**Canteen Automation Android Application**

Version 1.0 **Software Requirements Specification**

**Project Guide:** Mrs. Shweta Sharma

**Members:**

Parth Dadwal (1/16/FET/BCG/1/007)

Jatin Vashisht (1/16/FET/BCG/1/003)

Akshay Gupta (1/16/FET/BCG/1/018)

Sarthak Agarwal (1/16/FET/BCF/1/054)

College Name: Manav Rachna International Institute of Research and Studies.

Department: Faculty of Engineering and Technology.

# TABLE OF CONTENTS

Serial Title Page No.

No.

ABSTRACT 1

1. INTRODUCTION 2
   1. Purpose of Project 2
   2. Project and Product Overviews 3
   3. Team Architecture 4

* 1. Overall Description 5

1. PROBLEM STATEMENT 6
   1. Business Requirements 7
      1. Entry Point 7
      2. Selection of Product 7
      3. Reports 7
   2. System Requirement 7
      1. Usability 7
2. PROJECT UNDERSTANDING DOCUMENTS 8
   1. Purpose of Project 8
   2. Objective 8
3. REQUIREMENTS 9
   1. Specific Requirements 9
      1. External Interface Requirements 9
      2. Hardware Requirements 9

* + 1. Software Interface 9
  1. Functional Requirements 10
  2. Non-Functional Requirements 11

1. DESIGN TECHNIQUES 12
   1. HTML 12
   2. PHP 13
   3. MySQL 13
2. TIER ARCHITECTURE. 14
3. SOFTWARE PROCESS MODEL 16

* 1. Why not Evolutionary models? 16
  2. Why not Waterfall model? 16
  3. Why Incremental RAD model? 16
  4. Observation 17
  5. Determining project feasibility 17

1. DESIGN 18
   1. USE CASE DIAGRAM 18
   2. CLASS DIAGRAM 21
   3. SEQUENCE DIAGRAM 25
   4. ER DIAGRAM 29
   5. DATA FLOW DIAGRAM 31
2. DATABASE TABLES 34

# LIST OF FIGURES

Serial No. Figure Name Page No.

1. USE CASE DIAGRAM 18
   1. Use Case Diagram for Login 18
   2. Use Case Diagram for Registration 19
   3. Use Case Diagram for Order, Payment & 20

Update

1. CLASS DIAGRAM 21
   1. Class Diagram for Login 21
   2. Class Diagram for Registration 22
   3. Class Diagram for Order & Update 23
   4. Class Diagram for Payment 24

1. SEQUENCE DIAGRAM 25
   1. Sequence Diagram for Login 25
   2. Sequence Diagram for Registration 26
   3. Sequence Diagram for Payment 27
   4. Sequence Diagram for Placing Order 28
   5. Sequence Diagram for Updating Menu 28

1. ER DIAGRAM 29

1. DATA FLOW DIAGRAM 31
   1. DFD for Admin 32
   2. DFD for Canteen Owner 33
   3. DFD for Customer 33

# DATABASE TABLE

TABLE NAME PAGE NO.

1. Customer 34
2. Admin 35
3. Canteen Owner 35
4. Menu 36
5. Order 36
6. Payment 36

ABSTRACT

The Project “Canteen Automation System” enables the end users to register online, read and select the food from e-menu card and order food online by just selecting the food that the user want to have using android application. The results after selecting the food from the E-menu card will directly appear on the screen near the Chef who is going to cook the food for you. The system is the combination of Android as well as android Application. By using this application, the work of the waiter is reduced and we can also say that the work is nullified. The benefit of this is that if there is a rush in the Canteen then there will be chances that the waiters will be unavailable and the users can directly order the food to the chef online by using this application. The user will have a username and a password, by using which they can login into the system. This implies that the customer is the regular user of the Canteen.

The manual system involves paperwork in the form of maintaining various files and manuals. Maintaining critical information in the files and manuals is full of risk and a tedious process. Including a framework showing how to apply Internet technology progressively as skills and confidence grow, the project demonstrates the route from adapting materials to developing an online environment.

Nowadays people don’t have much time to spend in canteen by just there and waiting for the waiter to take their order. Many customers visit the canteen in their lunch break and recess so they have limited time to eat and return to their respective office and colleges. So this software helps them to save time and order food whenever they want without calling the waiter again and again.

1. INTRODUCTION:

In today’s age of fast canteen automation in the canteen, many canteens have chosen to focus on quick preparation and speedy delivery of orders. Until very recently, all of this delivery of orders were placed over the phone, but there are many disadvantages to this system, including the inconvenience of the customer needing to have a physical copy of the menu, lack of a visual confirmation that the order was placed correctly, and the necessity for the canteen to have an employee answering the phone and taking orders.

The main advantage of an online ordering system is that it greatly simplifies the ordering process for both the customer and the canteen. When the customer visits the ordering page of the app, they are presented with an interactive and up-to-date menu, complete with all available options and adjusting prices based on the selected options. After making a selection, the item is then added to their order, which the customer can review the details at any time before checking out. This provides instant visual confirmation of what was selected.

This system also greatly lightens the load on the canteen’s end, as the entire process of taking orders is automated. Once an order is placed on the application, it is entered into the database and then retrieved, in pretty much real-time, by a android based application on the canteen’s end. Within this application, all items in the order are displayed, along with their corresponding options and delivery details, in a concise and easy to read manner. This allows canteen employees to quickly go through the orders as they are placed and produce the necessary items with minimal delay and confusion.

## 1.1. PURPOSE OF PROJECT

Canteen Automation System is the system where customers order their food and receive food in the canteen without any delay as they can directly go and collect what they ordered without waiting for a turn or waiting time. This system aims to accelerate customer orders and customer order system used by employees to accept customer order.

The purpose of the system is to develop a simple Canteen Automation System and implement it, which later will be used for a web-based application.

The proposed “Canteen Automation System” is economically feasible because:

* The system requires very fewer time factors as compared to manual system
* The system will provide fast and efficient automated environment instead of slow and error prone manual system, thus reducing both time and manpower spent in running the system.
* The system will have GUI interface and very less user training is required to learn it.

## 1.2. PROJECT & PRODUCT OVERVIEWS

A complete canteen automation system makes canteen workers monitor overall day to day business analytics more accurately with an ease. It increases the presence of canteen fascinates core customers towards your food business leading to increased sales.

By bringing all necessities in one place canteen automation system benefits both the user as well as the canteen owner smartly. Ultimately all business operation will be mad easier and single handed with the required inbuilt features.

The working of canteen automation system is similar to an e-commerce website. Whenever your customers are busy with their work, instead of coming to your canteen they can just open your website, choose the menu they like and can simply order food.

As soon as order gets placed, you will be notified immediately, so that you can start order processing. One of the main advantages of canteen automation system is the order details are accurate when compared to the phone ordering system or manual system.

## 1.3. HOW ITS BENEFITS A CANTEEN

Canteen automation system saves the canteen’s time by avoiding the food orders over the phone that has to be done manually. By making the ordering process fully automated, it increases the cost effectiveness and productivity of the canteen with a less manpower. In addition, it keeps you one step ahead of your competitors who don’t serve online.

Canteen automation system even helps a casual dining canteen and provides an additional revenue source. It allows a canteen owner to easily update the online menu, food items etc and helps to stay in touch with the customers by offering discounts and targeted promotions. With a simplified management process, online ordering system makes it easy to handle multiple canteens from centralized application

## 1.4. HOW IT BENEFITS A STUDENT

Now a day’s recess time for students is not enough to order, wait, eat, and have their time. This canteen management app saves students time undoubtedly. Allow them to order and queue food online by creating a flexible ordering platform and serve them in time. The student can select the food items from the online menu on the canteen app and can order and put their order in a queue accordingly as per their interest. In fact, they can place order the food orders within their budget by adding or removing the food items as per the cost variations

This canteen automation app is useful for the canteen as well customer because canteen automation system saves the time of the canteen's workers as well of the customers and as saves the manpower.

## 1.5 TEAM ARCHITECTURE

There were four members in our project. The complete project is divided into two teams.

TEAM 1:

REQUIREMENT GATHERING & PLANNING

Planning identifies all deliverable services, describes the facilities, and defines the working to provide details about lab details to the faculties. This planning problem starts with a specification of user demand that is to be met by the production plan. For this context, the facilities provided to the user faculties are the major concern to be concentrated. This includes the easiest and efficient way to make them available an unoccupied lab. Consequently, all the fields must be kept while developing. Generally, a planning problem exists because there are limited production resources that cannot be stored from period to period. Choices must be made as to which resources to include and how to model their capacity and behavior, and their costs. Also, there may be uncertainty associated with the production function, such as uncertain yields or lead times. One might only include the most critical or limiting resource in the planning problem. Here comes the role of proper requirement gathering. If the requirements of the user are clear then it is quite easier for the developer to fulfill his all needs. As he can look into all the required resources, and with proper planning and cost estimation, he can achieve his software.

Thus, a proper planning and requirement gathering leads to an efficient software system.

TEAM 2:

DESIGNING AND BACKEND

Designing is the most important and the most efficient function while software development. Without a proper design, it is very difficult to develop an appropriate software that fulfills nearly all user demands. Therefore, managing designing part in an organization is a critical activity. A designer needs to ensure that the design created by him can be easily understood by the all the members of developing a team. A proper design will allow the coder to implement the system development planning properly. Similarly, the backend part that includes the database management plays a key role in any of the systems. Thus, the team working on this field must know proper handling and management of database and its tools.

## 1.6 OVERALL DESCRIPTION

Time Saving: Our canteen automation system is developed with a primary aim of 'Saving Time. The customer can order the food and it is also efficient for canteen workers because this system takes lesser time as compared to phone based or manually based system.

No Complication: Major complication part for canteen automation system is adding a product or managing products section. In our canteen automation system, no complicated part is involved in managing sections; we have taken immense care in this section and nullified all complications which make this system accurate and unique.

Cost Effective: It's cheaper. You don't have to purchase multiple copies of software to install on multiple computers. Multiple copies often require you to pay multiple licensing fees, but since you aren't actually purchasing any software with an online system, that's not a concern.

Security: Online systems are just as secure. Most online canteen automation system programs allow you to create multiple user accounts with various levels of access. Your data is stored on secure, protected servers that feature firewalls and other online security programs.

## 1.7 PRODUCT PERSPECTIVE

The system will be developed using JAVA, C++, XML and other android development tools.

Product Functions

The product developed will provide an easy access to the user to manage and alter the canteen automation according to the need.

General Constraints

* The cost constraint includes ordering additional hardware to run the new system
* No current web-site
* Security must be upgraded for on-line searching.

Assumptions

* The system will support all browsers.
* The processor must be at least Dual cored and can be any other latest ones etc.

## 2. PROBLEM STATEMENT

The Challenges encountered by the manual system in canteens is efficiency and customer satisfaction. The experience of ordering in most fast food canteens is not pleasant for customers. Customers have to make long queues before placing the order and when the order is placed, they have to wait near the counter until the order is prepared. The another problem is efficiency that food canteen should maintain in their standard operations and keep with the quality of their product and services no matter how much crowd is present in canteen but they have to maintain efficiency as well as quality of product .however, we think that there are some issue concerning the traditional way to order food in canteens.

The major issues are as follows

* Verbal communication between cashier and customer or we can say telephonic communication: The verbal communication between two parties for placing an order and the information about bill should also result in error means error also occurs in understanding what the person wants to say and especially in busy hours in canteens. When the place is very crowded and noisy, miscommunications are common. The problem is even Worsen if the cashier and/or the customers do not speak the native language.

* Food customization: The ability to customize food has been a crucial business strategy in the food industry since McDonald’s slogan “I’m Lovin’ it” for the campaign. Food customization allows some flexibility for customers who have a special request (less ice for soda, no pickles etc.) and is especially necessary for customers who are allergic to some ingredients .currently, the cashier has to memorize these requests since the existing system does not support any means of recording such request. The result is that the request are forgotten or miscommunication to the cook.

* Menu display: Today’s competition between food canteens motivates each canteen to launch new items on their menus on a more frequent basis. However, the menu in more canteens usually attached to a wall behind the counter and the customer are not aware of that new item because the menu is not up-to-date. Furthermore, not all item on the menu list has a graphical illustration to help customers to take Decision easily what they want. The print is sometimes small that it’s not visible to the customers.

2.1 BUSINESS REQUIREMENTS:

2.1.1 ENTRY POINT

The system is required to have two entry points:

1. Canteen Management Direct: A Canteen Management Direct entry point is where the manager can manage the canteen and service the customer requests.
2. Customer Linking: A Customer Linking entry point is where a prospective customer can view the details of products and place their order.

2.1.2 SELECTION OF PRODUCT

Based upon the customer’s request the feasibility of supplying product is evaluated. The customer request is fulfilled according to the order placed.

2.1.3 REPORTS

* Annual reports
* Monthly reports
* Sales reports
* Customer trends

2.2 SYSTEM REQUIREMENT

Performance and Scalability: The system is required to scale to support transaction volume. Web Pages should be light and render fast.

2.2.1 USABILITY

The web forms should be self-explanatory and usable. We do not want prospective clients dropping of the website because they cannot understand the forms and find them cumbersome.

## 3. PROJECT UNDERSTANDING DOCUMENTS

3.1 PURPOSE OF PROJECT

Canteen Management app is the system where customers order their food and receive food in the canteen without any delay as they can directly go and collect what they ordered without waiting for a turn or waiting time. This system aims to accelerate customer orders and customer order system used by employees to accept customer order.

The purpose of the system is to develop a simple Canteen automation Software and implement it, which later will be used for a web-based application.

The proposed “Canteen Management app” is economically feasible because

* The system requires very fewer time factors as compared to manual system
* The system will provide fast and efficient automated environment instead of slow and error prone manual system, thus reducing both time and manpower spent in running the system.
* The system will have GUI interface and very less user training is required to learn it.

3.2 OBJECTIVE

Our objective is to make a platform independent application to maintain a database of all orders ordered from various sources and all the different services required by each of them. Established canteen automation practices should provide the needed connectivity and accountability between those two operational units, and when managed properly, enhances the effectiveness of both operations.

* Registration
* Order
* Token generation
* Payment
* Update

The above are the modules of canteen automation system.

## 4. REQUIREMENTS

### 4.1 SPECIFIC REQUIREMENTS

#### 4.1.1 EXTERNAL INTERFACE REQUIREMENTS

The external system is to assume full responsibility for storage functions as well as warehouse management and warehouse control for an entire warehouse. The interfaces in this section are specified by documenting: the name and description of each scheme, source or input, destination or output, ranges, accuracy and tolerances, units of measure, timing, display formats, and organization, and data formats.

The user interface required to be developed for the system should be user-friendly and attractive. The interface between the user and the system will be WIMP (Windows, Icons, Menu, Pointers) keeping in mind that the system is to be run through a web browser. All operations will be off point and click nature with all navigations performed through windows of the system specifically buttons and menus:

Buttons: The button is activated when the user will click on the left click of the mouse within the bounds of the button. And thus, the action associated with it will be carried out.

Menu: All the operations will be arranged.

#### 4.1.2 HARDWARE REQUIREMENT

Here's what you Need to Use the canteen Management App for online records:

* Standard Internal memory on any device
* 2GB RAM
* Any standard mobile device Processor
* Input Devices: Touch pad
* Output Devices: Mobile device screen

#### 4.1.3 SOFTWARE REQUIREMENT

* Operating System: Android Oreo higher

4.2 Functional Requirements:

Users of the canteen automation system, namely canteen customers, must be provided the following functionality:

* Create an account.
* Manage their account.
* Log into the system.
* Navigate the canteen’s menu.
* Select an item from the menu.
* Add an item to their current order.
* Review their current order.
* Generate token for the order on user’s registered mobile number.
* Remove an item/remove all items from their current order.

After the approval of project for further development including funding:

* Provide payment details.
* Place an order.
* Receive confirmation in the form of an order number.

#### Menu Management System

The menu management system will be available only to canteen employees and will, as the name suggests, allow them to manage the menu that is displayed to users of the web ordering system. The functions afforded by the menu management system provide the user with the ability to, using a graphical interface:

* Add a new/update/delete food category to/from the menu.
* Add a new/update/delete food item to/from the menu.
* Add a new/update/delete option for a given food item.
* Update default options for a given food item.
* Update additional information (description, photo, etc.) for a given food item.

#### Order Retrieval System

Of the three components, the order retrieval system is functionally the simplest. Like the menu management system, it is designed to be used only by canteen employees, and provides the following functions:

* Retrieve new orders from the database.
* Display the orders in an easily readable, graphical way.
* Mark an order as having been processed and remove it from the list of active orders.
* Generate token for the retrieved order.

User Interface Specifications:

Each of the system components will have their own unique interface. These are described below.

#### Application Ordering System

Users of the Application ordering system will interact with the application through a series of simple forms. Each category of food has its own form associated with it which presents a drop-down menu for choosing which specific item from the category should be added to the order.

Adding an item to the order is accomplished by a single button touch. Users select which category of food they would like to order, and therefore which form should be displayed, by navigating a menu bar, an approach which should be familiar to most users.

Entering delivery and payment deals is done in a similar manner. The user is presented with a form and must complete the required fields, which include both drops down and text boxes, before checking out and receiving a confirmation number.

4.3 Non-functional Requirements:

Performance Criteria:

* Time:

The elapsed time between the submission of order process between the customer and cashier in a canteen should be as minimum as possible.

 User-friendly:

Our canteen management app should be more user friendly. The user interface should be kept simple and uncluttered. Since the different type of people will interact with this process so our project should be very easy to them to understand.

* Flexibility:

Our project should be so flexible that whenever we want to make changes in it very easily it can be done.

* Extensibility:

It should be able to accommodate the variations like:

* 1. The different order should be handled easily.
  2. It should be an option for cash on delivery, pay through card between customer and canteen.
* Portable:

Our project should be portable on any platform and available on websites easily and at a faster speed than others.

* Reusable:

All the customer web pages that are being used for customer information should be easily get processed so that many customers can interact with us very easily and very fast without any information destroy.

## 5. DESIGN TECHNIQUES

The design of the site has been done using the following technologies:-

* JAVA
* Json
* C++
* XML
* Other necessary languages for android development.

5.1 JAVA

Java is a programming language and computing platform first released by Sun Microsystems in 1995. There are lots of applications and websites that will not work unless you have Java installed, and more are created every day. Java is fast, secure, and reliable. From laptops to datacenters, game consoles to scientific supercomputers, cell phones to the Internet, Java is everywhere! As a result, the source code behind today’s Web pages is often a hideous concoction of tags and scripting.

5.2 XML

**Extensible Markup Language** (**XML**) is a [markup language](https://en.wikipedia.org/wiki/Markup_language) that defines a set of rules for encoding [documents](https://en.wikipedia.org/wiki/Electronic_document) in a [format](https://en.wikipedia.org/wiki/File_format) that is both [human-readable](https://en.wikipedia.org/wiki/Human-readable_medium) and [machine-readable](https://en.wikipedia.org/wiki/Machine-readable_data). The [World Wide Web Consortium](https://en.wikipedia.org/wiki/World_Wide_Web_Consortium)'s XML 1.0 Specification of 1998 and several other related specifications—all of them free [open standards](https://en.wikipedia.org/wiki/Open_standard)—define XML.

The design goals of XML emphasize simplicity, generality, and usability across the [Internet](https://en.wikipedia.org/wiki/Internet) It is a textual data format with strong support via [Unicode](https://en.wikipedia.org/wiki/Unicode) for different [human languages](https://en.wikipedia.org/wiki/Language). Although the design of XML focuses on documents, the language is widely used for the representation of arbitrary [data structures](https://en.wikipedia.org/wiki/Data_structure) such as those used in [web services](https://en.wikipedia.org/wiki/Web_service).

Several [schema systems](https://en.wikipedia.org/wiki/XML_schema) exist to aid in the definition of XML-based languages, while programmers have developed many [application programming interfaces](https://en.wikipedia.org/wiki/Application_programming_interface) (APIs) to aid the processing of XML data.

5.3 C++

**C++** is a [high-level](https://en.wikipedia.org/wiki/High-level_programming_language), [general-purpose programming language](https://en.wikipedia.org/wiki/General-purpose_programming_language) created by [Bjarne Stroustrup](https://en.wikipedia.org/wiki/Bjarne_Stroustrup)  as an extension of the [C programming language](https://en.wikipedia.org/wiki/C_(programming_language)), or "C with [Classes](https://en.wikipedia.org/wiki/Class_(programming))". The language has expanded significantly over time, and modern C++ has [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming), [generic](https://en.wikipedia.org/wiki/Generic_programming), and [functional](https://en.wikipedia.org/wiki/Functional_programming) features in addition to facilities for [low-level](https://en.wikipedia.org/wiki/Low-level_programming_language) [memory](https://en.wikipedia.org/wiki/Memory_(computing)) manipulation. It is almost always implemented as a [compiled language](https://en.wikipedia.org/wiki/Compiled_language), and many vendors provide [C++ compilers](https://en.wikipedia.org/wiki/List_of_compilers#C.2B.2B_compilers), including the [Free Software Foundation](https://en.wikipedia.org/wiki/Free_Software_Foundation), [LLVM](https://en.wikipedia.org/wiki/LLVM), [Microsoft](https://en.wikipedia.org/wiki/Microsoft), [Intel](https://en.wikipedia.org/wiki/Intel), [Oracle](https://en.wikipedia.org/wiki/Oracle_Developer_Studio), and [IBM](https://en.wikipedia.org/wiki/IBM), so it is available on many platforms.

C++ was designed with a bias toward [system programming](https://en.wikipedia.org/wiki/System_programming) and [embedded](https://en.wikipedia.org/wiki/Embedded_software), resource-constrained software and large systems, with [performance](https://en.wikipedia.org/wiki/Performance_(software)), efficiency, and flexibility of use as its design highlights. C++ has also been found useful in many other contexts, with key strengths being software infrastructure and resource-constrained applications,  including [desktop applications](https://en.wikipedia.org/wiki/Application_software), [video games](https://en.wikipedia.org/wiki/Video_games), [servers](https://en.wikipedia.org/wiki/Server_(computing)) (e.g. [e-commerce](https://en.wikipedia.org/wiki/E-commerce), [Web search](https://en.wikipedia.org/wiki/Web_search_engine), or [SQL](https://en.wikipedia.org/wiki/SQL) servers), and performance-critical applications (e.g. [telephone switches](https://en.wikipedia.org/wiki/Telephone_switches) or [space probes](https://en.wikipedia.org/wiki/Space_probes)).

C++ is standardized by the [International Organization for Standardization](https://en.wikipedia.org/wiki/International_Organization_for_Standardization) (ISO), with the latest standard version ratified and published by ISO in December 2017 (informally known as [C++17](https://en.wikipedia.org/wiki/C%2B%2B17)). The C++ programming language was initially standardized in 1998 as *ISO/IEC 14882:1998*, which was then amended by the [C++03](https://en.wikipedia.org/wiki/C%2B%2B03), [C++11](https://en.wikipedia.org/wiki/C%2B%2B11) and [C++14](https://en.wikipedia.org/wiki/C%2B%2B14) standards. The current C++17 standard supersedes these with new features and an enlarged [standard library](https://en.wikipedia.org/wiki/C%2B%2B#Standard_library). Before the initial standardization in 1998, C++ was developed by Danish computer scientist [Bjarne Stroustrup](https://en.wikipedia.org/wiki/Bjarne_Stroustrup) at [Bell Labs](https://en.wikipedia.org/wiki/Bell_Labs) since 1979 as an extension of the [C language](https://en.wikipedia.org/wiki/C_(programming_language)); he wanted an efficient and flexible language similar to C that also provided [high-level features](https://en.wikipedia.org/wiki/High-level_programming_language) for program organization.  [C++20](https://en.wikipedia.org/wiki/C%2B%2B20) is the next planned standard, keeping with the current trend of a new version every three years.

6. TIER ARCHITECTURE.

The various classes as obtained from the business class diagram is categorized as follows-

Form of the project Class Class

Application or

Presentation

Layer

Business Layer

or Logical Layer

Data Layer or

Data Access

Layer

The 3-tier architecture consists of three layers:

Presentation Layer - The website or windows forms application is called the presentation layer. The presentation layer is the most important layer simply because it’s the one that everyone sees and uses. Even with a well-structured business and data layer, if the presentation layer is designed poorly, this gives the users a poor view of the system. The presentation layer is the form where we design using the controls like textbox, labels, command buttons etc.

Business Layer - Though a website could talk to the data access layer directly, it usually goes through another layer called the business layer.

This layer is a class which we use to write the function which works as a mediator to transfer the data from Application or presentation layer data layer. In the three-tier architecture, we never let the data access layer to interact with the presentation layer.

This layer is also a class where we declare the variable corresponding to the fields of the database which can be required for the application and make the properties so that we can get or set the data using these properties into the variables. These properties are public so that we can access its values.

One of the best reasons for reusing logic is that applications that start off small usually grow in functionality. For instance, a company begins to develop a website, and as they realize their business needs, they later decide to add a smart client application and windows service to supplement the website. The business layer helps move logic to a central layer for “maximum

reusability.”

Business layer have been presented having two roles

* client application
* server component

An example of Business Layer-The Business layer has functions of which takes the parameters from the example given in the presentation layer .As the user inputs the data values, corresponding functions are called in the business layer which is further passed on through the data layer where corresponding procedures are called and the data is been updated.

The business layer is the class where we write the functions which get the data from the application layer and passes through the data access layer.

Data layer - The key component to most applications is the data. The data has to be served to the presentation layer somehow. The data layer is a separate component whose sole purpose is to serve up the data from the database and return it to the caller. This layer is also a class which we use to get or set the data to the database back and forth. This layer only interacts with the database. We write the database queries or use stored procedures to access the data from the database or to perform any operation to the database.

ADVANTAGE OF 3 TIER ARCHITECTURE

* Client-Server architecture is 2-Tier architecture because the client does not distinguish between Presentation layer and business layer.
* The increasing demands on GUI controls caused difficulty to manage the mixture of source code from GUI and Business Logic.
* Further, Client Server Architecture does not support enough the Change Management. Let suppose that the government increases the Entertainment tax rate from 4% to 8 %, then in the Client-Server case, we have to send an update to each client and they must update synchronously on a specific time otherwise, we may store invalid or wrong information.
* The Client-Server Architecture is also a burden to network traffic and resources. Let us assume that about five hundred clients are working on a data server then we will have five hundred ODBC connections and several ruffian record sets, which must be transported from the server to the clients.
* This categorization of the application makes the function more reusable easily and it becomes too easy to find the functions which have been written previously. If the programmer wants to make a further update in the application then he easily can understand the previously written code and can update easily.

DISADVANTAGES

* Increase complexity /effort
* More difficult to build 3 tier architecture rather than a 2 tier.
* Points of communication are doubled
* Maintenance tools are currently inadequate for maintaining server libraries.

## 7. SOFTWARE PROCESS MODEL

7.1 Why not Evolutionary models?

These models are best suited where requirements are fuzzy. These models are best suited for the systems where requirements keep on changing. But for our system requirements are crystal clear so it is not feasible to adopt any of the evolutionary models.

7.2 Why not Waterfall model?

Waterfall model can be adopted because in our case because requirements are known in advance but there are some limitations of waterfall model due to which it is not feasible to adopt:

* No parallelism of work.
* Time consuming
  1. Why Incremental RAD model?

Incremental model is advisable where requirements are clear and the development time is less. The striking feature of the incremental model is that each module can be completed and released as and when the requirement arises because of lack of time.

As in our system, many of the modules are not inter-related so can be released in isolation. The user can thus get a feel of these modules and give his feedback which can be utilized for making the software more user-friendly and in line with the user requirements.

Not only that the deadline set for this project is 3 months and we need a high adaptation model and again will be concentrating on parallelism because our team will be working on the different module at the same time. Moreover, we will be using latest tools such as Visio, Project Manager as a result of which we can work much faster. So looking into all these requirements we find Incremental RAD model is best suited for our system because it enables the development team to create a fully functional system within a very short period of time.

* 1. Observation

We have observed that our system that is Canteen Automation System would be of immense help to the client as currently everything is done manually, which results in a lot of time consumption, is error prone and also increases economic burden in the form of payment not paid by the customers. Moreover, such a manual system of managing canteen is quite unstructured. Our system would be efficient, accurate and easy to use.

* 1. Determining Project Feasibility

The feasibility study is not a full-blown systems study. Rather, the feasibility study is used to gather broad data to make a decision on whether to proceed with system study.

System project feasibility is assessed in three principal ways:

* Economically
* Technically
* Operationally

Economic Feasibility

The organization has evaluated the cost of software and hardware required for the system including the storage of data. The benefits expected from the system are studied to assess the reduced cost due to the new system.

Technical Feasibility:

Organization has shown a willingness to purchase all hardware and software tools which we recommend to successfully implement the system. Hence technically there are no limitations for the development of the system. Thus the project is technically feasible.

Operational Feasibility:

Operational feasibility is dependent on the humans who will be using the software once it’s ready and installed for use. The software will have a user-friendly interface which will be much convenient as compared to the current manual procedure. Thus, the project is operationally feasible

1. DESIGN

* 1. Use Case Diagram

A use case is a description of how end-users will use a software code. It describes a task or a series of tasks that users will accomplish using the software and includes the responses of the software to user actions.

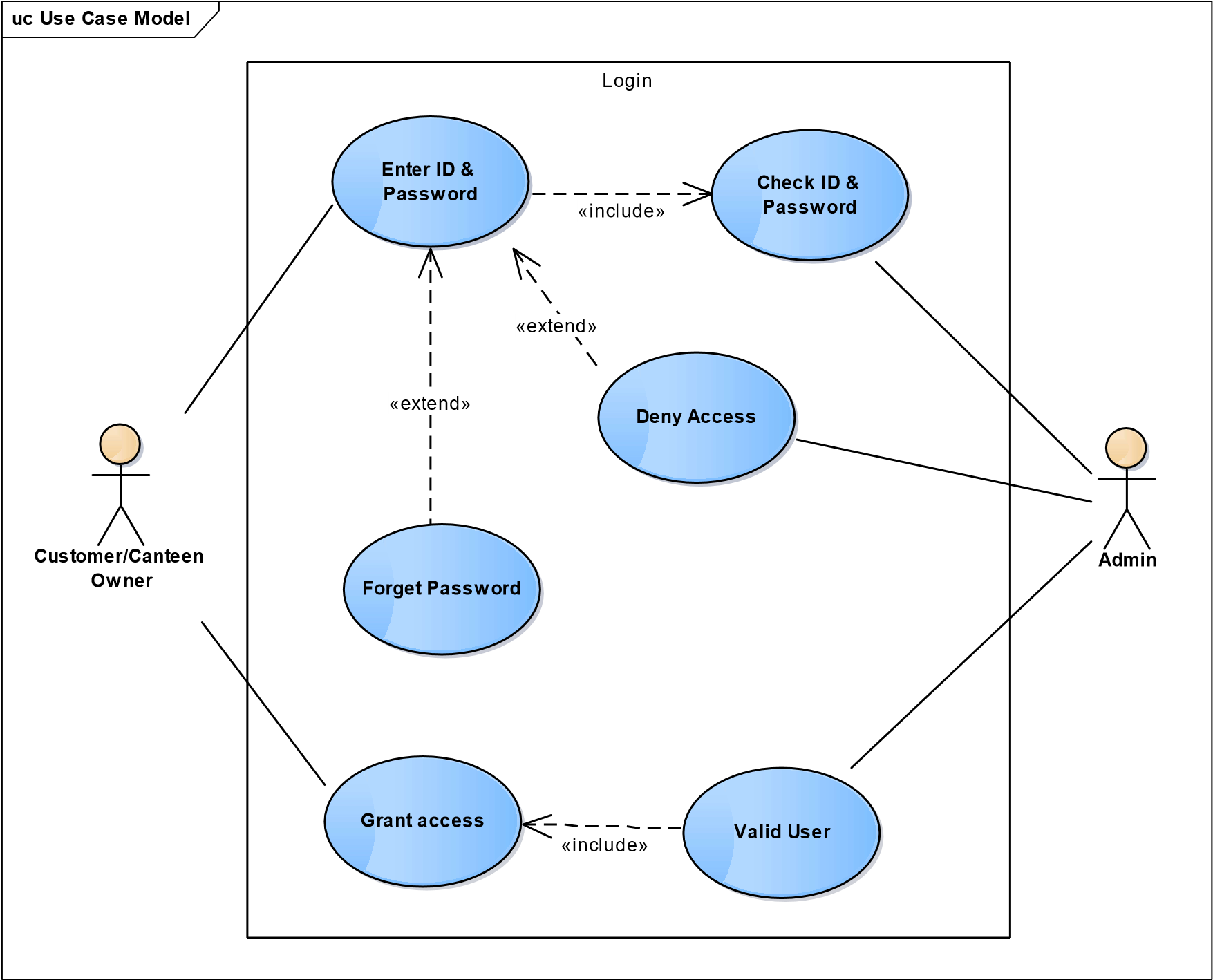


Fig 1.1 Use case for Login Page

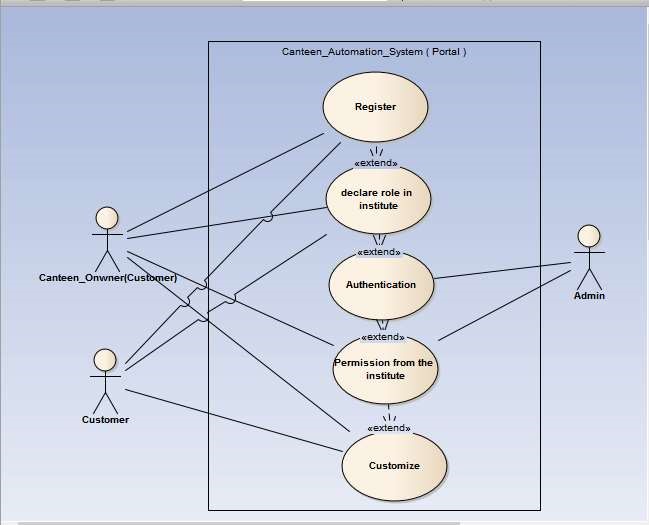


Fig. 1.2 Use Case Diagram for Registration

uc Use Case Model

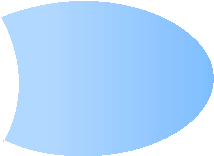
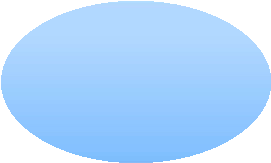
Order & Payment



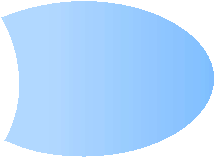
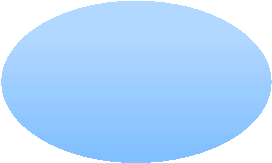
Canteen Owner



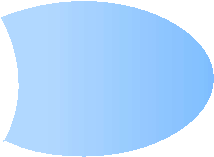
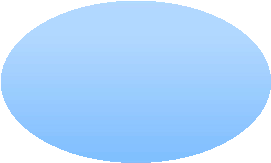
Customer



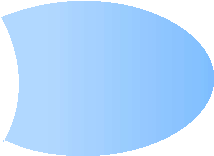
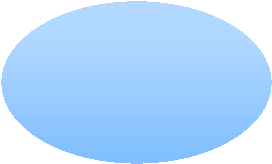
Update Menu



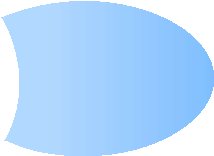
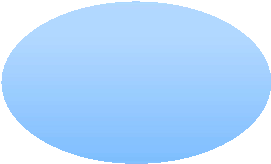
Monitor Canteen



Place Order



Read Order



Make Payment

«include»

«extend»

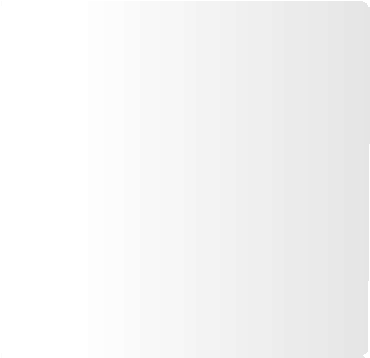
«extend»

Fig. 1.3 Use case for Order, Payment & Update

* 1. Class Diagram

In the Unified Modeling Language (UML), a class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, and the relationships between the classes.

class Class Model



Login UI

User id: char

-

-

Password: char

+

Get\_id\_Password(): boolean

Authenticate(): boolean

+

Enter\_id\_Password(): boolean

+



Customer/Canteen Owner

User id : char

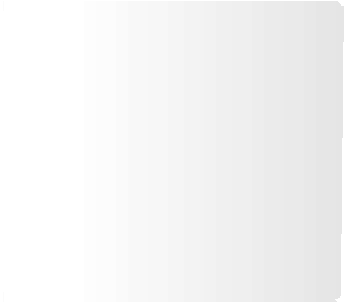
-

-

Password : char

+

Enter\_id\_password(): boolean



Administrator

-

Admin\_id: char

-

Password: char

+

Authenticate(): boolean

+

Get\_id\_password(): boolean

Fig. 2.1 Class diagram of Login Page

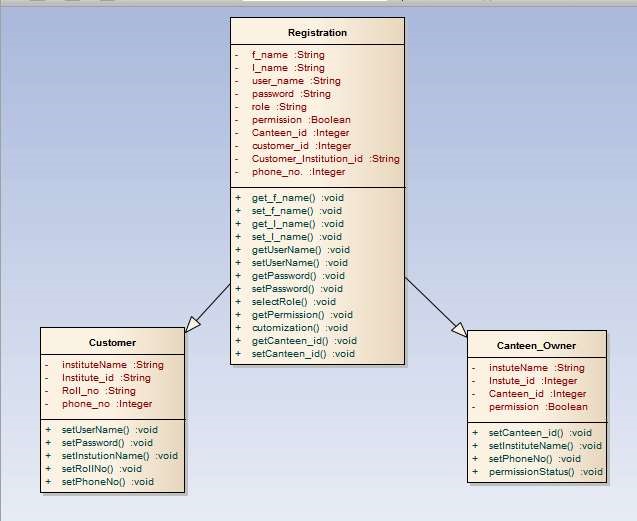
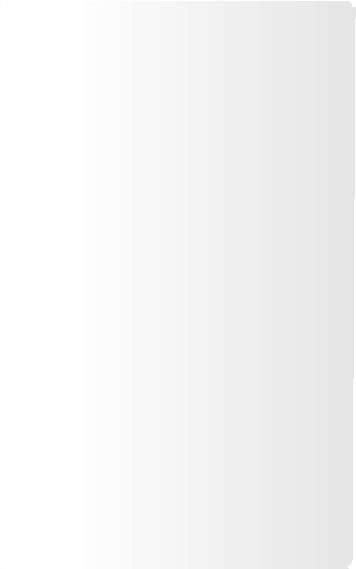


Fig. 2.2 Class Diagram for Registration

class Class Model



Order UI

Food items: char

-

cold drinks: char

-

order\_id: int

-

Tea and Coffee: char

-

Cust\_name: char

-

choose\_order(): char

+

make\_order(): char

+

order\_again(): char

+

confirm\_order(): char

+

bill(): char

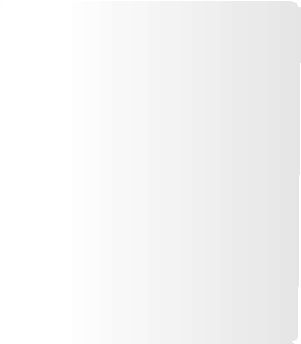
+

+

update\_menu(): char

+

update to\_do\_list(): char



Customer

-

Food items: char

-

cold drinks: char

Tea and Coffee: char

-

choose\_order(): char

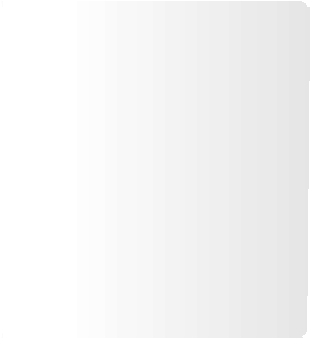
+

+

make\_order(): char

+

order\_again(): char



Canteen manager

-

Cust\_name: char

-

order\_id: int

confirm\_order(): char

+

+

bill(): char

+

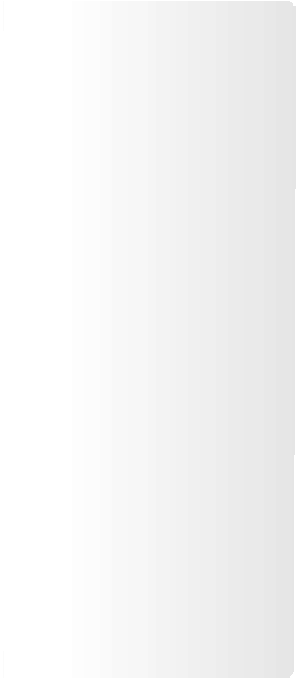
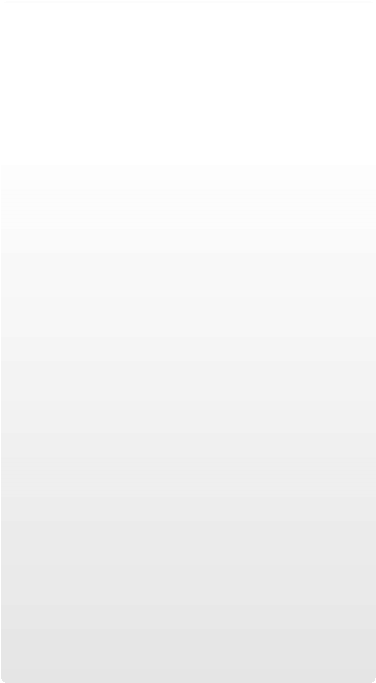
update menu(): char

+

update to\_do\_list(): char

Fig. 2.3 Class diagram of Order and Update

class Class Model



Payment UI

-

cash: int

card name: char

-

card type: char

-

-

payment id: char

-

Payment\_id: char

-

cust\_name: char

-

order\_id: int

select\_mode(): char

+

cash\_mode(): char

+

pay\_cash(): float

+

card\_mode(): char

+

+

card\_payment(): int

+

fee\_mode(): char

+

card\_error(): boolean

print\_receipt(): char

+

+

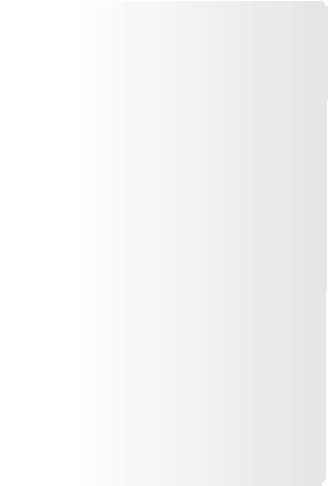
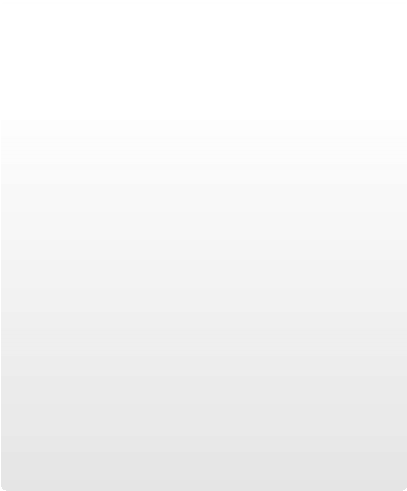
changes(): int

select\_fee\_mode(): char

+

+

proceed\_payment(): float



Customer

-

cash: int

-

card name: char

-

card type: char

-

payment id: char

select\_mode(): char

+

+

cash\_mode(): char

+

pay\_cash(): float

+

card\_mode(): char

+

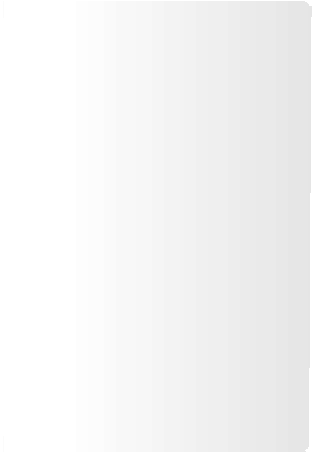
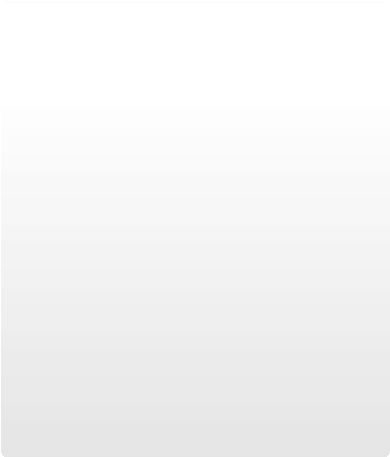
card\_payment(): int

+

fee\_mode(): char

+

card\_error(): boolean



Canteen manager

-

Payment\_id: char

-

cust\_name: char

order\_id: int

-

+

print\_receipt(): char

+

changes(): int

+

select\_fee\_mode(): char

+

proceed\_payment(): float

Fig. 2.4 Class diagram for Payment

* 1. Sequence Diagram:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

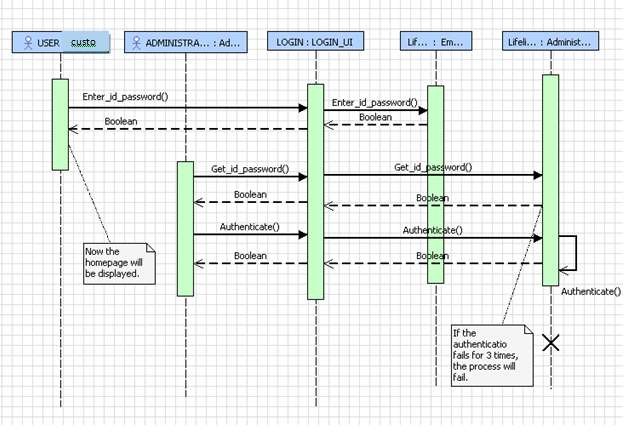


Fig. 3.1 Sequence of Login

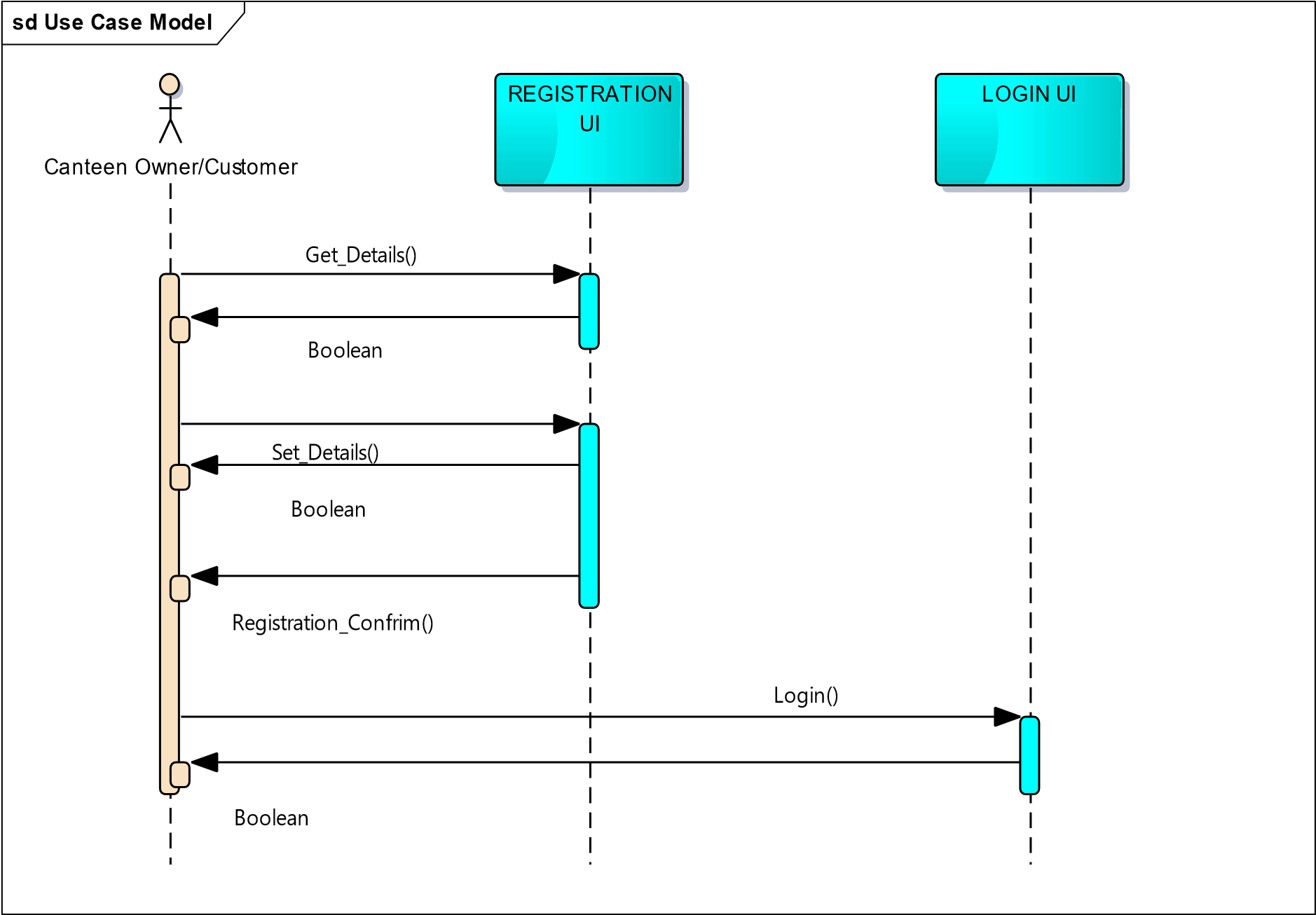


Fig. 3.2 Sequence Diagram for Registration

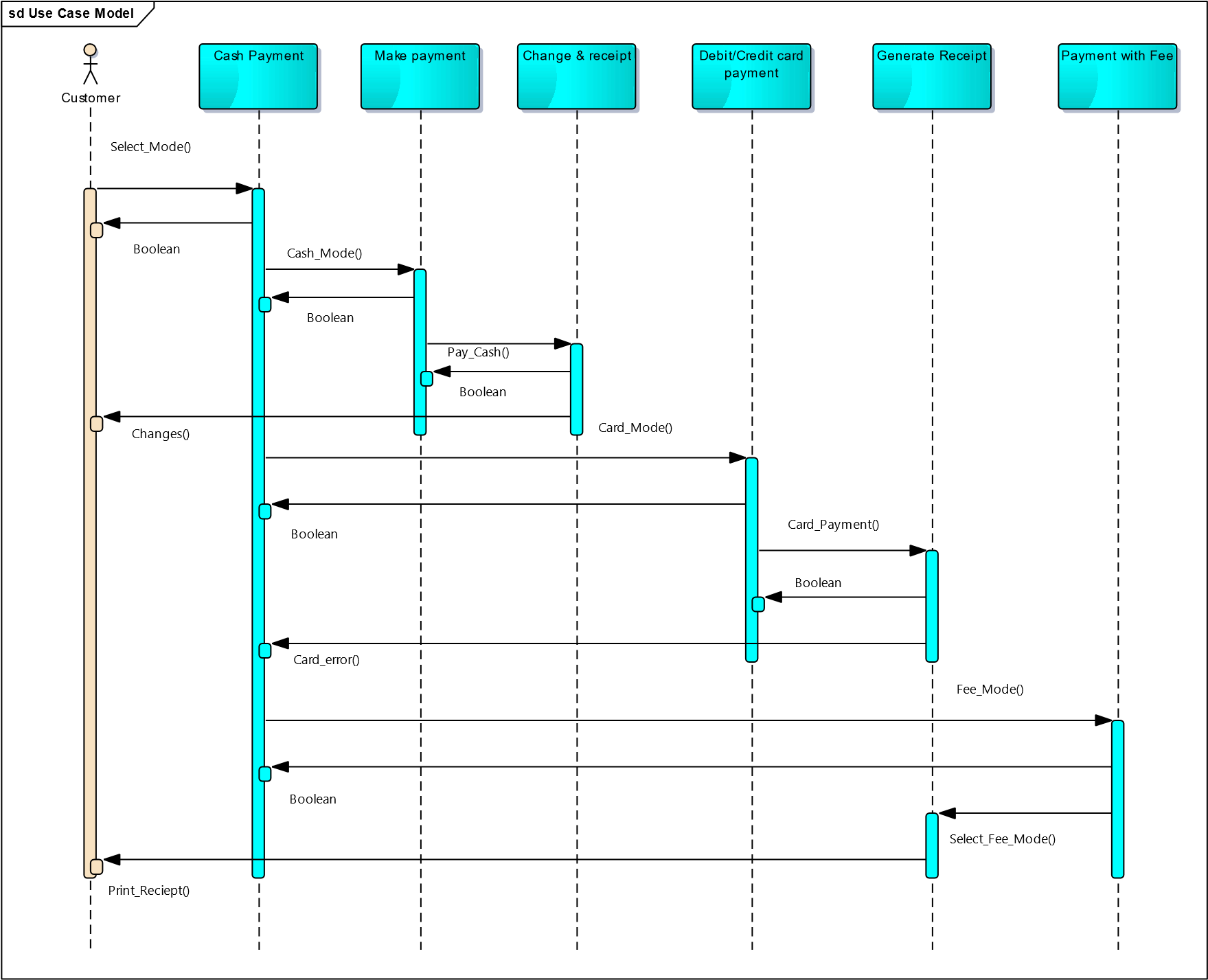


Fig. 3.3 Sequence diagram for payment

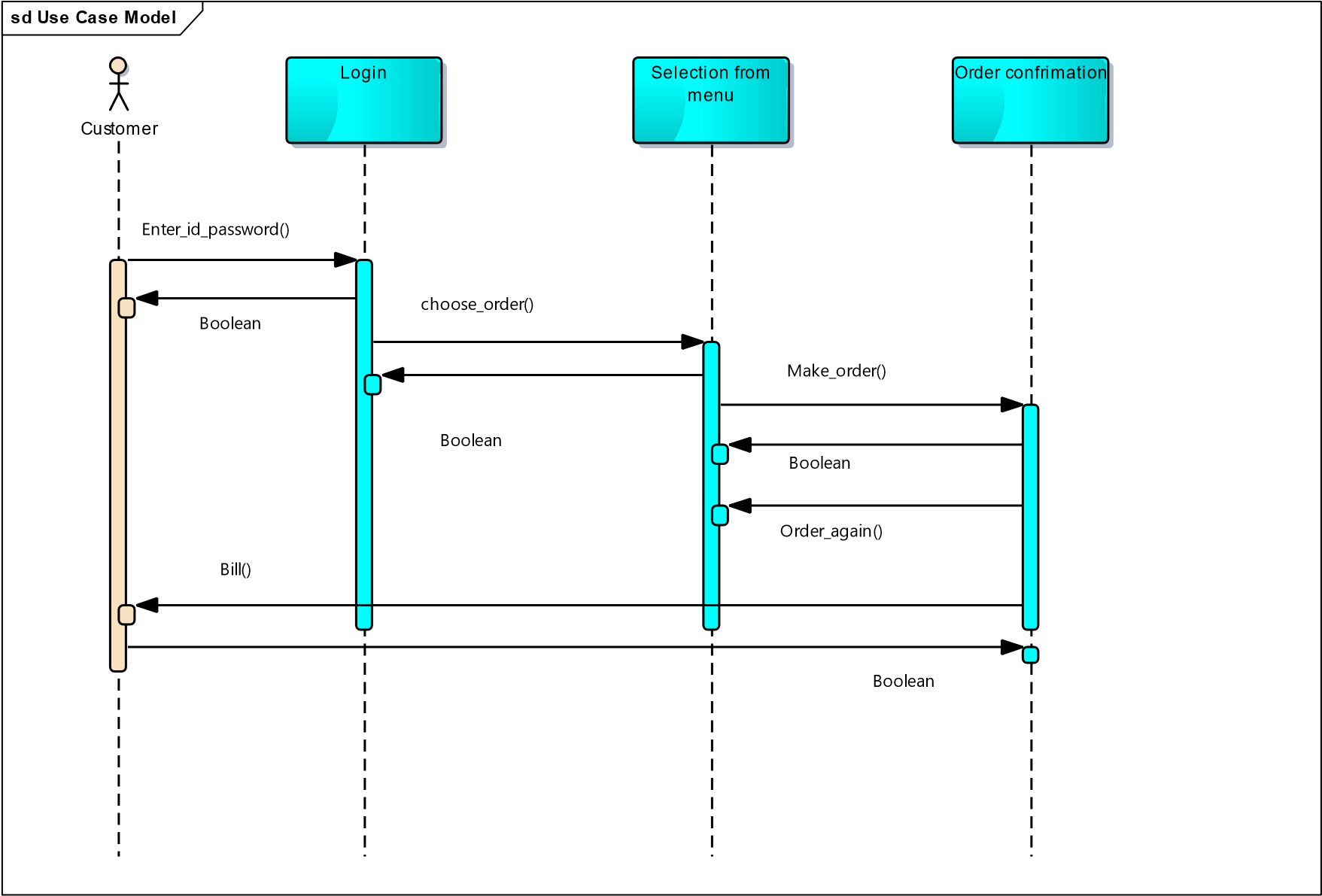
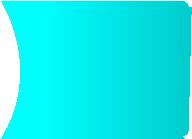


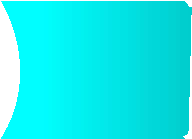
Fig. 3.4 Sequence diagram of placing order

sd Use Case Model

Canteen Owner



Menu



To Do List

Update\_Menu()

Boolean

Update\_todolist()

Boolean

Fig. 3.5 Sequence Diagram for Updating Menu

### 8.4 ER DIAGRAM

An entity-relationship (ER) diagram is a specialized graphic that illustrates the interrelationships between entities in a database. ER diagrams often use symbols to represent three different types of information. Boxes are commonly used to represent entities. Diamonds are normally used to represent relationships and ovals are used to represent attributes.

Symbols used in Entity-Relationship Diagram are as follows:

• Represent Data Entity.

* Represent connection Administrator

* Connect two Entities or One to One relation.

• Represent the relationship.

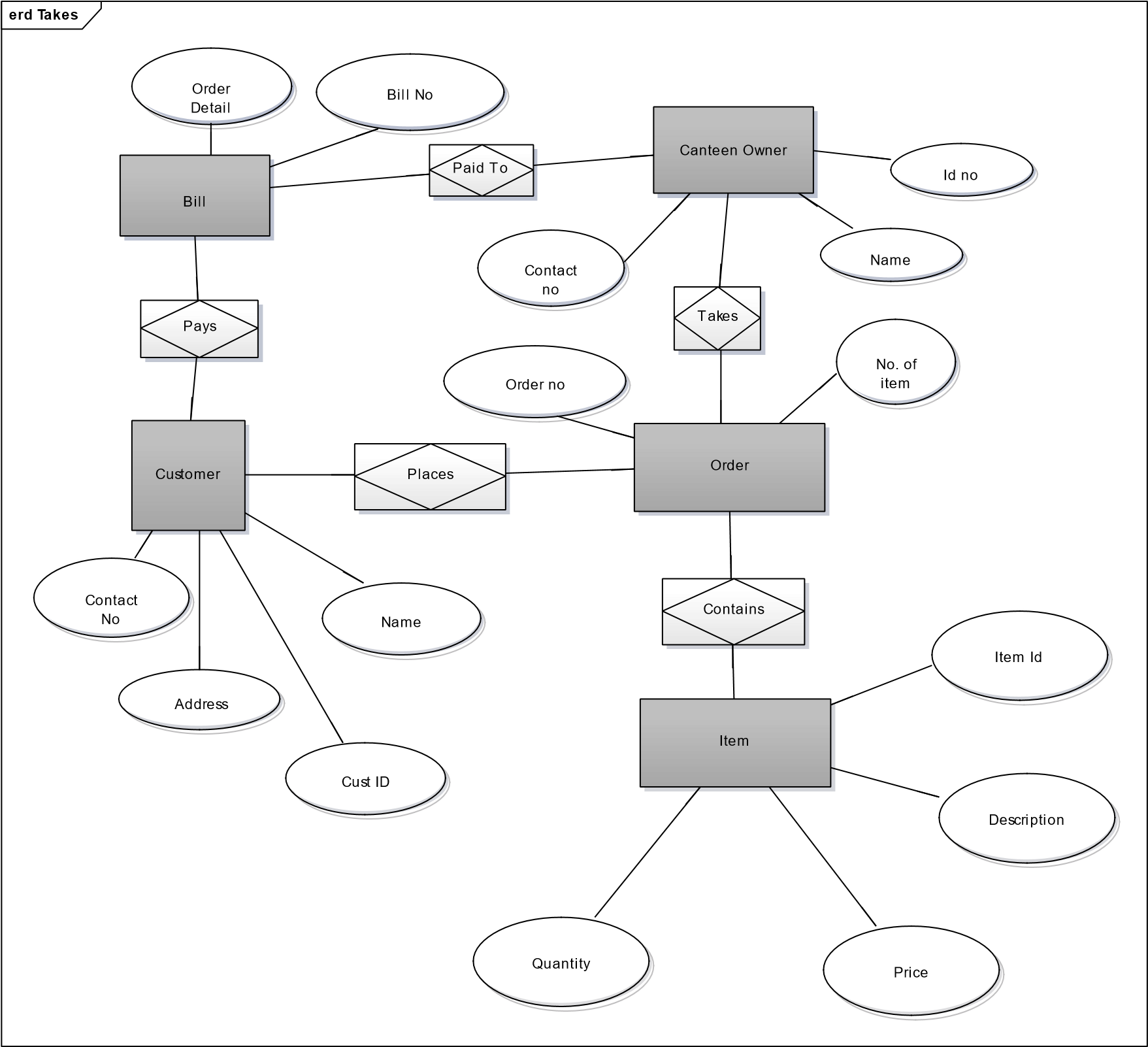


Fig. 4.1

### 8.5 DATA FLOW DIAGRAM

#### Level-0 DFD

Admin

Verification of College

Canteen

Automation

I/P

O/P

|  |
| --- |
| Canteen Owner |

Canteen Maintaining, I/P Automation O/P Canteen

|  |
| --- |
| Customer |

Canteen Placing Order

Automation

I/P O/P

#### Fig. 5.1

LEVEL 1 DFD

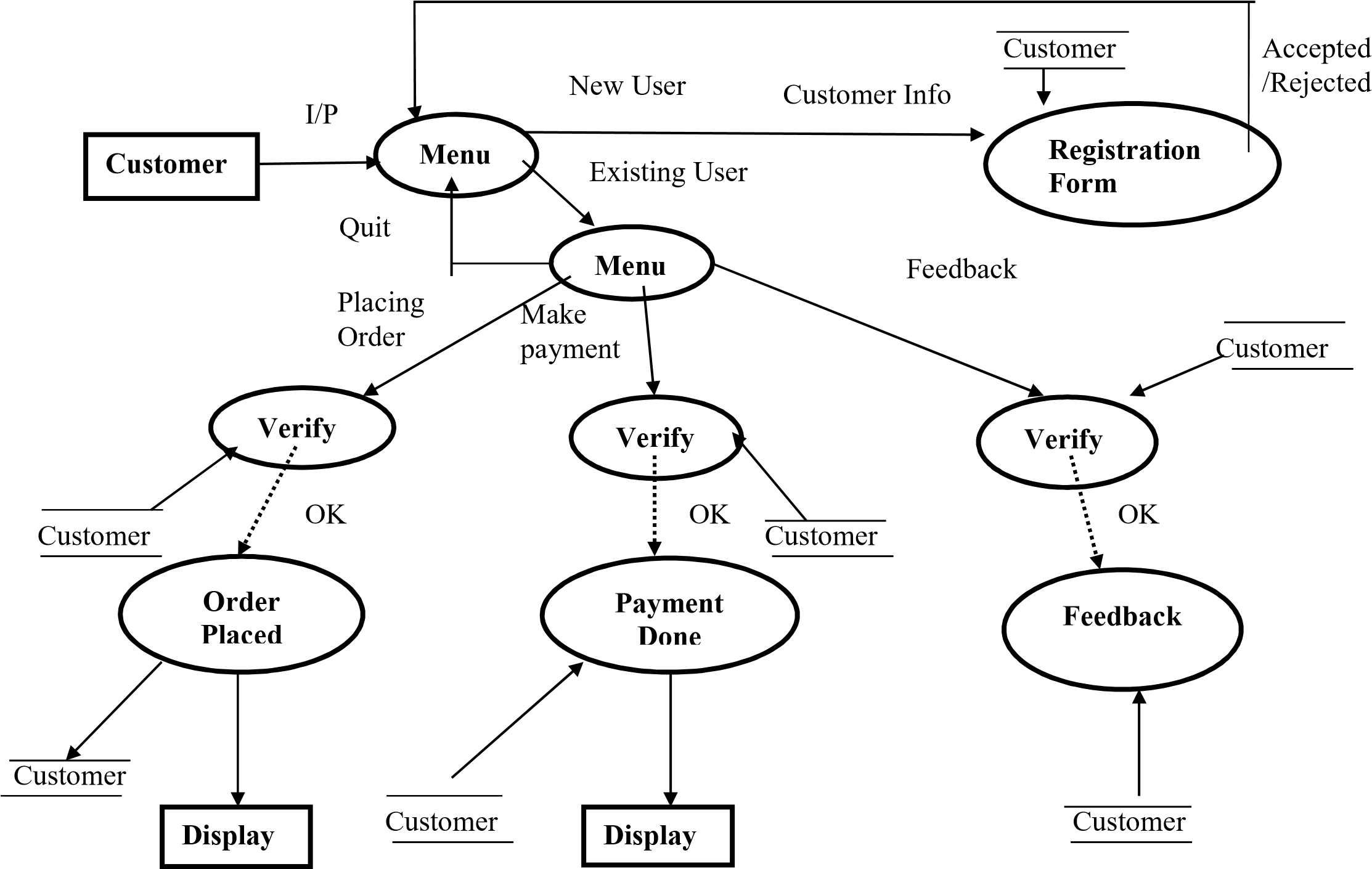
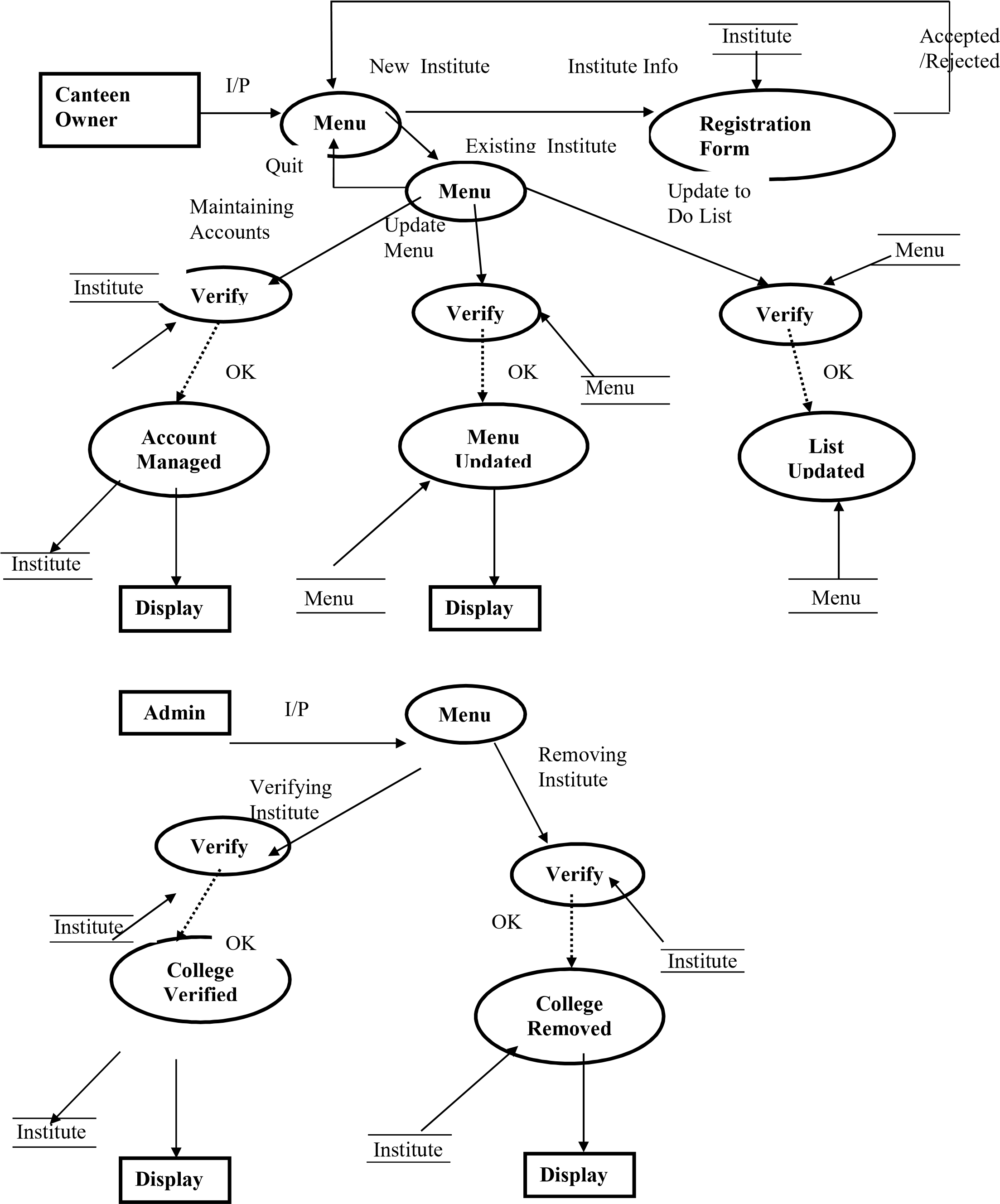


Fig. 5.2



## 9. DATABASE TABLES

Some abbreviations listed below -

* PK – Primary Key
* FK – Foreign Key
* NN – Not Null (Required)  UC – Unique Constraint
* SLT – Single Line of Text
* MLT – Multiple Lines of Text
* PG – Person or Group
* Yes/No (check box) – Yes/No

* 1. CUSTOMER

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Customer | | |
| Description | This table will contain the personal information of the customer who places an order in the canteen. | | |
| Primary Keys | Cust\_Id | | |
|  |  | | |
| Field Name | Data type | Constraints | Comments |
| Cust\_Id | Int | PK |  |
| FirstName | varchar(50) | NN |  |
| LastName | varchar(50) | NN |  |
| EmailId | varchar(50) | NN |  |
| UserName | varchar(50) | NN |  |
| Password | varchar(25) |  | Must contain 8 character |
| Mobile No. | varchar(10) |  | It could be home or cell, should take more than one and allow numbers + text like Home-408888-3333, cell-408-888-9999 |
| Address | varchar(100) |  |  |
| City | varchar(20) |  |  |
| State | varchar(20) |  |  |
| Country | varchar(50) |  |  |
| Zipcode | varchar(8) |  |  |

Table 1.1

* 1. ADMIN

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Admin | | |
| Description | This table will contain the personal information of the admin that verify the college registration. | | |
| Primary Keys | Admin\_Id | | |
| Foreign Keys | College\_Id | | |
|  |  | | |
| Field Name | Data type | Constraints | Comments |
| Admin\_Id | Int | PK |  |
| College\_Id | Int | FK |  |
| FirstName | varchar(50) | NN |  |
| LastName | varchar(50) | NN |  |
| EmailId | varchar(50) | NN |  |
| UserName | varchar(50) | NN |  |
| Password | varchar(25) |  | Must contain 8 character |
| Mobile No. | varchar(10) |  | It could be home or cell |
| Address | varchar(100) |  |  |
| City | varchar(20) |  |  |
| State | varchar(20) |  |  |
| Country | varchar(50) |  |  |
| Zipcode | varchar(8) |  |  |

Table 1.2

* 1. CANTEEN OWNER

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Canteen Owner | | |
| Description | This table will contain the personal information of the canteen owner. | | |
| Primary Keys | College\_Id | | |
| Foreign Keys | Cust\_Id,Order\_Id,Item\_No | | |
|  |  | | |
| Field Name | Data type | Constraints | Comments |
| College\_Id | Int | PK |  |
| Cust\_Id | Int | FK |  |
| Order\_Id | Int | FK |  |
| Item\_Id | Int | FK |  |
| FirstName | varchar(50) | NN |  |
| LastName | varchar(50) | NN |  |
| EmailId | varchar(50) | NN |  |
| UserName | varchar(50) | NN |  |
| Password | varchar(25) |  | Must contain 8 character |
| Mobile No. | varchar(10) |  | It could be home or cell |
| Address | varchar(100) |  |  |
| City | varchar(20) |  |  |
| State | varchar(20) |  |  |
| Country | varchar(50) |  |  |
| Zipcode | varchar(8) |  |  |

Table 1.3

* 1. MENU

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Menu | | |
| Description | This table will contain all the details of the menu card. | | |
| Primary Keys | Item\_Id | | |
| Foreign Keys | College\_Id,Cust\_Id,Order\_Id | | |
|  |  | | |
| Field Name | Data type | Constraints | Comments |
| Item\_Id | Int | PK |  |
| College\_Id | Int | FK |  |
| Order\_Id | Int | FK |  |
| Item\_Id | Int | FK |  |
| Cust\_Id | Int | FK |  |
| ItemName | varchar(50) | NN |  |
| Quantity | Int | NN |  |

Table 1.4

* 1. ORDER

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Order | | |
| Description | This table will contain the details of Order that customer does. | | |
| Primary Keys | Order\_Id | | |
| Foreign Keys | College\_Id,Cust\_Id,Bill\_Id,Item\_Id | | |
|  |  | | |
| Field Name | Data type | Constraints | Comments |
| Order\_Id | Int | PK |  |
| Item\_Id | Int | FK |  |
| College\_Id | Int | FK |  |
| Bill\_Id | Int | FK |  |
| Item\_Id | Int | FK |  |
| Cust\_Id | Int | FK |  |

Table 1.5

* 1. PAYMENT

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Payment | | |
| Description | This table will contain the details of payment done by the customer. | | |
| Primary Keys | Bill\_Id | | |
| Foreign Keys | College\_Id,Cust\_Id,Order\_Id,Item\_Id | | |
| Field Name | Data type | Constraints | Comments |
| Bill\_Id | Int | PK |  |
| College\_Id | Int | FK |  |
| Order\_Id | Int | FK |  |
| Item\_Id | Int | FK |  |
| Cust\_Id | Int | FK |  |
| Payment\_mode | varchar(50) | NN |  |
| Amount | Int | NN |  |
| Credit/Debit\_card\_no | Int | NN |  |

Table 1.6

WEBSITE REFERENCE

www.google.com www.tutorialpoint.com

www.w3schools.com