and non-functional requirements. Then next part is system design. In this part we will describe how we have designed our system. In implementations part of this chapter we will discuss about front end, forms design, report design, back end, database design.

Chapter 4 explains about the experimental results of our project and analysis of the result and also discuss about the applications of our project. In result analysis part we will discuss about the report or output of our system and we will know about the daily, weekly and yearly purchases of our system. In application part we will describe about the real time uses and financial benefits of our system.

Chapter 5 discusses about the system requirements, user interfaces or the screenshots of our system. In system requirements part we will discuss about the hardware requirements and software requirements for client and server. In user interfaces part we will give all the screenshots of our system.

Chapter 6 concludes the report of our system. In this chapter we will discuss about limitation and future works. In limitation part we will discuss about the limitation of our system. In future works we will discuss about the modules which we will develop in future.

1.8 Conclusion

The Internet has become a major resource in modern business; thus, restaurant management has gained significance not only from the entrepreneur's but also from the customer's point of view. For the entrepreneur, online based food ordering generates new business opportunities and for the customer. We have implemented a restaurant management system **"3 Food"**. In this system the foods are made of customers' choice and needs. Because nowadays restaurants' foods are very popular to the people. We have implemented very important module such as admin panel, login and user system, feedback and rating method, food availability etc. From the administrative view we have implemented the best enquiry of customers, storing all the purchases and bill report of a

customer etc. We have chosen this system because now a days it is very popular all over the world as well as in our country. This system is very useful for a user or customer. Customer can select, add and confirm food for order. A customer loves this system because he does not need to go outside for having breakfast, lunch or dinner if he/she want. He can get the food/meal by sitting at home. So, he/she saves many times by online food ordering system. In future we will work more in our system. We will do our system more user friendly so that customer can easily order the food from our restaurant. Though in our system now only cash on delivery and Bkash system is available but in future we will implement the rocket and online payment system for online ordering. We will also implement the comparison between two or more food according to the nutrition value.

Chapter- 2: Existing System

2.1 Introduction

The current literature on consumer online purchasing decisions has mainly concentrated on identifying the factors which affect the willingness of consumers to engage in online food order system. In the domain of consumer behavior research, there are general models of buying behavior that depict the process which consumers use in making a purchase decision. These models are very important to restaurant owners as they have the ability to explain and predict consumer's purchase behavior.

2.2 Existing System

We have tried to finding a unique project idea. **The Restaurant Management System- “3 Food”** is an existing idea, but we tried to represents it with new look and new architecture. We also tried to find importance and advantages of this project. In the existing system all transactions, purchasing of foods were done manually which was time consuming. Records were prepared manually as and when needed. Maintaining of records is very tedious task. There is no computer system for handling payments. All calculations are performed manually which may not be accurate always. But any user can use this existing software to search for any kind of food. Once they make of their mind to purchase any food, they can place an order and make a payment throw various available payment option. In this system all the transactions, food order records, payment records are computer based.

Software based restaurant management is very important for a country like Bangladesh. Restaurant owners should abandon the manual system and operate restaurants on a digital system. There are many software and websites in the world through which you can order food online and at the same time there are some such popular software and websites in Bangladesh. Such as **Foodpanda, HungryNaki, Pathao Food, Sheba Food** etc. Some of the software are discussing below.

2.2.1 Foodpanda

Foodpanda (stylized as foodpanda) is an online food delivery marketplace owned by Berlin-based company Delivery Hero SE and operates in about 50 countries. It is mostly active in Asia Pacific, Bulgaria and Romania. It is headquartered in Berlin, Germany. The service allows users to select from local restaurants and place orders via its mobile applications as well as its websites. The company has partnered with over 27,095 restaurants in 193 cities and works with over 15,733 delivery riders. The firm was acquired by Delivery Hero in early December 2016.

Foodpanda was launched on 13th November 2013, in Bangladesh and currently operating in three cities: Dhaka, Chittagong and Sylhet, and sporting the bright pink colors, foodpanda delivers food super-fast from over 2000 restaurants without any delivery charge [4].

2.2.2 HungryNaki

HungryNaki (Bengali: হাংরিনাকি); is the first Bangladeshi online food ordering and delivery service, launched on October 1, 2013, based in Dhaka, Bangladesh. "Naki" is a Bengali word, sums up the meaning of the word HungryNaki to - "Are you Hungry?"

The primary goal of HungryNaki is to provide customers with food from their favorite restaurants without having to fight through Dhaka's (one of the most densely populated cities in the world) chronic traffic congestion, waiting in the long queue or bad weather. The service let the users choose their preferred foods from nearby restaurants and place orders through its websites and mobile applications (Android and iOS), and delivers the order at the doorstep of the receivers, with the minimum delivery fee of 45 BDT. They send out SMS and make calls to confirm orders and/or food availability. They accept all the regular payment methods along with 'cash on delivery'. HungryNaki provides a review section on restaurant pages on their website and mobile apps, where their customers can give feedback or submit a complaint on the delivery process and the taste of the foods s/he received [5].

2.3 Supporting Literature

An existing restaurant management system consists of a language, web tools some web tools, IDE, algorithms and models. We have used C programming language, Code::Blocks, Notepad, diagram etc. Some of these are given below.

2.3.1 Language: C

The C Language is developed by Dennis Ritchie for creating system applications that directly interact with the hardware devices such as drivers, kernels, etc. C programming is considered as the base for other programming languages, that is why it is known as mother language.

It can be defined by the following ways:

- Mother language
- System programming language
- Procedure-oriented programming language
- Structured programming language
- Mid-level programming language

File: main.c

```
#include <stdio.h>
int main() {
printf("Hello C Programming\n");
return 0;
}
```

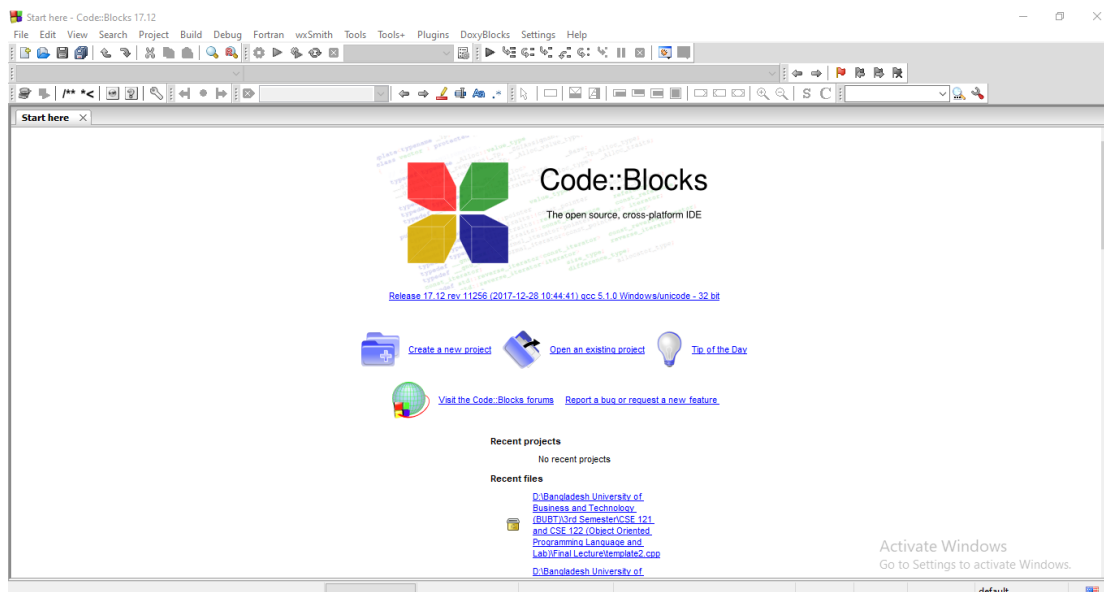
History of C language is interesting to know. Here we are going to discuss a brief history of the c language. C programming language was developed in 1972 by Dennis Ritchie at bell laboratories

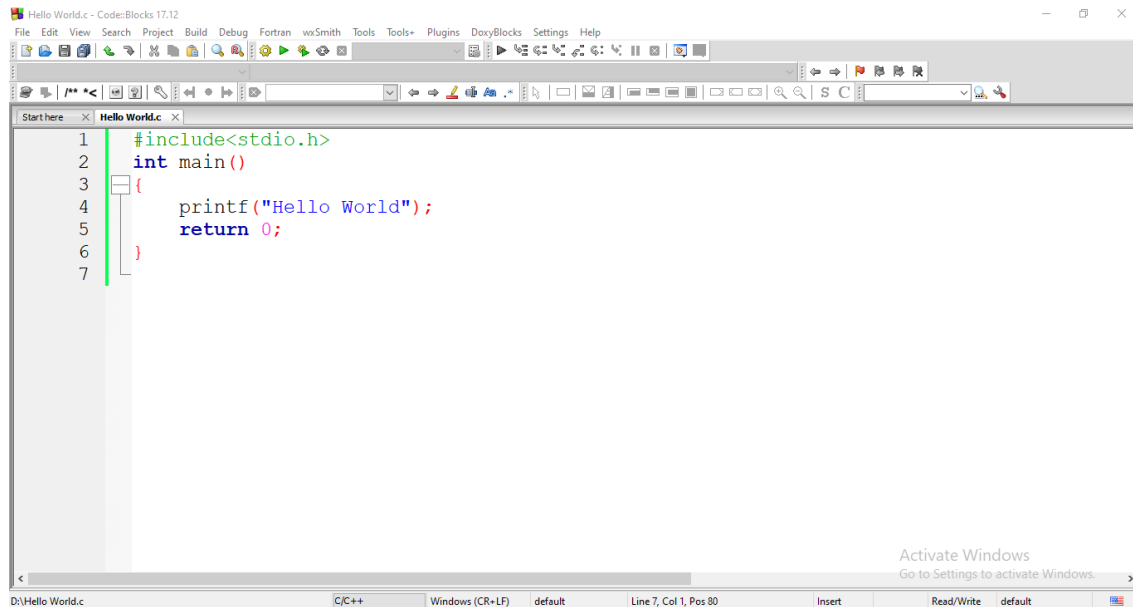
of AT&T (American Telephone & Telegraph), located in the U.S.A. Dennis Ritchie is known as the founder of the C language. It was developed to overcome the problems of previous languages such as B, BCPL, etc. Initially, C language was developed to be used in UNIX operating system. It inherits many features of previous languages such as B and BCPL [6].

2.3.2 Code::Blocks

Code::Blocks is a free, open-source cross-platform IDE that supports multiple compilers including GCC, Clang and Visual C++. It is developed in C++ using wxWidgets as the GUI toolkit. Using a plugin architecture, its capabilities and features are defined by the provided plugins. Currently, Code::Blocks is oriented towards C, C++, and Fortran. It has a custom build system and optional Make support.

Code::Blocks is being developed for Windows and Linux (the latest macOS version is 13.12 released on 2013/12/26) and has been ported to FreeBSD, OpenBSD and Solaris [7].

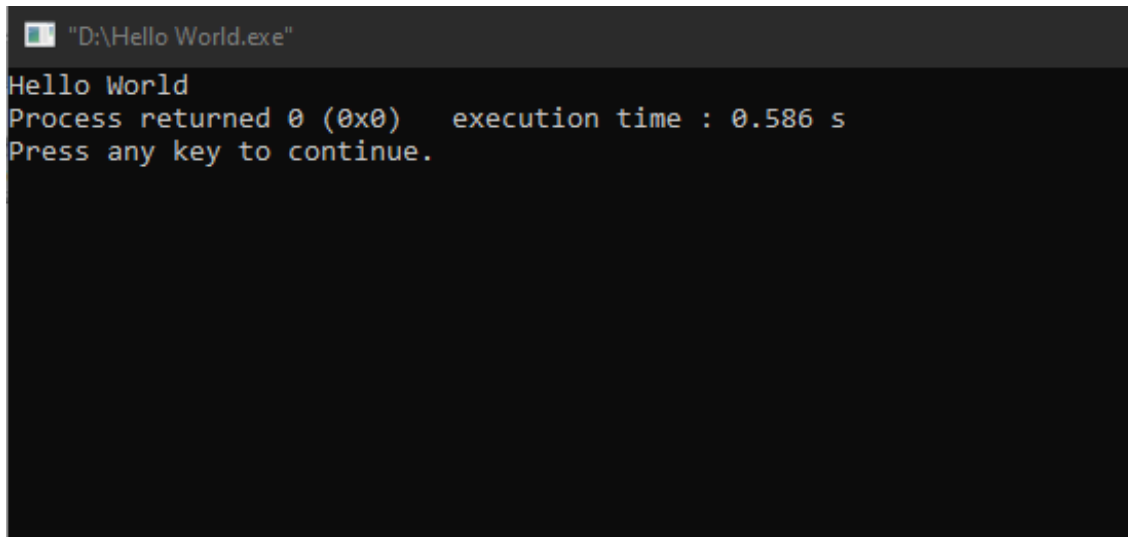




The screenshot shows the Code::Blocks IDE interface. The menu bar includes File, Edit, View, Search, Project, Build, Debug, Fortran, wxSmith, Tools, Tools+, Plugins, DoxyBlocks, Settings, and Help. The toolbar contains icons for file operations, editing, and building. The main editor window displays a C++ program in a file named 'Hello World.c'. The code is as follows:

```
1 #include<stdio.h>
2 int main()
3 {
4     printf("Hello World");
5     return 0;
6 }
7
```

The status bar at the bottom indicates the current file is 'D:\Hello World.c', the language is 'C/C++', the encoding is 'Windows (CR+LF)', the theme is 'default', and the cursor is at 'Line 7, Col 1, Pos 80'. An 'Activate Windows' watermark is visible in the bottom right corner.



The screenshot shows a Windows command prompt window titled '"D:\Hello World.exe"'. The output of the program is displayed as follows:

```
Hello World
Process returned 0 (0x0)   execution time : 0.586 s
Press any key to continue.
```

Features:

Compilers

Code::Blocks supports multiple compilers, including GCC, MinGW, Digital Mars, Microsoft Visual C++, Borland C++, LLVM Clang, Watcom, LCC and the Intel C++ compiler. Although the IDE was designed for the C++ language, there is some support for other languages, including Fortran and D. A plug-in system is included to support other programming languages.

Code editor

The IDE features syntax highlighting and code folding (through its Scintilla editor component), C++ code completion, class browser, a hex editor and many other utilities. Opened files are organized into tabs. The code editor supports font and font size selection and personalized syntax highlighting colours.

Debugger

The Code::Blocks debugger has full breakpoint support. It also allows the user to debug their program by having access to the local function symbol and argument display, user-defined watches, call stack, disassembly, custom memory dump, thread switching, CPU registers and GNU Debugger Interface.

GUI designer

As of version 13.12 Code::Blocks comes with a GUI designer called wxSmith. It is a derivative port of wxWidgets version 2.9.4.[8] To make a complete wxWidgets application, the appropriate wxWidgets SDK must be installed.

User migration

Some of Code::Blocks features are targeted at users migrating from other IDE's - these include Dev-C++, Microsoft Visual C++ project import (MSVC 7 & 10), and Dev-C++ Devpak support.

Project files and build system

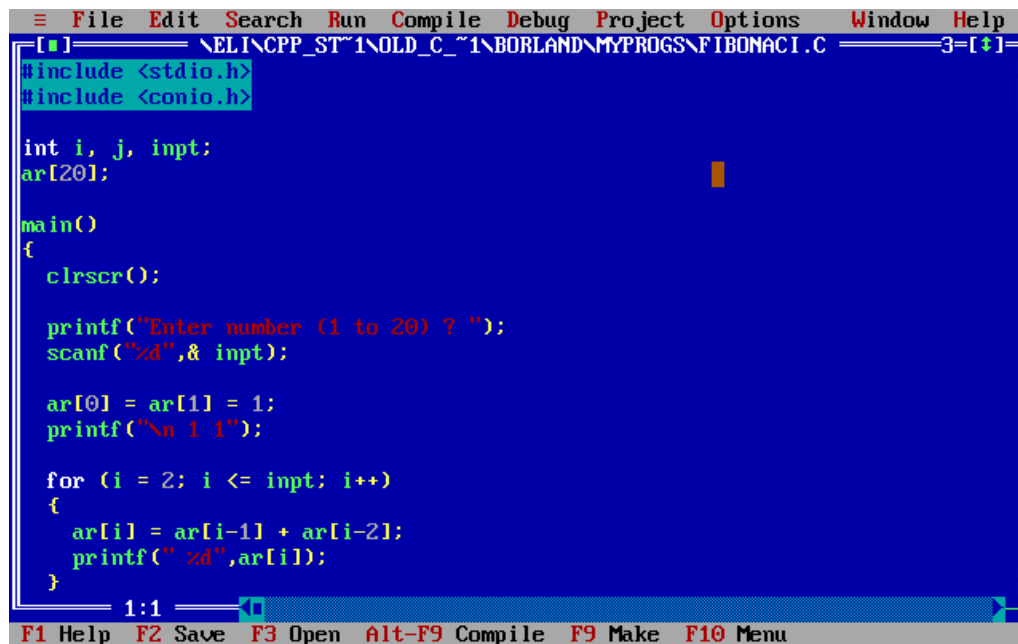
Code::Blocks uses a custom build system, which stores its information in XML-based project files. It can optionally use external makefiles, which simplifies interfacing with projects using the GNU or qmake build systems.

[N.B. We have done our project “Restaurant Management System” using Code::Blocks]

2.3.3 Borland Turbo C

Turbo C is a discontinued Integrated Development Environment and compiler for the C programming language from Borland. First introduced in 1987, it was noted for its integrated development environment, small size, fast compile speed, comprehensive manuals and low price.

In May 1990, Borland replaced Turbo C with Turbo C++. In 2006, Borland reintroduced the Turbo moniker.

A screenshot of the Turbo C++ integrated development environment. The window title is "\EL\CPP_ST~1\OLD_C_~1\BORLAND\MYPROGS\FIBONACI.C". The menu bar includes File, Edit, Search, Run, Compile, Debug, Project, Options, Window, and Help. The code editor has a blue background with text in green and red. The code is a C program to calculate Fibonacci numbers. The status bar at the bottom shows "1:1" and function key shortcuts: F1 Help, F2 Save, F3 Open, Alt-F9 Compile, F9 Make, and F10 Menu.

```
[■] \EL\CPP_ST~1\OLD_C_~1\BORLAND\MYPROGS\FIBONACI.C 3=[+]  
#include <stdio.h>  
#include <conio.h>  
  
int i, j, inpt;  
ar[20];  
  
main()  
{  
    clrscr();  
  
    printf("Enter number (1 to 20) ? ");  
    scanf("%d",& inpt);  
  
    ar[0] = ar[1] = 1;  
    printf("No 1 1");  
  
    for (i = 2; i <= inpt; i++)  
    {  
        ar[i] = ar[i-1] + ar[i-2];  
        printf(" %d",ar[i]);  
    }  
}
```

1:1

F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

Version history

Version 1.0 (May 13, 1987) offered the first integrated development environment for C on IBM PCs. Like many Borland products of the time, the software was bought from another company (in this case Wizard C by Bob Jervis), and branded with the "Turbo" name. It ran in 384 kB of memory. It allowed inline assembly with full access to C symbolic names and structures, supported all memory models, and offered optimizations for speed, size, constant folding, and jump elimination.

Version 1.5 (January 1988) was an incremental improvement over version 1.0. It included more sample programs, improved manuals and bug fixes. It was shipped on five 360 KB diskettes of uncompressed files, and came with sample C programs, including a stripped-down spreadsheet called mcalc. This version introduced the <conio.h> header file (which provided fast, PC-specific console I/O routines).

Version 2.0 (late 1988) featured the first "blue screen" version, which would be typical of all future Borland releases for MS-DOS. It was also available bundled with Turbo Assembler and Turbo Debugger. Turbo C 2.0 was also released (in Germany only) for the Atari ST; the program was not maintained by Borland, but sold and renamed PureC.

With the release of Turbo C++ 1.0 (in 1990), the two products were folded into one and the name "Turbo C" was discontinued. The C++ compiler was developed under contract by a company in San Diego, and was one of the first "true" compilers for C++ (until then, it was common to use pre-compilers that generated C code, ref. Cfront) [8].

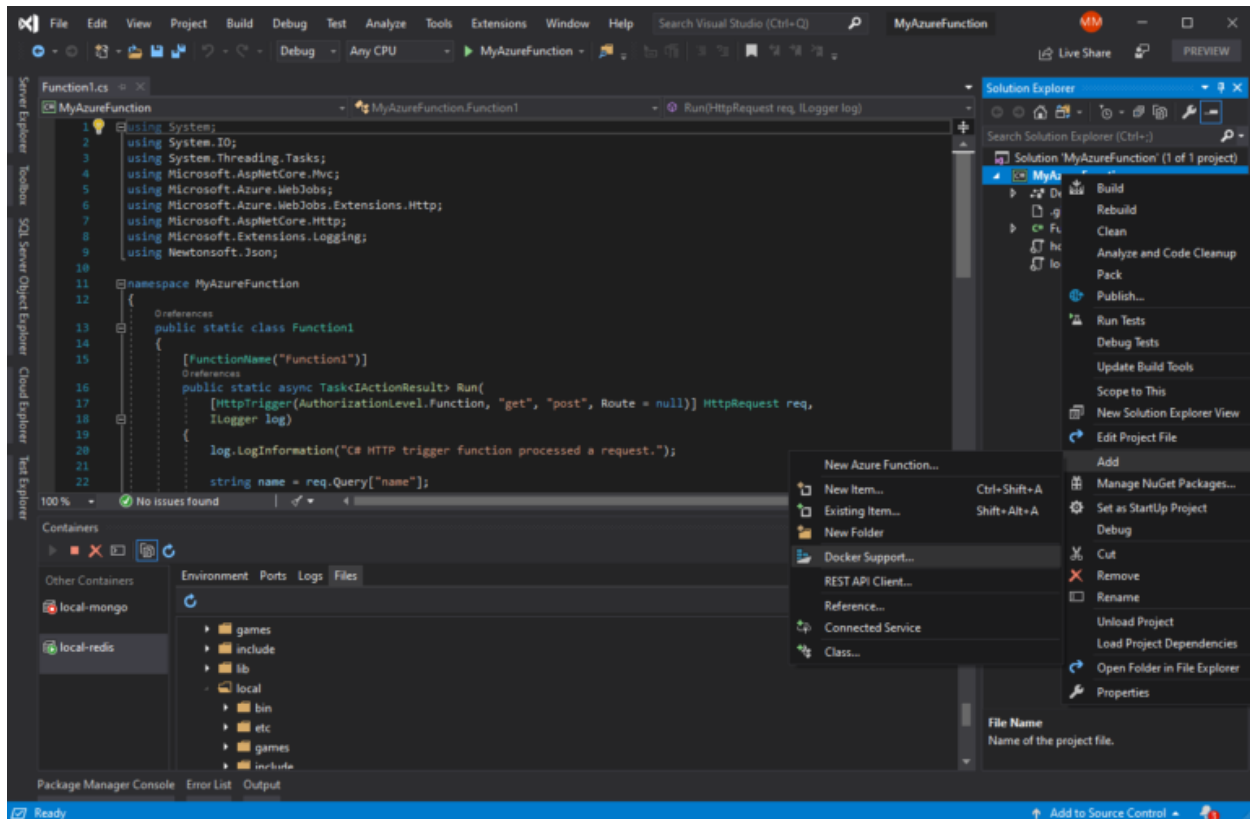
2.3.4 Microsoft Visual Studio

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a code profiler, designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug-ins that expand the functionality at almost every level—including adding support for source control systems (like Subversion and Git) and adding new toolsets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle (like the Azure DevOps client: Team Explorer).

Visual Studio supports 36 different programming languages and allows the code editor and debugger to support (to varying degrees) nearly any programming language, provided a language-specific service exists. Built-in languages include C, C++, C++/CLI, Visual Basic .NET, C#, F#, [9] JavaScript, TypeScript, XML, XSLT, HTML, and CSS. Support for other languages such as Python, Ruby, Node.js, and M among others is available via plug-ins. Java (and J#) were supported in the past.

The most basic edition of Visual Studio, the Community edition, is available free of charge. The slogan for Visual Studio Community edition is "Free, fully-featured IDE for students, open-source and individual developers" [9].



Designer

Visual Studio includes a host of visual designers to aid in the development of applications. These tools include:

Windows Forms Designer

The Windows Forms designer is used to build GUI applications using Windows Forms. Layout can be controlled by housing the controls inside other containers or locking them to the side of the form. Controls that display data (like textbox, list box and grid view) can be bound to data sources like databases or queries. Data-bound controls can be created by dragging items from the Data

Sources window onto a design surface.[33] The UI is linked with code using an event-driven programming model. The designer generates either C# or VB.NET code for the application.

WPF Designer

The WPF designer, codenamed Cider, was introduced with Visual Studio 2008. Like the Windows Forms designer it supports the drag and drop metaphor. It is used to author user interfaces targeting Windows Presentation Foundation. It supports all WPF functionality including data binding and automatic layout management. It generates XAML code for the UI. The generated XAML file is compatible with Microsoft Expression Design, the designer-oriented product. The XAML code is linked with code using a code-behind model.

Web designer/development

Visual Studio also includes a web-site editor and designer that allows web pages to be authored by dragging and dropping widgets. It is used for developing ASP.NET applications and supports HTML, CSS and JavaScript. It uses a code-behind model to link with ASP.NET code. From Visual Studio 2008 onwards, the layout engine used by the web designer is shared with Microsoft Expression Web. There is also ASP.NET MVC support for MVC technology as a separate download and ASP.NET Dynamic Data project available from Microsoft.

Class designer

The Class Designer is used to author and edit the classes (including its members and their access) using UML modeling. The Class Designer can generate C# and VB.NET code outlines for the classes and methods. It can also generate class diagrams from hand-written classes.

Data designer

The data designer can be used to graphically edit database schemas, including typed tables, primary and foreign keys and constraints. It can also be used to design queries from the graphical view.

Mapping designer

From Visual Studio 2008 onwards, the mapping designer is used by LINQ to SQL to design the mapping between database schemas and the classes that encapsulate the data. The new solution from ORM approach, ADO.NET Entity Framework, replaces and improves the old technology.

Other tools

Open Tabs Browser

The open tabs browser is used to list all open tabs and to switch between them. It is invoked using CTRL+TAB.

Properties Editor

The Properties Editor tool is used to edit properties in a GUI pane inside Visual Studio. It lists all available properties (both read-only and those which can be set) for all objects including classes, forms, web pages and other items.

Object Browser

The Object Browser is a namespace and class library browser for Microsoft .NET. It can be used to browse the namespaces (which are arranged hierarchically) in managed assemblies. The hierarchy may or may not reflect the organization in the file system.

Solution Explorer

In Visual Studio parlance, a solution is a set of code files and other resources that are used to build an application. The files in a solution are arranged hierarchically, which might or might not reflect the organization in the file system. The Solution Explorer is used to manage and browse the files in a solution.

Team Explorer

Team Explorer is used to integrate the capabilities of Azure DevOps (either Azure DevOps Services or Azure DevOps Server) into the IDE. In addition to version control integration it provides the ability to view and manage individual work items (including user stories, bugs, tasks and other documents). It is included as part of a Visual Studio installation and is also available as a standalone download.

Data Explorer

Data Explorer is used to manage databases on Microsoft SQL Server instances. It allows creation and alteration of database tables (either by issuing T-SQL commands or by using the Data designer). It can also be used to create queries and stored procedures, with the latter in either T-SQL or in managed code via SQL CLR. Debugging and IntelliSense support is available as well.

Server Explorer

The Server Explorer tool is used to manage database connections on an accessible computer. It is also used to browse running Windows Services, performance counters, Windows Event Log and message queues and use them as a data source.

Dotfuscator Community Edition

Visual Studio includes a free 'light' version of Dotfuscator

Text Generation Framework

Visual Studio includes a full text generation framework called T4 which enables Visual Studio to generate text files from templates either in the IDE or via code.

ASP.NET Web Site Administration Tool

The ASP.NET Web Site Administration Tool allows for the configuration of ASP.NET websites.

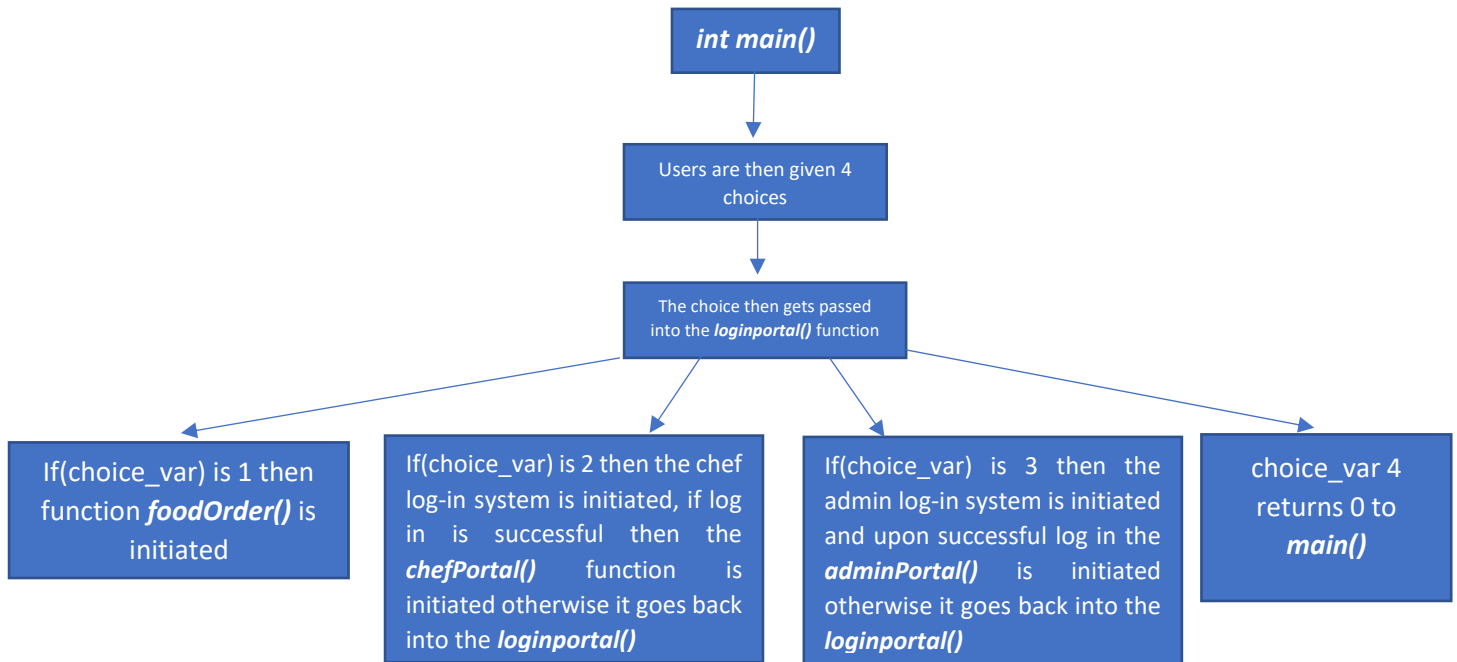
Visual Studio Tools for Office

Visual Studio Tools for Office is a SDK and an add-in for Visual Studio that includes tools for developing for the Microsoft Office suite. Previously (for Visual Studio .NET 2003 and Visual Studio 2005) it was a separate SKU that supported only Visual C# and Visual Basic languages or was included in the Team Suite. With Visual Studio 2008, it is no longer a separate SKU but is included with Professional and higher editions. A separate runtime is required when deploying VSTO solutions.

[N.B. Borland Turbo C and Microsoft Visual Studio are optional IDEs for this project.]

2.4 Used Diagram

Functions:



In the following functions:

1. This is `foodOrder()` function so that the customers can directly go to the menu bar and select food including order confirmation, payment method and feedback.
2. This function is for chef named `chefPortal()`. A chef can check orders, see admin messages and the feedbacks of customers.
3. `adminPortal()` function is for admin that the admin can check the orders, give message to chef about food quality and see the feedback.
4. The last function is for all kind of choice.

2.5 Analysis of Existing System

Maintaining digital system in restaurant and Purchasing food over the internet without the need of going physically to the restaurant is what restaurant management system and online food order system all about. Online food ordering is just like a retail store shopping that we do by going to the market, but it is done through the internet. Online food ordering has made purchasing food painless and added more fun. Restaurants and online food order software offer food description, food pictures, comparisons, price and much more and the benefits of online food purchasing is that by having direct access to consumer, the restaurants can offer foods that cater to the needs of consumer, cookies can be used for tracking the customers' selection over the internet or what is of their interest when they visit the software again. Online food purchasing makes use of digital technology for managing the flow of information, foods, and payment between consumer, software owners and suppliers. Cart is one of the important facilities provided in online food purchasing, this lets customer to browse different foods and once they select an item to purchase, they can place the item in cart, and continue browsing till the final selection. Customers can even remove the items from cart that were selected earlier before they place the final order. There are many pros of digital restaurant management system and online food purchasing system. The first pro of online food purchasing system is convenience. This software is available for 24 hours a day, and many consumers have Internet access both at work and at home. Other establishments such as online stores and schools provide access as well. Another pro of restaurant management system and online system is information and reviews. Restaurant describes foods for sale with menu, tastes and comparisons. Some restaurants even allow customers to comment or rate their food items. There are also dedicated review sites that host user reviews for different food items. Moreover, another pro of restaurant management system and online system is price and selection. Customers can compare prices of different food items and they can select the best item through menu and enjoy the item.

2.6 Conclusion

From this chapter we know the details about the related work of our project. There are two popular International and Bangladeshi online food purchasing system such as Foodpanda and HungryNaki. Because of these international online food ordering and purchasing system we have gathered many information very easily and this information are very effective for us to develop our system. Our project is an existing system. So, we can learn many things from these existing systems. Moreover, we gather knowledge and also can learn about many functionalities and modules from these existing systems. These existing systems are very useful for developing our system. Now a days these systems are very popular all over the world. That's why it is very easy for us to find out the required information for our system.

Chapter- 3: Proposed Model

3.1 Introduction

Purchasing food online is an online shopping application, which provides the online food ordering and purchasing facility available for everyone. Any type of the food will be available for the customer, and it can be easily purchased faster when compare with manual system. This type of software's concentrates more on user friendly interfaces and promotes users to purchase faster and easier. There is a facility available to do online purchase. This system has registration facility. The registration process is faster and easier compared to any existing software.

In the user interface of software there will be accomplished with options to find new food available and most purchased and customer satisfied food. Customer can avail this facility and buy the food faster. All foods in the software will be highlighted with the image of the food. By click on the image it will take you directly to the buy page, from where customers can purchase easily.

Purchasing food online is provided with customer support page. Using this page any customer can get the assistance from the available customer support executive online. There we have support over phone, email and chat. Customers can use this facility any time. As security issue pays major importance today, extra attention towards security is done in restaurant and online food purchasing software.

3.2 Feasibility Study

The objective of feasibility study is to determine whether or not the proposed system is feasible. A feasibility study is a preliminary investigation of a proposed system to decide whether the system can run smoothly with the organization, will the organization realize the benefits that are expected and to decide will the organization go for it.

Feasibility is defined as the practical extent to which a project can be performed successfully. To evaluate feasibility, a feasibility study is performed, which determines whether the solution considered to accomplish the requirements is practical and workable in the software. Information such as resource availability, cost estimation for software development, benefits of the software to the organization after it is developed and cost to be incurred on its maintenance are considered during the feasibility study. The objective of the feasibility study is to establish the reasons for developing the software that is acceptable to users, adaptable to change and conformable to established standards. Various other objectives of feasibility study are listed below.

- To analyze whether the software will meet organizational requirements
- To determine whether the software can be implemented using the current technology and within the specified budget and schedule
- To determine whether the software can be integrated with other existing software.

Three Types of feasibility study:

- Technical Feasibility.
- Operational Feasibility.
- Economic Feasibility.

3.2.1 Technical feasibility

In this, one has to test whether the system can be developed using existing technology or not. It is evident that necessary hardware and software are available for development and implementation of proposed system. We acquired the technical knowledge of working in languages, and then only we have started designing our project. The system is self-explanting and does not need any entire sophisticated training. A system has been built by concentrating on the graphical user interface concepts, the software can also be handled very easily with a novice uses. The overall time that a user needs to get trained is less than 15 minutes.

The system has been added with features of menu device and button interaction methods, which makes him the master as he starts working through the environment. As the software that were used as developing this software are very economical and are readily available in the market the

only time that is lost by the customer is just installation time. Technical feasibility performs following tasks.

- I. Analyzes the technical skills and capabilities of the software development team members.
- II. Determines whether the relevant technology is stable and established.
- III. Ascertains that the technology chosen for software development has a large number of users so that they can be consulted when problems arise or improvements are required.
- IV. It mentions new hardware requirements of proposed system.
- V. It mentions computer with new configuration requirements of proposed system.
- VI. It mentions new software requirements of the proposed system.

3.2.2 Operational Feasibility

Operational feasibility means how much the system is user interactive. In this project, the management will know the details of each project where he may be presented and the data will be maintained as decentralized and if any inquires for that particular contract can be known as per their requirements and necessities. Operational feasibility also performs following tasks.

- I. Determines whether the problems anticipated in user requirements are of high priority.
- II. Determines whether the solution suggested by the software development team is acceptable.
- III. Analyzes whether users will adapt to a new software.
- IV. Determines whether the organization is satisfied by the alternative solutions proposed by the software development team.
- V. It finds if any job reconstruction is required or not.
- VI. Watches the feelings of the customers as well as user.
- VII. System should provide right & accurate information to user or customer at right place as well as at right time.

3.2.3 Economical Feasibility

Economic feasibility is a measure of the cost-effectiveness of a project or solution. As a part of this, the costs and benefits associated with the proposed system are compared and the project is economically feasible only if tangible and intangible benefits outweigh the cost. The cost for proposed restaurant management system is outweighing the cost and efforts involved in maintaining the registers, books, files and generation of various reports. The system also reduces the administrative and technical staff to do various jobs that single software can do. So, this system is economically feasible. Economic feasibility also performs following tasks.

- I. Cost incurred on software development to produce long-term gains for an organization.
- II. Cost required to conduct full software investigation (such as requirements elicitation and requirements analysis).
- III. Cost of hardware, software, development team, and training.
- IV. Cost involves in purchase or rental of equipment.
- V. Cost of phones & mobile communication equipment.
- VI. Cost of salaries of employee.
- VII. Cost of maintenance of equipment.

3.3 Requirement Analysis

In current competitive scenario every business establishment needs quality process to increase their efficiency as well as improve their productivity. It is of vital importance that manual, time consuming and monotonous operations are automated so as to streamline the working of an organization. Since, the existing system takes more time and manpower for processing. It is keeping in mind this business philosophy that we propose a new restaurant management system and online food purchasing system. Our system will deal with all the aspects sales of popular foods. In order to start gathering requirements, first it is necessary to identify each group affected by this project and understand everyone's needs.

Nonfunctional requirements are the properties that your food must have. Think of these properties as the characteristics or qualities that make the food attractive, or usable, or fast, or reliable. Nonfunctional requirements do not alter the food's functionality. That is, the functional

requirements remain the same no matter what properties you attach to them. However, the reason that this functionality is part of the food is to give it the desired characteristics. So, you might think of the functional requirements as those that do the work, and the nonfunctional requirements as those that give character to the work. Nonfunctional requirements make up a significant part of the specification. They are important as the customer and user may well judge the food on its non-functional properties.

Thinking of the functional requirements as the business requirements- That is, if you speak with a user or one of the business people, they will describe the things that the food must do in order to complete some part of their work. Keep in mind that the requirements materials will become a contract of the food to be built. Thus, the functional requirements must fully describe the actions that the intended food can perform. I also relate it to a food you might purchase at a restaurant, if you look at the bullet features list on the back of the box, it is describing the nutrition of the food.

3.3.1 Non-Functional Requirements

There are many non-functional requirements of our system. The first nonfunctional requirement is efficiency requirement. When an online food purchasing cart is implemented customer can purchase food in an efficient manner. The next functional requirement is reliability requirement. The system should provide a reliable environment to both customers and owner. All orders should be reaching at the admin without any errors. Now another nonfunctional requirement is usability requirement. This system is designed for user interactive environment and ease of use. The implementation requirement means implementation of the system using C in front end it will be used for database connectivity. And the database part is developed by MySQL. Responsive designing is used for making the software compatible for any type of screen. Moreover, delivery requirement states that the whole system is expected to be delivered in given time.

In its first stage, the web template is required mainly to analyze the platform capabilities, show code examples to developers and attract potential customers. For this reason, all nonfunctional requirements are highly focused on those areas. Other areas of great importance as well, such as compatibility and performance, are left aside from the current project because of the excessive workload that it means.

From a developer point of view the quality of the code takes a very important role, so it should be well organized, easy to understand and reusable. Therefore, it would be considered a good practice to use variables and functions with self-explanatory names and keep a well commented code.

The platform should allow to test any web application built on top of it. In order to prove it is allowed, the template should be completed with automated functional tests, being careful of keeping these tests independent from the backend data in use. That way a change in the data, very likely to happen in a template web-shop, will not affect the results. The same principle should be applied to the code in general, to keep the template from being non-functional when the data used is different.

Although major part of the required security is located on the e-commerce and payment platforms, there are some risks server side that must be top priority when it comes to online system. For example, some data needs a careful treatment, like user related data such as addresses, passwords and payment information. Particular attention must be paid with the checkout process in order to avoid fraud.

The major non-functional requirements are describing below:

Performance

The system must be interactive and the delays involved must be less. So, in every action response of the system, there are no immediate delays. In case of opening windows forms, of popping error messages and saving the settings or sessions there is delay much below 2 seconds, In case of opening databases, sorting questions and evaluation there are no delays and the operation is performed in less than 2 seconds for opening ,sorting, computing, posting > 95% of the files. Also, when connecting to the server the delay is based editing on the distance of the 2 systems and the configuration between them so there is high probability that there will be or not a successful connection in less than 20 seconds for sake of good communication.

Safety

Information transmission should be securely transmitted to server without any changes in information.

Reliability

As the system provide the right tools for discussion, problem solving it must be made sure that the system is reliable in its operations and for securing the sensitive details.

Availability

If the internet service gets disrupted while sending information to the server, the information can be sent again for verification.

Security

The main security concern is for users account hence proper login mechanism should be used to avoid hacking. The tablet id registration is way to spam check for increasing the security.

Hence, security is provided from unwanted use of recognition software.

Usability

As the system is easy to handle and navigates in the most expected way with no delays. In that case the system program reacts accordingly and transverses quickly between its states.

3.3.2 Functional Requirements

Now our system consists of many functional requirements.

Our first functional requirement is admin login system. Admin can handle the all system. Admin can add food and category of the food. Moreover, admin can add the brand name of food and also can confirm the customer order. Admin has the ability to analysis and inquiry the sales and food report. Apart from admin can check the availability of the food. Admin can also communicate with chef about the orders and food quality.

The second functional requirement is chef login system. Here the chef can see the amount of order and what types of food a customer has ordered. He/she can also see the feedback of customers and admin's messages.

Thirdly, the most functional need is the menu option. Here customers can choose the food they need and choose and they can buy food at fixed or offered prices. We have used modern payment system in this software which makes it easy for customers to buy food.

In our system there are many functional requirements are existing for the customer. Customer can see the details of the food and also can see the available food. Customer can add the food to cart for order. Moreover, Customer can choose his food by price filtering and brand filtering.

Another functional requirement is confirming option. Customer must fill up the confirmation form to order any food by giving all the details of him. Apart from if any customer wants to contact with us then he or she can contact with us very easily by filling up the contact us form. That's all about the functional requirements of our project.

In order to prove the value of the platform and identify any possible lack of functionality, the system should have all the common features of a regular software. Accordingly, it has been considered that the initial appropriate set of functionalities for this project include those related to browsing and purchasing foods, as well as management of a customer account.

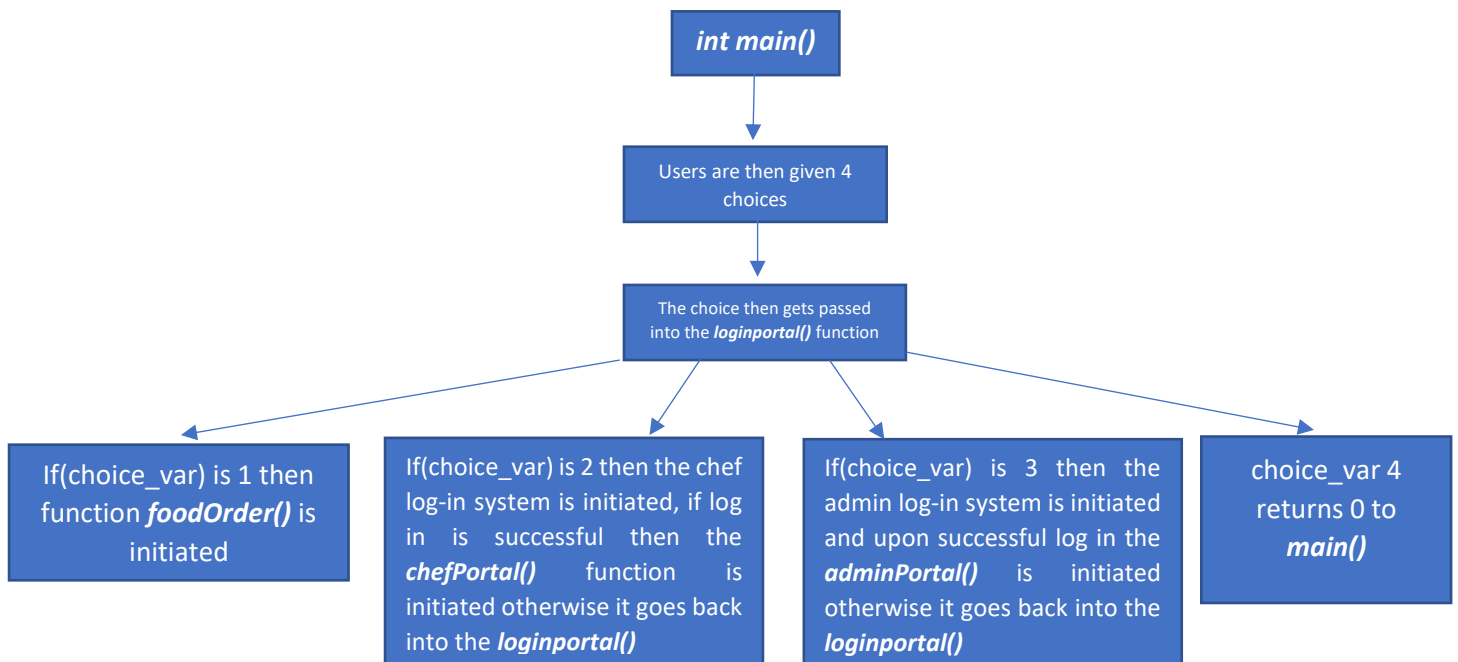
3.4 System Design

System design is the solution for the creation of a new system or software. This phase focuses on the detailed implementation of the feasible system. It emphasizes on translating design. Specifications to performance specification. System design has two phases of development. They are logical design and physical design.

During logical design phase the analyst describes inputs (sources), outputs (destinations), databases (data stores) and procedures (data flows) all in a format that meets the user requirements. The analyst also specifies the needs of the user at a level that virtually determines the information flow in and out of the system and the data resources. Here the logical design is done through data flow diagrams and database design. The physical design is followed by physical design or coding. Physical design produces the working system by defining the design specifications which specify exactly what the candidate system must do. The programmers write the necessary programs that accept input from the user perform necessary processing on accepted data and produce the required report on a hard copy or display it on the screen.

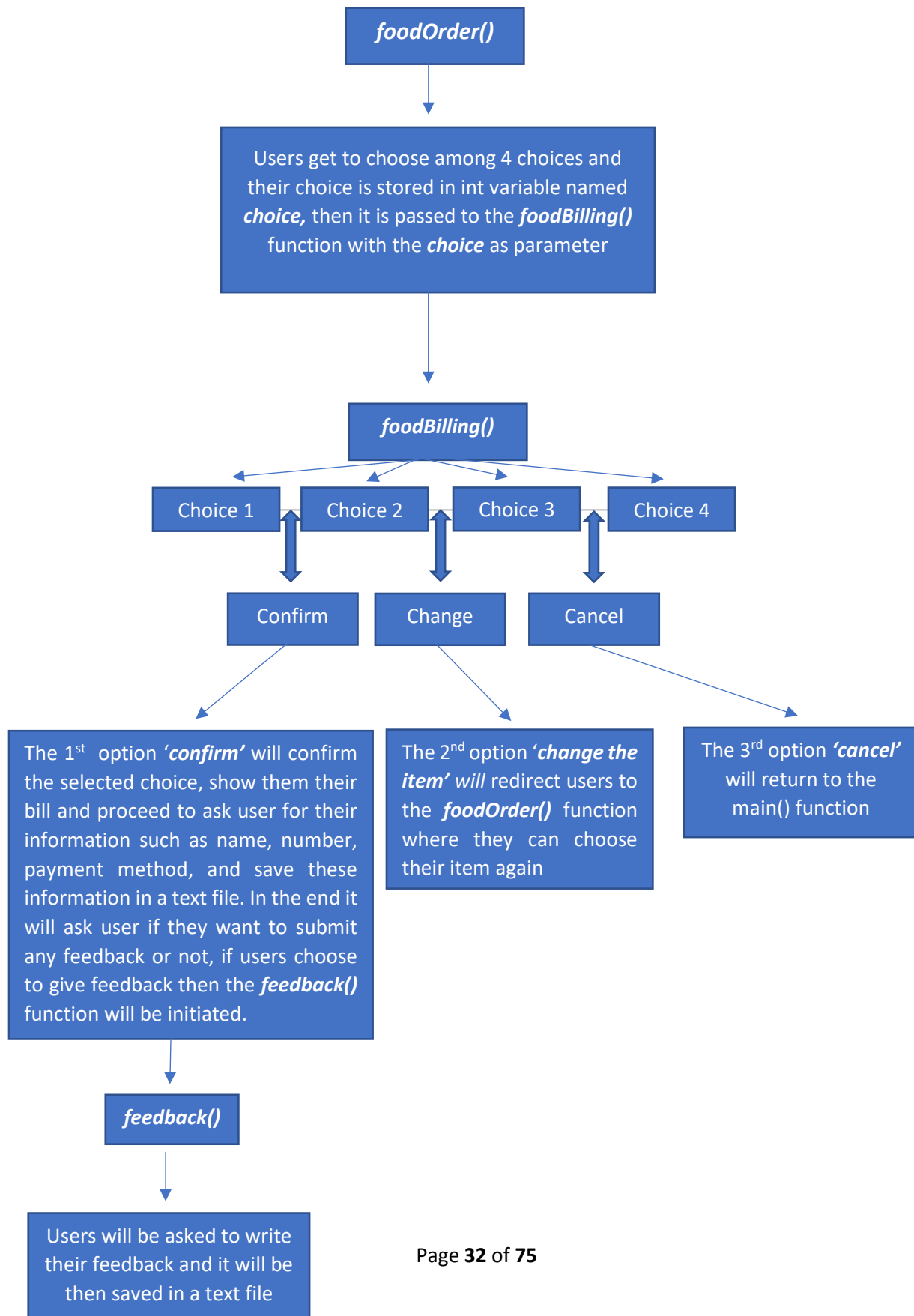
3.4.1 Diagram

The first few functions:

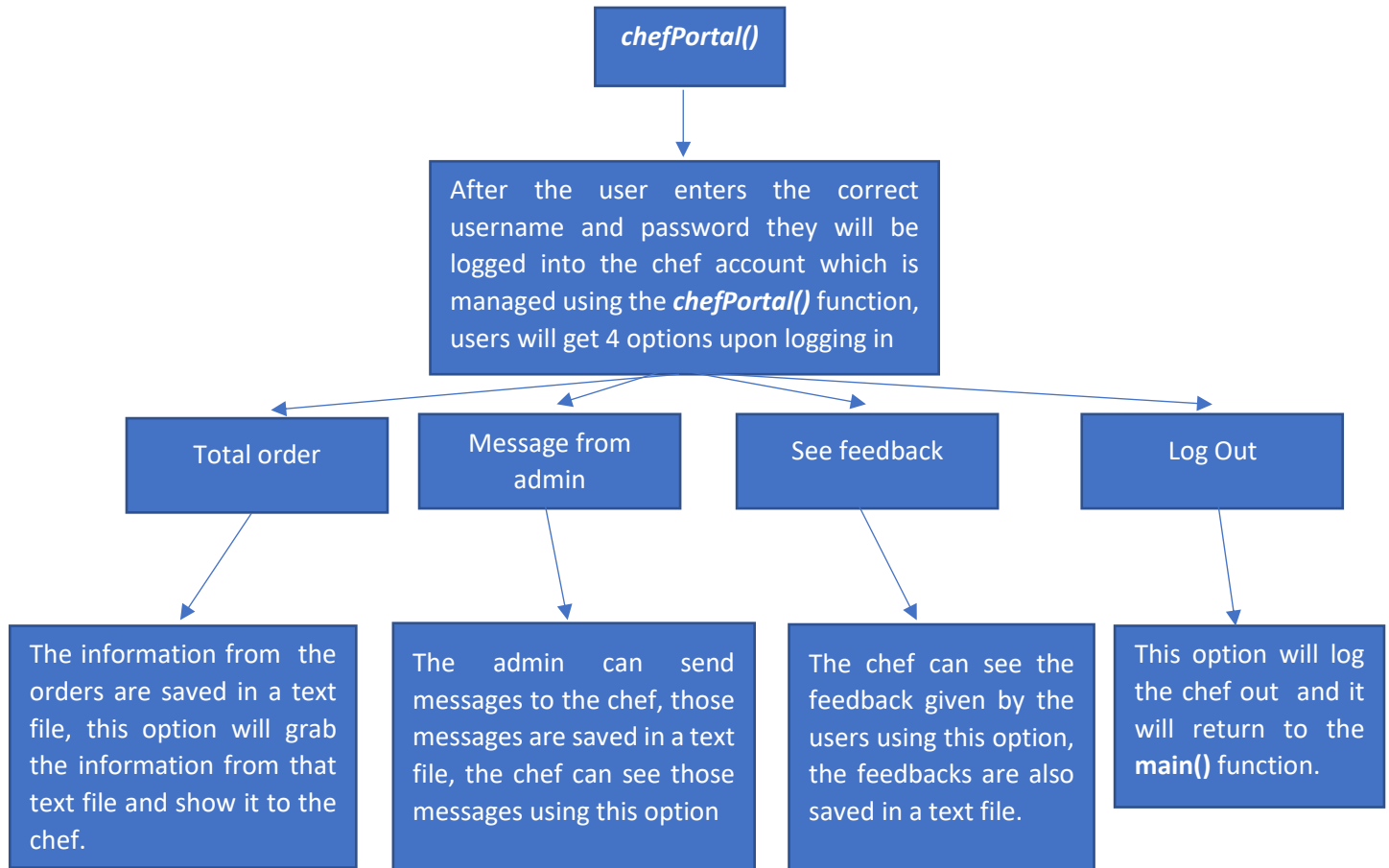


(All of these three options/core systems will be explained below)

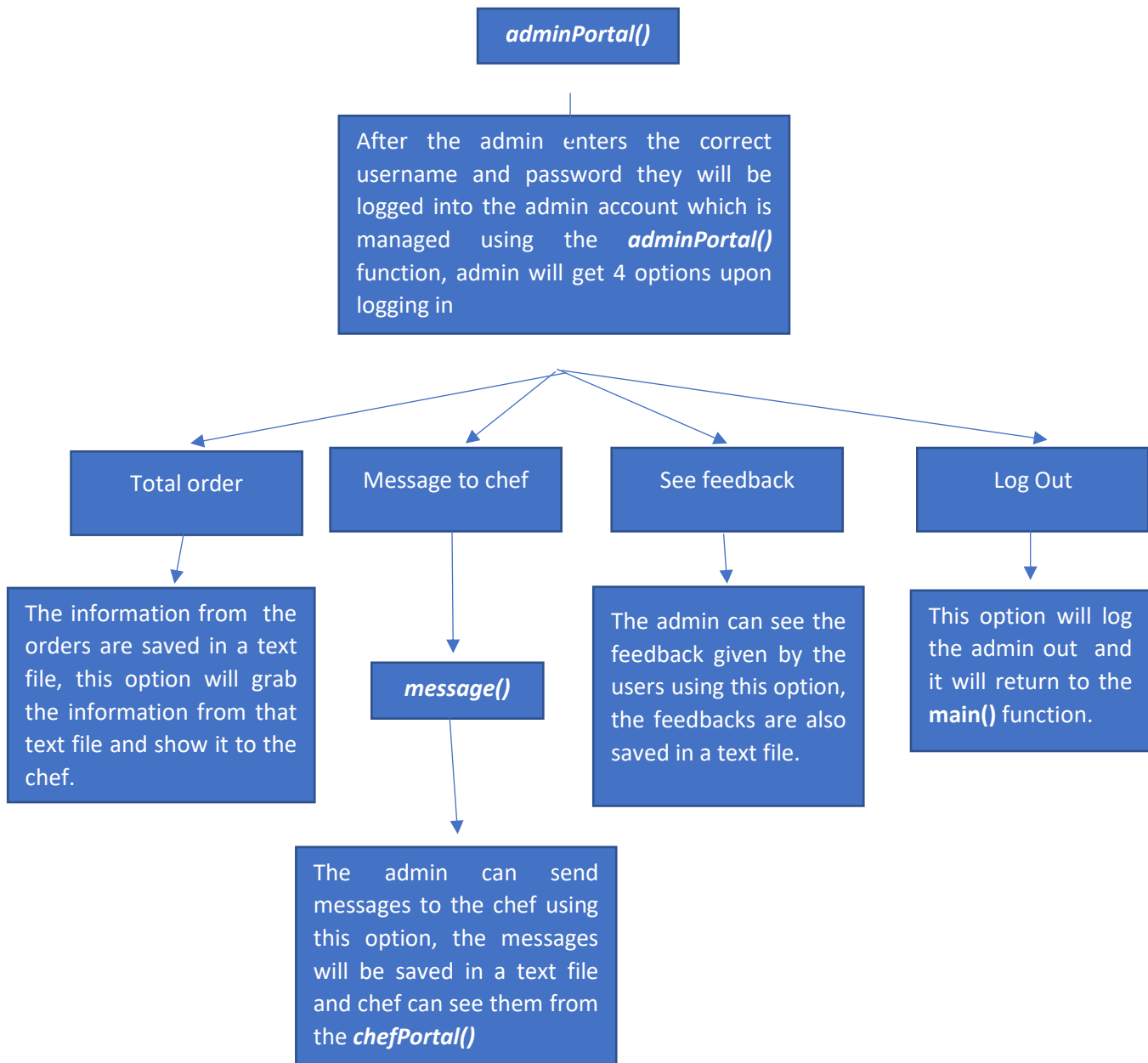
The food ordering and billing system:



Chef login system:



Admin login system:



3.4.2 Function

foodorder():

```
int foodOrder()  
{  
    printf("\nHere are the list of food item we have today...\n\n");  
    printf("1. Regular Chicken Burger\n");  
    printf("2. Pizza\n");  
    printf("3. BBQ Sub Sandwich\n");  
    printf("4. Regular Cold Coffee\n");  
  
    printf("Enter your choice: ");  
    scanf("%d", &choice);  
  
    foodBilling(choice);  
}
```

chefportal():

```
int chefPortal()  
{  
    printf("\n\tWelcome to chef portal.\n\n");  
    printf("1. Total Order\n");  
    printf("2. Message from admin\n");  
    printf("3. See Feedback\n");  
    printf("4. Logout\n\n");  
    printf("Enter your choice: ");  
    scanf("%d", &choice);  
}
```

adminportal():

```
int adminPortal()  
{  
    printf("\n\tWelcome to admin portal.\n\n");  
    printf("1. Total Order\n");  
    printf("2. Message to chef\n");  
    printf("3. See Feedback\n");  
    printf("4. Logout\n\n");  
    printf("Enter your choice: ");  
    scanf("%d", &choice);  
}
```

3.5 Implementations

Implementation is the stage where the theoretical design is turned into a working system. The most crucial stage in achieving a new successful system and in giving confidence on the new system for the users that it will work efficiently and effectively. The system can be implemented only after thorough testing is done and if it is found to work according to the specification. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the changeover and an evaluation of change over methods a part from planning. Two major tasks of preparing the implementation are education and training of the users and testing of the system. The more complex the system being implemented, the more involved will be the systems analysis and design effort required just for implementation. The implementation phase comprises of several activities. The required hardware and software acquisition is carried out. The system may require some software to be developed.

3.5.1 Front End

The front-end part, comprising of pages which users can view, is created by using templates so that it can expedite design and loading processes. This part, in the appearance, the main menu is consistent, thus it is easy to use. Considered as the most frequently access when comparing with other parts of the software, front end pages have to use template, and in turn, saving bandwidth and number of database accesses. This part is consisting of C programming language.

In front end we have developed and designed many function modules such as admin login, chef login, menu etc. by C language including structure and file. For purchasing food from our project, a customer has to install out application or software in mobile or laptop. After that they can be able to see the interface and designs of our project. All the pages can be seen from home page. Then he/she can order food using some simple steps. He/she have to confirm order and select payment method. After completing the payment, he/she can write feedback for developing our project or system. That's all about our project designing development or front-end part.

3.5.2 Back End

This part provides facility for each restaurant owner to edit and modify information in his own store. Providing validation check for member identification, the back-end system can securely protect users' proprietary information. In addition, all page views employ session variables to deter manually defined variables by users. Applying user friendly approach, and focusing on web programming inexperience, the user can effortlessly manage his back-end information. Inside the back end, users can control and view all information. Besides that, using content management design, the back-end part encompasses with these modules: food management, restaurant information management, web board management, member management, promotion management, banner management, currency management and article management. This part is consisting of C programming language.

In back end part each function module in system background uses the same design scheme, which involves management functions of food, food order reports, food list and orders to add, delete, update and check. They all have adopted the front-end page requests sent to the controller action, invoke the service of business processing module, the last action to deal with the results back to the view layer, and the service layer is call one or more DAO methods for data processing.

We have developed admin login module for management of our system. In order to enter background management system, administrator must log into our system by providing username and password. When the administrator input correct user name and password, he can enter the main interface of the system background. In background of the system admin can manage the product information, order information, product category, sales report, and product report. All the background system is developed by using C program Language.

Here, an admin can see the sales report, total order, see the feedbacks of customers and can instruct the chef about food. A chef can see the message of admin, total order and feedback. That's all about the back-end development of our system.

3.6 Conclusion

We have discussed in this chapter about the feasibility study of online food perching system and restaurant management system. Apart from the requirement analysis, system design and implementation are described in this chapter. In this chapter, used diagram and used functions are described. Here, also described the screenshot of functions. In requirement analysis here discussed about the functional and nonfunctional requirement of our system. Moreover, in implementation here discussed how we implemented our system. That's all about the proposed system chapter of our project.

Chapter- 4: Experimental Result

4.1 Introduction

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work food or product. It provides a way to check the quality, functionality of components, sub-assemblies, assemblies and / or a finished food it is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

This chapter mainly describes the qualitative practice to be used to provide data to examine the issues acknowledged and extend the understanding of consumer value creation in the framework of what and why consumers purchase online. Tentative and descriptive research can provide the multiple outlooks necessary to obtain multiple approvals of online, offline and channel switching behavior during the buying decision process. This typically involves sampling the population, surveying them and using inferential statistics to analyze the responses. The focus of the analysis is to expect the determining factors influencing, in this case, what and why consumers purchase online and offline as well, why they switch from one way to another. The data gathered during the depth interviews were used to identify common questions concerning consumer behavior as it relates to the pure online and offline buying process as well as channel switching from one trade channel to another during the buying choice process. Deepness of the interviews and concentration groups provide an efficient means of spreading and emerging theoretical concepts to improve the ultimate research design. And are used in this research to better appreciate what and why consumers use the Internet to purchase food and in specific why they choose one channel over another in general. Below defines how this qualitative phase of the research is directed and classifies which of the research objectives each activity supports for both the depth interviews.

4.2 Result Analysis

In this chapter we have analyzed our system. By analyzing we can see the output of the different types of report such as sales report, food report, all food list and feedback of customers. By analyzing we have understood that out all the modules are working. An admin can handle our system by logged in successfully. Some of the reports are describing as follows.

4.2.1 Sales Report

Only admin and chef have the permission to see the sales report. And to see it, admin and chef have to login in this software by their portals and they can see it in “Total Order” option.

Admin Panel

```
3 Food

1. Menu
2. Chef Login
3. Admin Login
4. Exit

Enter your choice: 3

Admin profile

Enter user name: siam
Enter password: 
successfully logged in...

Welcome to admin portal.

1. Total Order
2. Message to chef
3. See Feedback
4. Logout

Enter your choice: 1
Chamak Sarkar Pizza 3 01316055805 BDT: 540
Siam BBQ Sub Sandwich 5 01485738293 BDT: 750
Pallab Pizza 3 13413425 BDT: 540
Bijoy BBQ Sub Sandwich 5 01367485746 BDT: 750
Pallab Sarkar Pizza 2 01847361253 BDT: 360
Nazmus Sakib Regular Cold Coffee 6 0173564632 BDT: 360
MD Nazmul Ahsan Regular Chicken Burger 5 01746287489 BDT: 500

Press any key to continue . . .
```

Chef Panel

```
3 Food

1. Menu
2. Chef Login
3. Admin Login
4. Exit

Enter your choice: 2
Chef profile

Enter user name: chamak
Enter password: 
successfully logged in...

Welcome to chef portal.

1. Total Order
2. Message from admin
3. See Feedback
4. Logout

Enter your choice: 1
Chamak Sarkar Pizza 3 01316055805 BDT: 540
Siam BBQ Sub Sandwich 5 01485738293 BDT: 750
Pallab Pizza 3 13413425 BDT: 540
Bijoy BBQ Sub Sandwich 5 01367485746 BDT: 750
Pallab Sarkar Pizza 2 01847361253 BDT: 360
Nazmus Sakib Regular Cold Coffee 6 0173564632 BDT: 360
MD Nazmul Ahsan Regular Chicken Burger 5 01746287489 BDT: 500

Press any key to continue . . .
```

4.2.2 Food List

To see the food list, customers have to click on “Menu” option and then they can see the food items.

```
3 Food

1. Menu
2. Chef Login
3. Admin Login
4. Exit

Enter your choice: 1

Here are the list of food item we have today...

1. Regular Chicken Burger
2. Pizza
3. BBQ Sub Sandwich
4. Regular Cold Coffee
Enter your choice:
```

4.2.3 Feedback

Only admin and chef have the permission to see the feedback of the customers. And to see it, admin and chef have to login in this software by their panels and they can see it in “See Feedback” option.

Admin Panel

```
3 Food

1. Menu
2. Chef Login
3. Admin Login
4. Exit

Enter your choice: 3

Admin profile

Enter user name: siam
Enter password: 
successfully logged in...

Welcome to admin portal.

1. Total Order
2. Message to chef
3. See Feedback
4. Logout

Enter your choice: 3
Very nice management and the foods are so delicious...
press any key to continue . . .
```

Chef Panel

```
3 Food

1. Menu
2. Chef Login
3. Admin Login
4. Exit

Enter your choice: 2
Chef profile

Enter user name: chamak
Enter password: 
successfully logged in...

Welcome to chef portal.

1. Total Order
2. Message from admin
3. See Feedback
4. Logout

Enter your choice: 3
Very nice management and the foods are so delicious...
Press any key to continue . . .
```

4.3 Application

In real-time, digital restaurant management and online food purchasing system is very popular system. Every company of the world wants to sell their foods on online because of cost consuming. Nowadays customer is not wanting to come in restaurant or shop for buying foods, they can save their time by online purchasing and they can get their food at home. That's why all the companies want to sell their foods on online. Online restaurants typically enable customers to use "search" features to find specific food items. Online customers must have access to the Internet and a valid method of payment in order to complete a transaction, such as Bkash, Rocket, a credit card, an Interact-enabled debit card, or a service such as PayPal. For ordered food, the delivery boy ships the products to the customs. The largest of these online food purchasing applications are Uber Eats, Foodpanda, HungryNaki etc. User can order any food from home; they just need internet connection. User can access online food ordering system and restaurant organizations by smart

phone, tablet and computer. Any user can get his needed food from restaurant to his/her home by paying online transaction system. In our system customers can buy food only by cash on delivery, Bkash and Rocket. We will develop online transaction in future. We will be developed price filtering option which is very user-friendly. Customers can choose their food by selecting the price by price range bar. In our system customers will also choose their food by categories. They can also see the available foods which is very helpful for them to order. That means our system has stock availability. Customers can also use our system in computer, smart phone and tablet. We have developed searching option in our system where customers can search their foods by a search bar and they can get their desire foods.

There is no doubt that selling online can bring many added benefits to certain businesses. It is the responsibility of the business owner to inform their selves, assess the situation, make a decision and take the necessary steps towards reaching their ultimate goals.

4.4 Conclusion

In this chapter we have discussed about the experimental results of our project. Here, we have analyzed the sales report and food item report. We can know about which category of the foods are sold more than others category of the foods. Here we have also analyzed that which brands foods are sold more. We can know about the daily sales report of our system by this daily sales report and also can know which category of the foods are sold more than others and also food list. We have discussed about the feedback customer's feedback. Moreover, we have discussed about the application, financial benefits and advantages of online food ordering system and restaurant management system. That's all about the experimental results.

Chapter- 5: User Manual

5.1 Introduction

To be used efficiently, all computer software needs certain hardware components or the other software resources to be present on a computer. These pre-requisites are known as (computer) system requirements and are often used as a guideline as opposed to an absolute rule. Most software defines two sets of system requirements: minimum and recommended. With increasing demand for higher processing power and resources in newer versions of software, system requirements tend to increase over time. Industry analysts suggest that this trend plays a bigger part in driving upgrades to existing computer systems than technological advancements.

5.2 System Requirements

5.2.1 Hardware Requirements

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware. A hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in case of operating systems. An HCL lists tested, compatibility and sometimes incompatible hardware devices for a particular operating system or application. The following sub-sections discuss the various aspects of hardware requirements.

Hardware Requirements for Server

Processor	: Intel dual Core, Core i3
Ram	: 1 GB
Hard Disk	: 80 GB
CPU Speed	: 2.6 GHz
Monitor	: EGA / SVGA (display), 800 × 600 24 bits True Color.

Hardware Requirements for Client

Processor	: Pentium 4, Intel dual Core, Core i3
Ram	: 512 MB
Hard Disk	: 40 GB (minimum 8 GB for mobile or tablet)
CPU Speed	: 2.6 GHz
Monitor	: EGA / SVGA (display), 800 × 600 24 bits True Color.
Mobile Devices	: All the mobile devices & Tabs

5.2.2 Software Requirements

Software Requirements deal with defining software resource requirements and pre-requisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or pre-requisites are generally not included in the software installation package and need to be installed separately before the software is installed.

Software Requirements for Server

Operating System	: Windows 7/ XP/8/10
Front End	: C
Database	: Notepad (save data)
Dependency Manager	: Composer
Local server setup package	: Code::Blocks and Notepad

Software Requirements for Client

Operating System	: Windows 7/XP/8/10
Browsers	: Chrome, Firefox, Opera Mini (All the browsers)

5.3 User Interfaces

5.3.1 Home Page

```
3 Food

1. Menu
2. Chef Login
3. Admin Login
4. Exit

Enter your choice:
```

This is the home page of our project. Customer can see the all login pages including menu page of our software in home page. From here, a customer can see food by clicking “Menu” option.

5.3.2 Login

Admin Login Page

```
3 Food

1. Menu
2. Chef Login
3. Admin Login
4. Exit

Enter your choice: 3

Admin profile

Enter user name: siam
Enter password: 1998
successfully logged in...

Welcome to admin portal.

1. Total Order
2. Message to chef
3. See Feedback
4. Logout

Enter your choice:
```

Admin have to login from this “Admin Login” option from home page and then admin have to insert user name and password to login.

Chef Login Page

```
3 Food

1. Menu
2. Chef Login
3. Admin Login
4. Exit

Enter your choice: 2
Chef profile

Enter user name: chamak
Enter password: 2001
successfully logged in...

Welcome to chef portal.

1. Total Order
2. Message from admin
3. See Feedback
4. Logout

Enter your choice:
```

Chef have to login from this “Chef Login” option from home page and then admin have to enter user name and password to login.

5.3.3 Order/Menu

```
3 Food

1. Menu
2. Chef Login
3. Admin Login
4. Exit

Enter your choice: 1

Here are the list of food item we have today...

1. Regular Chicken Burger
2. Pizza
3. BBQ Sub Sandwich
4. Regular Cold Coffee
Enter your choice:
```

Here, a customer can see the available food items and can order food.

5.3.4 Confirm Order Option

```
3 Food

1. Menu
2. Chef Login
3. Admin Login
4. Exit


Enter your choice: 1

Here are the list of food item we have today...

1. Regular Chicken Burger
2. Pizza
3. BBQ Sub Sandwich
4. Regular Cold Coffee
Enter your choice: 1

Regular Chicken Burger
price: 100tk
Cooking time: 30min (maximum)

Quantity: 1
1.confirm
2.change item
3.cancel
Enter your choice: 1
Enter your name:
```



By this option, a customer has to confirm order to purchase a food and there are three confirmation options. He/she can change order or cancel order but to purchase a food, he/she must click on confirm. After confirming order, he/she has to enter his/her name.

5.3.5 Payment Option

```
3 Food

1. Menu
2. Chef Login
3. Admin Login
4. Exit

Enter your choice: 1

Here are the list of food item we have today...

1. Regular Chicken Burger
2. Pizza
3. BBQ Sub Sandwich
4. Regular Cold Coffee
Enter your choice: 1

Regular Chicken Burger
price: 100tk
Cooking time: 30min (maximum)

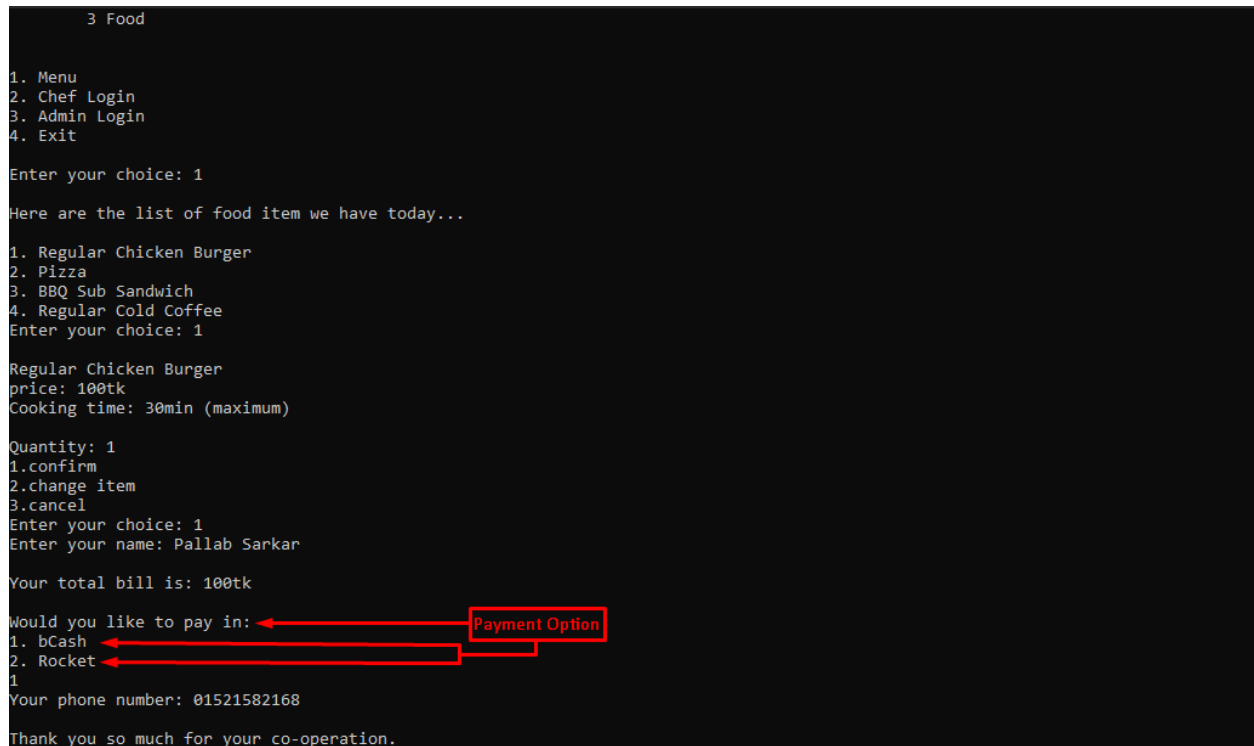
Quantity: 1
1.confirm
2.change item
3.cancel
Enter your choice: 1
Enter your name: Pallab Sarkar

Your total bill is: 100tk

Would you like to pay in:
1. bCash
2. Rocket
1

Your phone number: 01521582168

Thank you so much for your co-operation.
```



After confirming order, the customer have to pay using digital payment method. He/she has to select Bkash or Rocket payment option and then enter contact number.

5.3.6 Feedback

```
3 Food

1. Menu
2. Chef Login
3. Admin Login
4. Exit

Enter your choice: 1

Here are the list of food item we have today...

1. Regular Chicken Burger
2. Pizza
3. BBQ Sub Sandwich
4. Regular Cold Coffee
Enter your choice: 1

Regular Chicken Burger
price: 100tk
Cooking time: 30min (maximum)

Quantity: 1
1.confirm
2.change item
3.cancel
Enter your choice: 1
Enter your name: Pallab Sarkar

Your total bill is: 100tk

Would you like to pay in:
1. bCash
2. Rocket
1
Your phone number: 01521582168

Thank you so much for your co-operation.
Please wait a bit. Your desire food will be served very soon :)

Do you want to give any feedback? ('y' or 'n'): y
Enter your Feedback: This is a great software
Press any key to continue . . .
```

After completing the payment, a customer can write any feedback in the “Do you want to give any feedback” option. After clicking “y”, the customer can write his/her feedback or comment in this section.

5.4 Conclusion

In this chapter we have discussed about the system requirements. From there anyone can know about the hardware requirements and software requirements. Then we have putted the screenshot of our projects interface and described about the functionalities of our screenshotted pages. Apart from here we can know about the uses of our system. Our interface is very user interactive. So users can use it very easily and order any food without any hesitation.

Chapter- 6: Conclusion and Further Work

6.1 Conclusion

The Internet has become a major resource in modern business, thus restaurant management system and online food purchasing system have gained significance not only from the entrepreneur's but also from the customers point of view. For the entrepreneur, online food purchasing system generates a new business opportunity and for the customer, it makes the comparative food purchasing possible. A cart design must be accompanied with user-friendly food ordering cart application logic. It should be convenient for the customer to view the items of their cart and to be able to remove or add items to their cart. The cart application described in this project provides a number of features that are designed to make the customer more comfortable. This project helps in understanding the creation of an interactive web page and technologies user to implement it. During the course of the project, we have across the wide variety of problems and difficulties. We have learned the appropriate intricate working behind the dynamic website, how tricky data manipulation can be occurred sometimes but we have done everything.

Successfully, the system has been designed in response to the system analysis. All possible error in the program have been eliminated. Necessary validation techniques have been used and normal, abnormal and extremely data was used to test the system. However, doing this project has been a good boost to our confidence as the future IT member of our global village.

Today the use of software is becoming vital in the field of Restaurant Management System- **"3 Food"**. We try our best to develop nice looking, powerful, user friendly and secured software for personal or professional use. We think it is helpful for general people of all categories. We try to fulfill all necessary requirement and feature that provide other online base restaurants, health care or medical service software.

This software is not totally dynamic but it is extendable. If the software needs to extend then we must have to work little on the database. So, we have the demand to develop a software that considers all the issues. Keeping all these things in mind we have tried our best to build a software

which is secured, dynamic, extendable and reusable. There still have the opportunity to add other features that are not mentioned here.

6.2 Limitations

Though we have tried our level best to make our system flawless and user friendly by using the modern technologies, some minor functional and design inconsistencies exist in our system due to time constraint, design of prototype and cost constraints. The limitations of our system are:

- No comparison between foods.
- No personalized inbox to the user.
- No visa and master card payment system.

6.3 Future Works

Our system is developed based on demand of user's satisfaction and facilities. In our system we have used the modern web technologies to make our system fast, convenient and efficient for all of the personnel mentioned. Due to time and cost constraint it was not possible to fulfill all requirements and functionalities those were planned. But in future these planned functionalities and more improvement will be possible to pursue. The functionalities to be implemented are:

- Add to compare between foods.
- Providing personalized inbox to the user.
- Add master card and visa card system.

References

1. <https://www.coursehero.com/file/21848251/Smart-Restaurant-Management-System/>
2. <https://www.slideshare.net/venkatesm/venkateswaran2010new-microsoft-powerpointpresentation>
3. https://www.academia.edu/38342821/Restaurant_Management_System_Synopsis
4. <https://en.wikipedia.org/wiki/Foodpanda>
5. https://en.everybodywiki.com/HungryNaki.com_Ltd.
6. <https://www.javatpoint.com/c-programming-language-tutorial>
7. <https://en.wikipedia.org/wiki/Code::Blocks>
8. https://en.wikipedia.org/wiki/Borland_Turbo_C
9. https://en.wikipedia.org/wiki/Microsoft_Visual_Studio
10. Various links provided by google.com.

Appendices

Language: C

Technology: Microsoft Visual Studio

For saving Database using .txt (Notepad)

File Edit Selection View Go Run Terminal Help

EXPLORER: TIMELINE

Loading timeline for Untitled-1...

#include <stdio.h> Untitled-1

```
1  #include <stdio.h>
2  #include <conio.h>
3  #include <string.h>
4  #include <stdlib.h>
5  #include <ctype.h>
6
7  struct user
8  {
9      char name[30], pass[30];
10 }admin, chef;
11
12 struct customer
13 {
14     char name[30], food[30];
15     int amount;
16 }c[100];
17
18 int loginPortal(int a);
19 int adminPortal();
20 int chefPortal();
21 int foodOrder();
22 int foodBilling(int a);
23 int feedback();
24 void messages();
25 int choice;
26 int main()
27 {
28     printf("\tFIVE STAR HOTEL\n\n");
29     printf("1. Order Food\n");
30     printf("2. Chef Login\n");
31     printf("3. Admin Login\n");
32     printf("4. Exit\n\n");
33     printf("Enter your choice: ");
34     scanf("%d", &choice);
```

The image displays two screenshots of a Visual Studio Code editor window, showing C code for a login portal. The top screenshot shows the definition of the `loginPortal` function, and the bottom screenshot shows the main function logic.

Top Screenshot (loginPortal function):

```
#include <stdio.h>

int loginPortal(int a)
{
    FILE *fp;
    char name[30], pass[30], ch;

    if(a!=4){
        if(a==1)
            foodOrder();
        else if(a==3){
            printf("\nAdmin profile\n\n");
            fp=fopen("project of resturent management admin pass.txt", "r");
            fflush(stdin);
            printf("Enter user name: ");
            gets(name);
            fflush(stdin);
            printf("Enter password: ");
            scanf("%d", pass);
            /*int i;
            for(i=0; i!=13; i++)
            {
                ch = getch();
                pass[i] = ch;
                ch = '*' ;
                printf("%c", ch);
            }*/

            fflush(stdin);
            fscanf(fp, "%s", admin.name);
        }
    }
}
```

Bottom Screenshot (main function logic):

```
fflush(stdin);
fscanf(fp, "%s", admin.name);
fflush(stdin);
fscanf(fp, "%d", admin.pass);

if(strcmp(admin.name, name) == NULL && strcmp(admin.pass, pass) == NULL){
    printf("successfully logged in...\n\n");
    adminPortal();
}

else if(strcmp(admin.name, name) == NULL && strcmp(admin.pass, pass) != NULL){
    printf("password is incorrect!!\n\n");
    printf("try again.\n\n");

    fclose(fp);
    loginPortal(a);
}

else if(strcmp(admin.name, name) != NULL && strcmp(admin.pass, pass) == NULL){
    printf("user name is incorrect!!\n\n");
    printf("try again.\n\n");

    fclose(fp);
    loginPortal(a);
}

else if(strcmp(admin.name, name) != NULL && strcmp(admin.pass, pass) != NULL){
    printf("user name and password both are incorrect!!\n\n");
    printf("try again.\n\n");

    fclose(fp);
    loginPortal(a);
}

else if(a==2){
    // ...
}
```

The image displays two screenshots of a Visual Studio Code editor window, showing C code for a restaurant management system. The editor is titled "#include <stdio.h> • Untitled-1 - Visual Studio Code".

Top Screenshot (Lines 98-131):

```
98 }
99 }
100 else if(a==2){
101     printf("chef profile\n\n");
102     fp=fopen("project of resturent management chef pass.txt", "r");
103
104     fflush(stdin);
105     printf("Enter user name: ");
106     gets(name);
107     fflush(stdin);
108     printf("Enter password: ");
109     scanf("%d", &pass);
110
111     fscanf(fp,"%s", chef.name);
112     fscanf(fp,"%d", chef.pass);
113
114
115     if(strcmp(chef.name, name) == NULL && strcmp(chef.pass, pass) ==NULL){
116         printf("successfully logged in...\n\n");
117         chefPortal();
118     }
119     else if(strcmp(chef.name, name) == NULL && strcmp(chef.pass, pass) !=NULL){
120         printf("password is incorrect!!\n");
121         printf("try again.\n");
122
123         fclose(fp);
124         loginPortal(a);
125     }
126     else if(strcmp(chef.name, name) != NULL && strcmp(chef.pass, pass) ==NULL){
127         printf("user name is incorrect!!\n");
128         printf("try again.\n");
129
130         fclose(fp);
131         loginPortal(a);
```

Bottom Screenshot (Lines 130-163):

```
130         fclose(fp);
131         loginPortal(a);
132     }
133     else if(strcmp(chef.name, name) != NULL && strcmp(chef.pass, pass) != NULL){
134         printf("user name and password both are incorrect!!\n");
135         printf("try again.\n");
136
137         fclose(fp);
138         loginPortal(a);
139     }
140 }
141 else
142 {
143     printf("Wrong Choice!!\n\n\n");
144     main();
145 }
146 }
147 }
148
149
150 int foodOrder()
151 {
152     printf("\nHere are the list of meal we have today...\n\n");
153     printf("1. Biryani\n");
154     printf("2. Korma\n");
155     printf("3. Teheri\n");
156     printf("4. Kacchi\n");
157
158     printf("Enter your choice: ");
159     scanf("%d", &choice);
160
161     foodBilling(choice);
162 }
163 }
```

```
#include <stdio.h>
...
162 }
163
164 int foodBilling(int a)
165 {
166     char name[30], address[100], num[20], fb, message[300];
167     int method, plate;
168     FILE *fp;
169     if(a==1)
170     {
171         printf("\nBiriya\n");
172         printf("price: 150tk\n");
173         printf("cooking time: 30min(maximum)\n");
174         printf("How many plate of Biriya you want to take: ");
175         fflush(stdin);
176         scanf("%d", &plate);
177         printf("1.confirm\n");
178         printf("2.change item\n");
179         printf("3.cancel\n");
180         printf("Enter your choice: ");
181         scanf("%d", &choice);
182
183         if(choice == 1)
184         {
185             fp=fopen("project of resturent managment odered food.txt", "a+");
186             printf("Enter your name: ");
187             fflush(stdin);
188             gets(name);
189             printf("\nYour total bill is: %d\n", 150*plate);
190             printf("Would you like to pay in:\n");
191             printf("1. bCash\n");
192             printf("2. Rocket\n");
193             fflush(stdin);
194             scanf("%d", &method);
195
196             while(method>0||method<0)
197             {
198                 if(method == 1 || method == 2)
199                     break;
200                 else
201                 {
202                     printf("Enter the Correct Value: \n");
203                     printf("1. bCash\n");
204                     printf("2. Rocket\n");
205                     fflush(stdin);
206                     scanf("%d", &method);
207                 }
208             }
209
210             printf("Your phone number: ");
211             fflush(stdin);
212             scanf("%s", &num);
213
214             fprintf(fp, "%s ", name);
215             fprintf(fp, "Biriya ");
216             fprintf(fp, "%d ", plate);
217             fprintf(fp, "%s ", num);
218             fprintf(fp, "BDT: %d\n", 150*plate);
219             fclose(fp);
220
221             printf("\nThank you so much for your co-operation.\n");
222             printf("Please wait a bit. Your desire food will be served very soon :)");
223
224             printf("\nDo you want ti give any feedback? ('y' or 'n'): ");
225             fflush(stdin);
226             scanf("%c", &fb);
227         }
228     }
229 }
```

```
#include <stdio.h>

225     fflush(stdin);
226     scanf("%c", &fb);
227     if(fb == 'y' || fb == 'Y')
228     |     feedback();
229
230     system("pause");
231     system("cls");
232     main();
233
234     else if(choice==2)
235     |     foodOrder();
236
237     else if(choice==3)
238     |     main();
239
240     else
241     {
242     |     printf("Invalid input!!\n");
243     |     foodOrder();
244     |
245     }
246
247     else if(a==2)
248     {
249     |     printf("\nKorma\n");
250     |     printf("price: 120tk\n");
251     |     printf("cooking time: 30min(maximum)\n\n");
252     |     printf("How many plate of Korma you want to take: ");
253     |     fflush(stdin);
254     |     scanf("%d", &plate);
255     |     printf("1.confirm\n");
256     |     printf("2.change item\n");
257     |     printf("3.cancel\n");
258     |     printf("Enter your choice: ");
259     |     scanf("%d", &choice);
```

```
256     printf("3.cancel\n");
257     printf("Enter your choice: ");
258     scanf("%d", &choice);
259
260     if(choice == 1)
261     {
262     |     fp=fopen("project of resturent managment odered food.txt", "a+");
263     |     printf("Enter your name: ");
264     |     fflush(stdin);
265     |     gets(name);
266     |     printf("\nYour total bill is: %d\n\n", 120*plate);
267     |     printf("Would you like to pay in:\n");
268     |     printf("1. bCash\n");
269     |     printf("2. Rocket\n");
270     |     fflush(stdin);
271     |     scanf("%d", &method);
272     |     printf("Your phone number: ");
273     |     fflush(stdin);
274     |     scanf("%s", &num);
275
276     |     fprintf(fp, "%s ", name);
277     |     fprintf(fp, "Korma ");
278     |     fprintf(fp, "%d ", plate);
279     |     fprintf(fp, "%s ", num);
280     |     fprintf(fp, "BDT: %d\n", 120*plate);
281     |     fclose(fp);
282
283     |     printf("\nThank you so much for your co-operation.\n");
284     |     printf("Please wait a bit. Your desire food will be served very soon :) \n");
285
286     |     printf("\nDo you want ti give any feedback? ('y' or 'n'): ");
287     |     fflush(stdin);
288     |     scanf("%c", &fb);
289     |     if(fb == 'y' || fb == 'Y')
```

```
#include <stdio.h>
...
290
291     system("pause");
292     system("cls");
293     main();
294
295 }
296 else if(choice==2)
297     foodOrder();
298
299 else if(choice==3)
300     main();
301
302 else
303 {
304     printf("Invalid input!!\n");
305     foodOrder();
306 }
307
308 if(a==3)
309 {
310     printf("\nTeheri\n");
311     printf("price: 80tk\n");
312     printf("cooking time: 30min(maximum)\n\n");
313     printf("How many plate of Biryani you want to take: ");
314     fflush(stdin);
315     scanf("%d", &plate);
316     printf("1.confirm\n");
317     printf("2.change item\n");
318     printf("3.cancel\n");
319     printf("Enter your choice: ");
320     scanf("%d", &choice);
321
322     if(choice == 1)
323     {
324         fp=fopen("project of resturent managment odered food.txt", "a+");
325         printf("Enter your name: ");
326         fflush(stdin);
327         gets(name);
328         printf("\nYour total bill is: %d\n\n", 80*plate);
329         printf("Would you like to pay in:\n");
330         printf("1. bCash\n");
331         printf("2. Rocket\n");
332         fflush(stdin);
333         scanf("%d", &method);
334         printf("Your phone number: ");
335         fflush(stdin);
336         scanf("%s", &num);
337
338         fprintf(fp, "%s ", name);
339         fprintf(fp, "Tehri ");
340         fprintf(fp, "%d ", plate);
341         fprintf(fp, "%s ", num);
342         fprintf(fp, "BDT: %d\n", 80*plate);
343         fclose(fp);
344
345         printf("\nThank you so much for your co-operation.\n");
346         printf("Please wait a bit. Your desire food will be served very soon :)\n");
347
348         printf("\nDo you want ti give any feedback? ('y' or 'n'): ");
349         fflush(stdin);
350         scanf("%c", &fb);
351         if(fb == 'y' || fb == 'Y')
352             feedback();
353
354         system("pause");
355         system("cls");

```

```
#include <stdio.h>

...
354     system("pause");
355     system("cls");
356     main();
357 }
358 else if(choice==2)
359     foodOrder();
360
361 else if(choice==3)
362     main();
363
364 else
365 {
366     printf("Invalid input!!\n");
367     foodOrder();
368 }
369 }
370 if(a==4)
371 {
372     printf("\nKacchi\n");
373     printf("price: 160tk\n");
374     printf("cooking time: 30min(maximum)\n\n");
375     printf("How many plate of kacchi you want to take: ");
376     fflush(stdin);
377     scanf("%d", &plate);
378     printf("1.confirm\n");
379     printf("2.change item\n");
380     printf("3.cancel\n");
381     printf("Enter your choice: ");
382     scanf("%d", &choice);
383
384     if(choice == 1)
385     {
386         fp=fopen("project of resturent managment odered food.txt", "a+");
387         printf("Enter your name: ");
388     }
389     {
390         fp=fopen("project of resturent managment odered food.txt", "a+");
391         printf("Enter your name: ");
392         fflush(stdin);
393         gets(name);
394         printf("\nYour total bill is: %d\n\n", 160*plate);
395         printf("Would you like to pay in:\n");
396         printf("1. bCash\n");
397         printf("2. Rocket\n");
398         fflush(stdin);
399         scanf("%d", &method);
400         printf("Your phone number: ");
401         fflush(stdin);
402         scanf("%s", &num);
403
404         fprintf(fp, "%s ", name);
405         fprintf(fp, "Kacchi ");
406         fprintf(fp, "%d ", plate);
407         fprintf(fp, "%s ", num);
408         fprintf(fp, "BDT: %d\n", 160*plate);
409         fclose(fp);
410
411         printf("\nThank you so much for your co-operation.\n");
412         printf("Please wait a bit. Your desire food will be served very soon :)");
413
414         printf("\nDo you want to give any feedback? ('y' or 'n'): ");
415         fflush(stdin);
416         scanf("%c", &fb);
417         if(fb == 'y' || fb == 'Y')
418             feedback();
419
420         system("pause");
421         system("cls");
422         main();
423     }
424 }
```



```
#include <stdio.h>

417     system("cls");
418     main();
419 }
420 else if(choice==2)
421     foodOrder();
422
423     else if(choice==3)
424         main();
425
426     else
427     {
428         printf("Invalid input!!\n");
429         foodOrder();
430     }
431 }
432 else
433 {
434     printf("Invalid input\n");
435     printf("try again..\n");
436     main();
437 }
438 }
439
440 int feedback()
441 {
442     FILE *fp;
443     char message[100];
444     printf("Enter your Feedback: ");
445     fflush(stdin);
446     fp=fopen("project feedback.txt", "w");
447     gets(message);
448     fprintf(fp, "%s\n", message);
449     fclose(fp);
450 }
451
452 int adminPortal()
453 {
454     printf("\n\twelcome to admin portal.\n\n");
455     printf("1. Total Order\n");
456     printf("2. Message to chef\n");
457     printf("3. See Feedback\n");
458     printf("4. Logout\n\n");
459     printf("Enter your choice: ");
460     scanf("%d", &choice);
461
462     if(choice == 1)
463         totalOrder();
464 //     else if(choice == 2)
465 //         totalCustomer();
466     else if(choice == 2)
467     {
468         message();
469         adminPortal();
470     }
471     else if(choice == 3)
472     {
473         char feedback[200];
474         FILE *fp;
475         fp = fopen("project feedback.txt", "r");
476         fscanf(fp, "%s", feedback);
477         fclose(fp);
478         puts(feedback);
479         system("pause");
480         system("cls");
481         adminPortal();
482     }
```

```
#include <stdio.h>

480     system("cls");
481     adminPortal();
482 }
483 else if(choice==4)
484 {
485     printf("successfully logged out...\n");
486     system("pause");
487     system("cls");
488     main();
489 }
490 else
491 {
492     printf("wrong input\n");
493     adminPortal();
494 }
495 }
496
497 void message()
498 {
499     char messages[100];
500     FILE *fp;
501     fp=fopen("message.txt", "w");
502     printf("Enter your message: ");
503     fflush(stdin);
504     gets(messages);
505     fprintf(fp, "%s", messages);
506     fclose(fp);
507     adminPortal();
508 }
509
510
511 int chefPortal()
512 {
513     printf("\n\twelcome to chef portal.\n\n");
514     printf("1. Total Order\n");
515     printf("2. Message from admin\n");
516     printf("3. See Feedback\n");
517     printf("4. Logout\n\n");
518     printf("Enter your choice: ");
519     scanf("%d", &choice);
520
521     if(choice == 1)
522     {
523         FILE *fp;
524         fp=fopen("project of resturent management odered food.txt", "rb");
525         fseek(fp, 0, SEEK_END);
526         long fsize = ftell(fp);
527         fseek(fp, 0, SEEK_SET);
528
529         char *string = malloc(fsize + 1);
530         fread(string, fsize, 1, fp);
531         fclose(fp);
532
533         string[fsize] = 0;
534
535         printf("%s\n\n", string);
536
537         system("pause");
538         system("cls");
539         chefPortal();
540     }
541     // else if(choice == 2)
542     //     totalCustomer();
543     else if(choice == 2)
544     {
545         char messages[300];
546         FILE *fp;
547         fp=fopen("message.txt", "r");
```

The image displays two screenshots of a Visual Studio Code editor window, showing C code for a restaurant management system. The editor is titled "#include <stdio.h> • Untitled-1 - Visual Studio Code".

Top Screenshot (Lines 548-581):

```
548      fscanf(fp, "%[^\n]", messages);
549      fclose(fp);
550      //puts(feedback);
551      /*fseek(fp, 0, SEEK_END);
552      long fsize = ftell(fp);
553      fseek(fp, 0, SEEK_SET);
554
555      char *string = malloc(fsize + 1);
556      fread(string, fsize, 1, fp);
557      fclose(fp);
558
559      string[fsize] = 0;
560
561      */
562      puts(messages);
563      system("pause");
564      system("cls");
565      chefPortal();
566  }
567  else if(choice == 3)
568  {
569      char feedback[200];
570      FILE *fp;
571      fp = fopen("project feedback.txt", "r");
572      fscanf(fp, "%[^\n]", feedback);
573      fclose(fp);
574      puts(feedback);
575      system("pause");
576      system("cls");
577      chefPortal();
578  }
579  else if(choice==4)
580  {
581      printf("successfully logged out...\n");
```

Bottom Screenshot (Lines 580-613):

```
580  {
581      printf("successfully logged out...\n");
582      system("pause");
583      system("cls");
584      main();
585  }
586  else
587  {
588      printf("wrong input\n");
589      chefPortal();
590  }
591  }
592
593  void totalOrder()
594  {
595      FILE *fp;
596      fp=fopen("project of resturent managment odered food.txt", "rb");
597      fseek(fp, 0, SEEK_END);
598      long fsize = ftell(fp);
599      fseek(fp, 0, SEEK_SET); //same as rewind(f);
600
601      char *string = malloc(fsize + 1);
602      fread(string, fsize, 1, fp);
603      fclose(fp);
604
605      string[fsize] = 0;
606
607      printf("%s\n\n", string);
608
609      system("pause");
610      system("cls");
611      adminPortal();
612  }
613  }
```