CANTEEN AUTOMATION SYSTEM USING ANDROID

Main Project Report

Submitted by

NOREEN VALLONTHAIEL

Reg No: FIT20MCA-2083

Submitted in partial fulfillment of the requirements for the award of the degree of

Master of Computer Applications

Of

A P J Abdul Kalam Technological University

Focus on Excellence

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY (FISAT)®

ANGAMALY-683577, ERNAKULAM(DIST)

JULY 2022

DECLARATION

I, **NOREEN VALLONTHAIEL** hereby declare that the report of this project work, submitted to the Department of Computer Applications, Federal Institute of Science and Technology (**FISAT**), Angamaly in partial fulfillment of the award of the degree of Master of Computer Application is an authentic record of my original work.

The report has not been submitted for the award of any degree of this university or any other university.

Date :

Place: Angamaly **Noreen Vallonthaiel**

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY (FISAT)®

ANGAMALY, ERNAKULAM-683577

CERTIFICATE

This is to certify that the project report titled **" CANTEEN AUTOMATION SYSTEM USING ANDROID "** submitted by **NOREEN VALLONTHAIEL**, **(Reg No: FIT20MCA-2083)** towards partial fulfillment of the requirements for the award of the degree of Master of Computer Applications is a record of bonafide work carried out by her during the year 2022.

Project Guide
Ms. Manju Joy

Head of the Department
Dr. Deepa Mary Mathews

Submitted for the viva-voce held on . . . . *at* . . . .

Examiner :

ACKNOWLEDGEMENT

Gratitude is a feeling which is more eloquent than words, more silent than silence. To complete this project work I needed the direction, assistance and co-operation of various individuals, which is received in abundance with the grace of God.

I hereby express my deep sense of gratitude to **Dr. Manoj George,** Principal, FISAT and **Dr. C Sheela,** Vice principal, FISAT for allowing me to utilize all the facilities of the college.

My sincere thanks to **Dr. Deepa Mary Mathews,** HOD, Department of Computer Applications FISAT ,who had been a source of inspiration. I express heartiest thanks to **Ms. Manju Joy,** my internal guide for her encouragement and valuable suggestions. I express my heartiest gratitude to my scrum master **Dr. Sujesh P Lal,** and the faculty members in my Department for their encouragement and constructive suggestions and comment during the project work.

Finally I wish to express my thanks to my parents, friends and well-wishers who extended their help in one way or other in preparation of my project. Besides all, I thank GOD for everything.

ABSTRACT

Canteen automation system is to provide fast services to college students. Usually people have to go to canteen and order the foods and they have to wait in queue for a long time to get the orders. But with the help of this you just have to follow a very simple process to order your stuffs. And you need not to wait in the long queue.

This application will provide the list of menu list with different categories. User can select any item from canteen and can order for it by using debit card payment. Users have to register with valid details to login with canteen. Users first choose the canteen and see their menu by selecting the date. Also, they can choose from the category such as breakfast, lunch, dinner etc and can order the food from the menu after ordering and payment process a bill will be generated where the details of the order are shown and a QR code will be generated.

Canteen automation system manages the all details of food items which contains name, description, preparation time, price etc. Also, canteen can view the details of the students orders and can view the conformed order and update the status of the order accordingly. when student order a food then QR code will be generated for the canteen users to see the details of the student that ordered. Also, students can check their order, order history and able to delete the order according to order status.

Contents

1 INTRODUCTION 8

2 PROOF OF CONCEPT 10

3 SYSTEM ANALYSIS 14

3.1 Feasibility Study 15

3.1.1 Technical feasibility 16

3.1.2 Economic feasibility 16

3.1.3 Operational feasibility 17

4 USER CHARACTERISTICS 18

5 REQUIREMENT ANALYSIS AND SPECIFICATION 20

5.1 System Specification 21

5.1.1 Hardware specification 21

5.1.2 Software Specification 21

5.2 About Software Tools and Platform 22

5.3 Input Design 23

5.4 Output Design 24

5.5 Database Design 24

6 IMPLEMENTATION 26

6.1 Modules 29

6.2 Data Flow Diagram (DFD) 30

6.3 Entity Relationship Diagram 41

7 SYSTEM TESTING 43

7.1 White Box Testing 44

7.2 Black Box Testing 44

7.3 Testing and validation 44

7.4 Testing 44

7.4.1 Unit Testing 45

7.4.2 Integration Testing 46

7.4.3 Output Testing 46

7.5 Validation Check 47

8 CONCLUSION AND FUTURE SCOPE 48

8.1 Conclusion 49

8.2 Future Scope 50

9 APPENDIX 51

9.1 Source Code 52

9.2 Screenshot 66

10 REFERENCES 78

Chapter 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.

As we aware of the problems of the traditional system is facing such as:-
1.Verbal communication between cashier and customer or telephonic communication.
2.Food customization.
3.Menu display.

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, 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, it is entered into the database and then retrieved in pretty much real-time, by a android 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.

Chapter 2

PROOF OF CONCEPT

A lot of studies have been made on canteen automation system in the literature.The major focus on making canteen workers monitor overall day to day business analytic more accurately with an ease. It increases the presence of canteen fascinates core customers towards 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 made easier and single handed with the required inbuilt features. Whenever your customers are busy with their work, instead of coming to your canteen they can just open your application, 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. This canteen automation system is useful for the canteen as well customer because canteen automation system save the time of the canteen's workers as well of the customers and as saves the manpower.

Self-service or self-ordering in canteen industry refers to the canteen taking orders from customers through applying various types of technologies such as internet and many others. Self-service or self-ordering is successful when it is applied at canteens in many other countries. The usage of the self-service or self-ordering technology is proven to benefit most of the investors.

Odesser-Torpey (Odesser-Torpey, 2008) reports that most of the Americans hate waiting for an order. Therefore, they prefer self-service technology, which can be in form of text-messaging, the internet and kiosk. Usually, the customer prefers self-service because of speed and convenience in making order and transactions that self-activated terminals are more likely to serve as ordering innovation in the future. The implementation of alternative ordering can increase check size, free up counter, staff that need to serve customers and take money handling out of service equation.

Bhatnagar (Bhatnagar, 2006) mentioned that the innovation of kiosk and computerized table top ordering screen will force restaurant industry re-jigger an often-used acronym quick service. Customer can get information or search for recipes from the kiosk and internet. The kiosk and internet also take orders and receives credit cards and debit cards payment. As a result, wrong order and long queue can be avoided, order staff can be arranged to somewhere else and focus to speed up delivery orders. On the other hand, a table-top touch screen order system can take customer orders as well, handle other customer requests such as refill drinks, call a waiter and make payment by credit or debit card.

Bytes, a restaurant located at Canterbury has been successfully standing apart from the competitors because of applying online service ordering and the payment concepts. The system used in bytes allows the customers make an order through the touch screen, and the order will be directed to the bar or kitchen. The system also offers games after a customer has placed orders while internet access will be provided to customers in the future. Touch screen ordering reduces the need of the waiter. The system also provides database for customers' habits and preferences, generate the management reports, perform analysis as well as allows the menu to be uploaded instantly. (Brickers, 2006).

Based on study, it is possible for applying the online food ordering system to the fast food restaurants and canteens in Nigeria. This is because the system can improve workplace efficiency, increase sales of the restaurant as well as reduce making incorrect orders. As a result, it is worth for investing on the system, whereby it can shorten the return on investment.

In addition, the system should be supported by the food origin taste and services to maintain the customers loyalty and satisfaction. However, the widely implementing of automated canteen ordering system may cause the influx of labor due to the elimination of waiters in restaurant industry. Even the system is important to be implemented, yet there is still some risk in other factors such as a direct interaction and restaurant design concept, which need to be considered for ensuring the success of the system.

Gan (Gan, 2002) proposed to develop an automated canteen ordering system that allows customers to place orders anytime at any place. The system helps to manage order from customer as well as advertise promotion. It allows kitchen staff to view ordering information, management to manage fast raw materials and staff to search customer delivery and profile information. This system helps to reduce queue issues during peak hours, speed up food preparation and increase customer volumes. As a result, market share of fast food restaurant/canteen can be boosted up and increases return of investment for the investor.

De Leon (De Leon, 2008) mentioned that there are several aspects that should be included in a good, automated ordering system. System should be simple to navigate, not clustered and easy to make an order. (Sharma, 2007) designed with professionals looking with search engines optimize capability and available 24hours. The system should also have a secure payment gateway to protect their customers' credit card information, fast and keep track on orders and sales history easily as well as generate a comprehensive sales report. (Sharma, 2007).

Chapter 3

SYSTEM ANALYSIS

System study refers to the process of examining a situation with the intent of improving it through better process and methods. System study is, therefore, the process of gathering and interpreting facts, diagnosing problem and using the information to recommend changes in the system, in other words, it means a detailed explanation or description. Before computerizing a system under consideration, it must be analyzed.

We need to study how it functions currently, what are its problems, and what are the requirements that the proposed software should meet. Every organization has a system generating and monitoring various kinds of data and reports that are very relevant for the smooth functioning of that organization. Effective decisions are to be taken by the management based on current information which provide accurate and help the management.

The main aim of the project is to maintain all the records by computerized the canteen automation system. The system is a manual system. There are a lot of limitations like large number of man power. On using this project students could view and order directly. It gives computerization helps the supplier to reduce the paperwork.

The main components of making software are:

- System and software requirement analysis.
- Design and implementation of software.
- Ensuring, verifying and maintaining software integrity.

3.1 Feasibility Study

Feasibility study is made to see if the project on completion will serve the purpose of the organization for the amount of work, time and effort spent on it. Study lets the developer foresee the future of the project and its usefulness. Finding out whether a new system is required or not. The study is carried out to the best system that meet performance requirement. This entails identification, description and evaluation of candidate system and selection of the best system for the best system for the job. It simply identifies whether the proposed system is feasible to the organization or not.

There are three aspects in the feasibility study portion of the preliminary investigation

- Technical feasibility
- Economic feasibility
- Operational feasibility

3.1.1 Technical feasibility

Technical feasibility is the need of hardware and software, which are needed to implement the proposed system in the organization. Technical requirements are to be fulfilled to make the proposed system work. This should be necessarily predetermined so as to make the system more components. It is the study that determines whether the work for the project can be done with current equipment's, existing software technologies and available personnel.

Technical feasibility is the most difficult area to access at the stage of the system development process. Because objectives, function and performance are somewhat hazy, anything seems possible if the right assumptions are made.Technical feasibility centers on the existing computer system [hardware, software] and to what extent it can support the proposed addition.

3.1.2 Economic feasibility

Economic feasibility deals with the analysis of cost against benefits i.e. whether the benefits to be enjoyed due to the new system are worthy, when compared to the costs to be spent on the system.

Economic justification is generally the “bottom-line” consideration for the most system. Long term corporate income strategies, impact on other profit centers or products, cost of the resources needed for development, and potential market growth. Hence this project was economically justified for development in this organization. Especially in the present scenario, where the objectives are towards compatibility, reduced cost is weighed against the ultimate income or benefit derived from the developed system.

This system is having been implemented such that it can be used to analysis the traffic. It does not require any extra equipment or hardware to implement. It is economical feasible to use.

3.1.3 Operational feasibility

During feasibility analysis operational study is must. Because according to software engineering principles operational feasibility or in the other word's usability should be very high. Although analysis is done and found that system is operational. Managing the time and work system is possible to develop operationally.

Chapter 4

USER CHARACTERISTICS

The Canteen Automation System Android Application provide the user to perform their task in an easy and much less complexity to avoid redundancy. This system ensures that the user accessing the system can ensure maximum efficiency and they can depend on the system for the desired result. There are three user characteristics in this system: -

- Administrator
- Canteen
- Students

Administrator

Administrator is the super user and main controller of this system. His/her controls registration of both canteen user and students. Also, can view registered canteens and students.

Canteen

Once the canteen details are registered, a concern person of canteen can login using their login credentials. In this particular canteen have login with username and password. They can add food menu based on date and can view the details entered about menu on the basis of date. Also, they can view the details of students order.

Students

The student will have little privilege when compared to the administrator. Student can register and login to the page. They can edit account. They can search the menu based on date or view the ordered details and see the transaction history.

Chapter 5

REQUIREMENT ANALYSIS AND SPECIFICATION

5.1 System Specification

5.1.1 Hardware specification

Processor	i5 or i7 (i7 is better)
RAM	12GB (Minimum)
Hard Disk	500GB or above
Other	Mouse and Keyboard

5.1.2 Software Specification

Tool	Android studio, Python
Language	Python, Java
Operating System	Windows 11 or later
Front End	Android
Back end	Django

5.2 About Software Tools and Platform

It is very important to select the appropriate software so that the software works properly. Below is the software that are required to make a new system:

- Android
- Django

Frontend: Android

Android is a software package and linux based operating system for mobile devices such as tablet computers and smartphones. The operating system have developed a lot in last 15 years starting from black and white phones to recent smart phones or mini computers. One of the most widely used mobile OS these days is android. The Android is software that was founded in Palo Alto of California in 2003. It is developed by Google and later the OHA (Open Handset Alliance). Java language is mainly used to write the android code even though other languages can be used. The goal of android project is to create a successful real-world product that improves the mobile experience for end users. There are many code names of android such as Lollipop, KitKat, Jelly Bean, Ice cream Sandwich, Froyo, Ecliar, Donut etc which is covered in next page. OHA is a consortium of 84 companies such as Google, Samsung, AKM, Synaptics, KDDI, Garmin, Teleca, Ebay, Intel etc. It was established on 5th November, 2007, led by Google. It is committed to advance open standards, provide services and deploy handsets using the Android Platform.

Backend: Django

Django is a free and open-source, Python-based web framework that follows the model–template–views (MTV) architectural pattern. It is maintained by the Django Software Foundation (DSF), an independent organization established in the US as a non-profit.

Django's primary goal is to ease the creation of complex, database-driven websites. The framework emphasizes reusability and ”pluggability” of components, less code, low coupling, rapid development, and the principle of don't repeat yourself. Python is used throughout, even for settings, files, and data models. Django also provides an optional administrative create, read, update and delete interface that is generated dynamically through introspection and configured via admin models.

5.3 Input Design

Input design is the process of converting a user-oriented description of the inputs to a computer-based system into a programmer-oriented specification. The quality of system input determines the quality of system output. Input specification describes the manner in which data enter the system for processing. Input design features can ensure the reliability of the system and produce result from accurate data or they can result in the production of errors. The input design also determines whether the user can interact efficiently with the system. Input design requires consideration of the needs of the data entry operator. Three data entry considerations are: • The field length must be documented • The sequence of fields must match the sequence of the fields on the source document. • The data format must be identified to the data entry operator. In our system almost all inputs are being taken from the databases. To provide adequate inputs we have to select necessary values from the databases and arrange it to the appropriate controls. Inaccurate input data are the most common cause of errors in data processing. Errors entered by data entry can be controlled by input design. Input design is the process of converting user-oriented inputs to a computer-based format. There are three major approaches for entering data into the computer. They are menus, formatted forms and prompts. A menu is a selection list that simplifies computer data access or entry. Instead of remembering what to enter, the user choices from a list of option. A formatted form is a preprinted form or a template that request the user to enter data in appropriate location. It is a fill-in-the-blank type form. The form is flashed on the screen as a unit. In prompt the system displays one enquiry at a time, asking the user for a response.

5.4 Output Design

One of the important features of an information system for users is the output it produces. Output is the information delivered to users through the information system. Without quality output, the entire system appears to be unnecessary that users will avoid using it. Uses generally merit the system solely by its output. In order to create the most useful output possible. One works closely with the user through an interactive process, until the result is considered to be satisfactory. Output design has been an ongoing activity almost from the beginning of the project. In the study phase, outputs were identified and described general in the project directive. A tentative output medium was then selected and sketches made for each output. In the feasibility analysis, a "best" new system was selected; its description identified the input and output media. In the design phase the system has included an evaluation and selection of specific equipment for the system. Output design generally deals with the results generated by the system i.e., reports. These reports can be generated from stored or calculated values. Reports are displayed either as screen window preview or printed form. Most end users will not actually operate the information system or enter data through workstation, but they will use the output from the system. Outputs from computer systems are required primarily to communicate the results of processing to the user. They are also used to provide a permanent copy of these results for later consultation.

5.5 Database Design

The Database design is the process of producing a detailed data model of a database. The logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a Data Definition Language, which can then be used to create a database. A fully attributes for each entry. The term database design can be used to describe many different parts of the design of an overall data base system. Principally, and most correctly, it can be though to fast the logical design of the base data structures used to store the data. In the relational model, these are the tables and views. In an object data base, the entities and relationships map directly to object classes and named relationships. However, the term data base design could also be used to apply to the overall process of designing, not just the base data structures, but also the forms and queries used as part of the overall data base applications within the database management system.

Chapter 6

IMPLEMENTATION

Implementation is the stage in the project where theoretical design is turned into a working system and is giving confidence on the new system for the users which will work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementations, design of methods to achieve the changeover, an evaluation, of change over methods. Apart from planning major tasks of preparing the implementation are education and training of users. The major complex system being implemented the more evolved will be the system analysis and the design effort required just for implementation. An implementation coordination committee based on policies of individual organisation has been appointed. The implementation process begins with preparing plan for implementation of the system. According to this plan the activities are to be carried out discussions made regarding the equipment and resources and the additional equipment has to be acquired to implement the new system.

Implementation is the final and important phase. The most critical stage in achieving a successful new system and in giving the users confidence that the new system and in giving the users confidence that the new system will work and be effective. The system can be implemented only after thorough testing is done and if found to working according to the specification.

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 made easier and single handed with the required inbuilt features.

The working of canteen automation system whenever your customers are busy with their work, instead of coming to your canteen they can just open your app, 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.

How it 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.

How it benefits a student?

Now a day's people are getting busy with their work, making the canteen online even saves customers time undoubtedly. Is allow them to order food online by creating a flexible ordering platform and serve them in time. The customer can select the food items from the online menu on the canteen website and can order 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 system is useful for the canteen as well customer because canteen automation system save the time of the canteen's workers as well of the customers and as saves the manpower.

6.1 Modules

The Restaurant Table Booking Android System provides us facilities to manage the activities taking place in restaurant Table Booking. There are modules in this project. They are:

1. Registration management
2. Login management
3. Food management
4. Sales management

1. Registration management

There are two registration for student and canteen which are entered with basic details. .

2. Login management

There are login for three users:- admin, canteen and student. Each users deals with username and password.

3. Food management

Food module deals with the food items severed in a canteen mainly based on date. This module also contains the option to select a category such as breakfast, lunch, snacks.

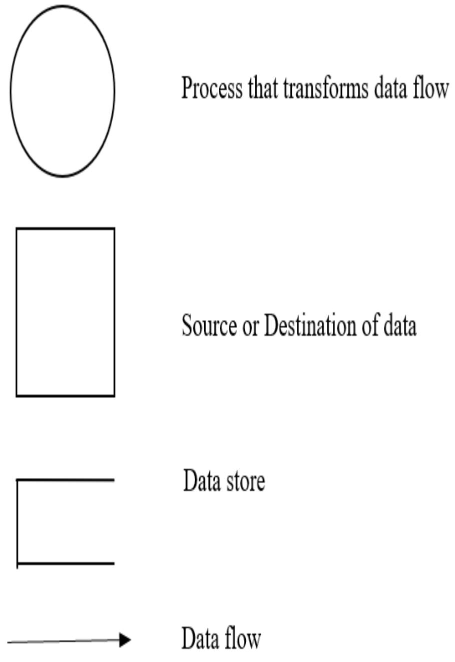
4. Sales management

This module performs the actions of managing order in the system. Customer can make order. We can perform the following task such as add order, update existing order details and view the order details. Also, payment process happens after ordering food. After, payment a bill will be generated for student to view details of the order etc.

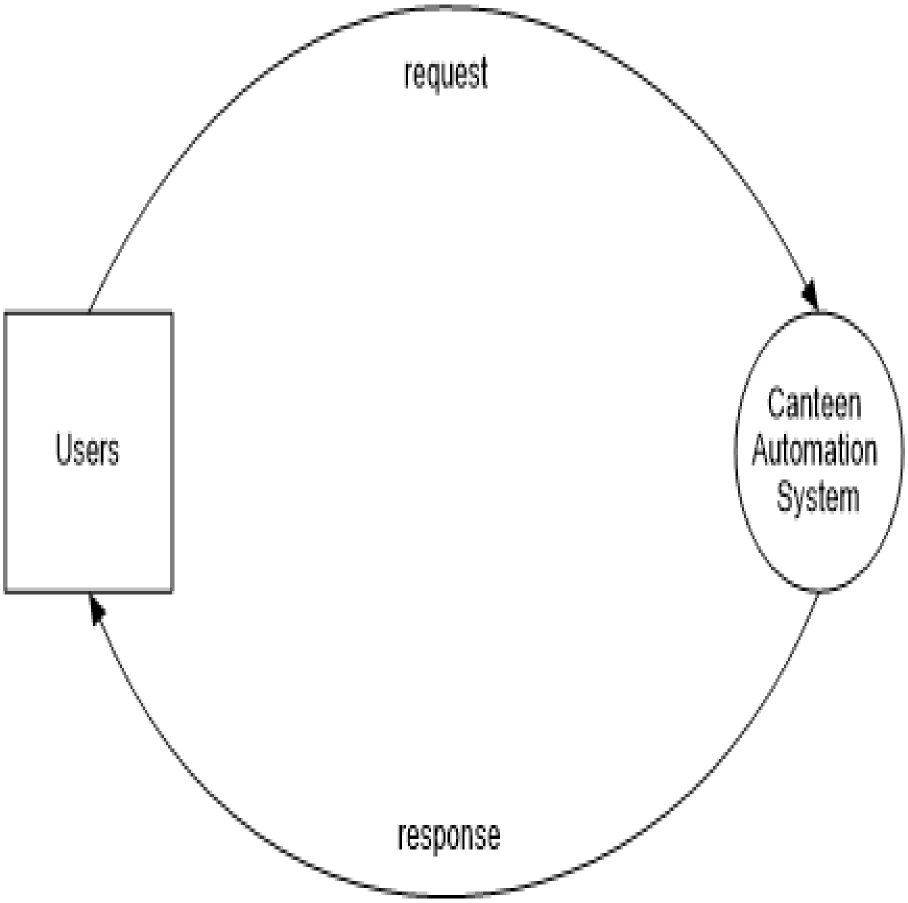
6.2 Data Flow Diagram (DFD)

A data flow diagram (DFD) is graphical tool used to describe and analyze movement of data through a system. These are central tool and the basis from which the other components are developed. The transformation of data from input to output, trough processed, may be described logically and independently of physical components associated with the system. These are known as the logical data flow movement of data between people, departments and workstations. A full description of a system actually consists of a set of data flow diagrams.

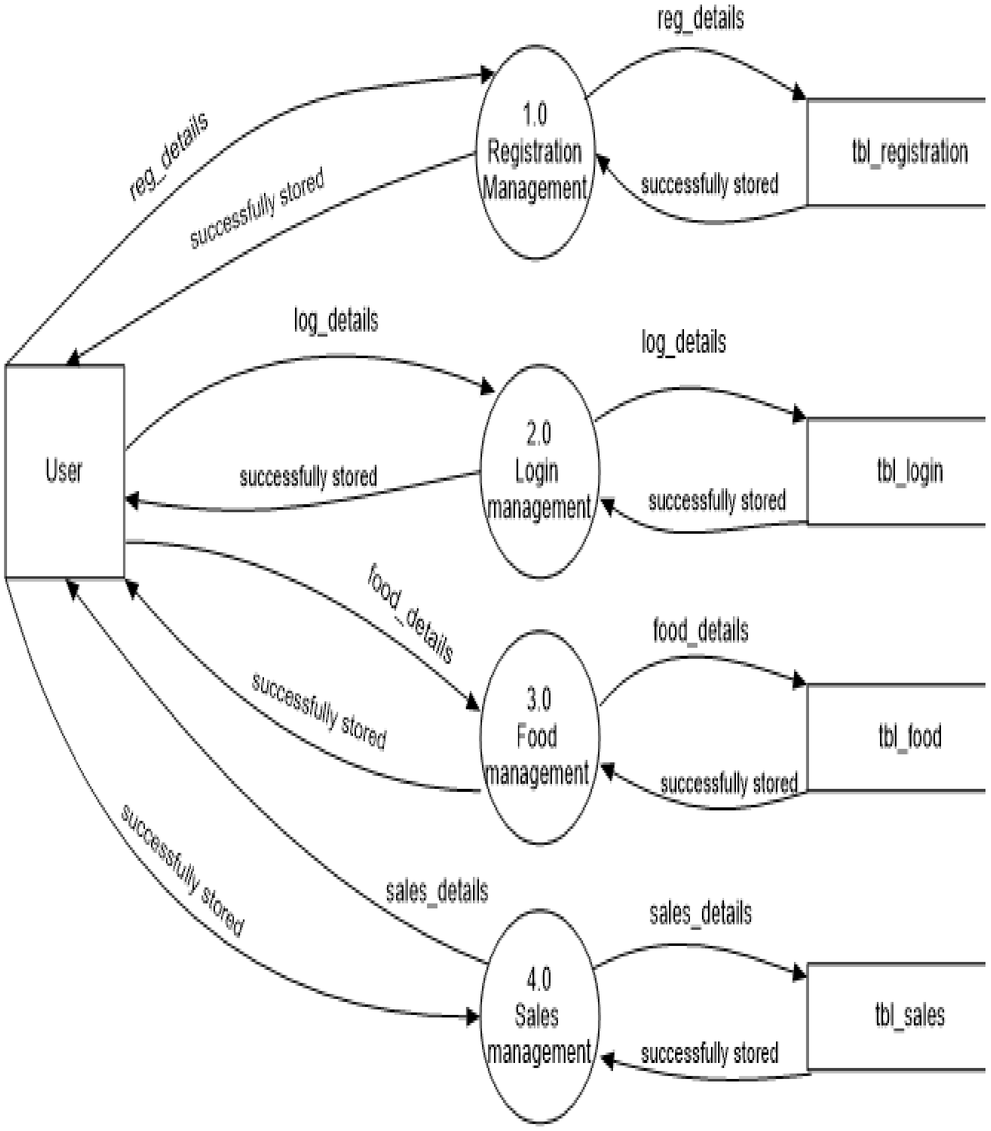
A DFD is also known as a "bubble chart" has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. So, it is the starting point of the design to the lowest level of detail. A DFD consists of a series of bubbles joined by data flows in the system.
In the DFD, there are four symbols


Process that transforms data flow
Source or Destination of data
Data store
Data flow

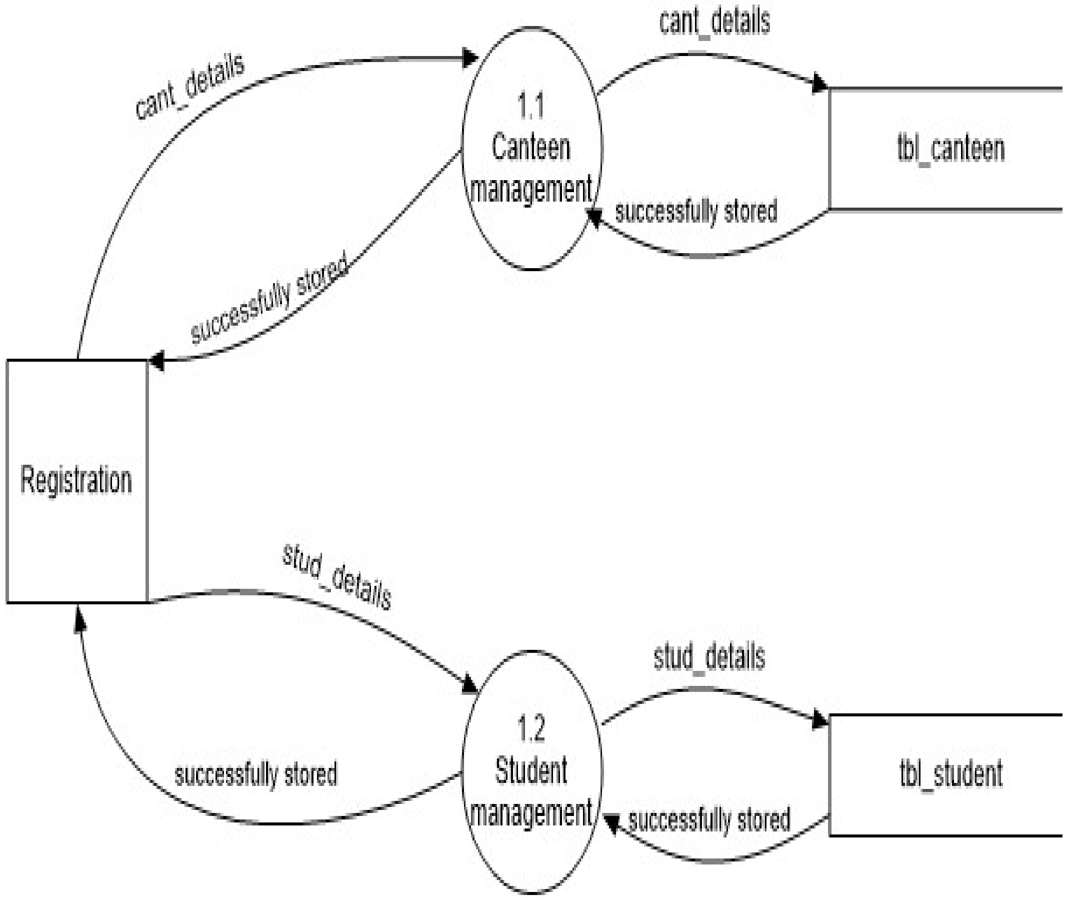
Level 0 DFD - showing Canteen Automation System



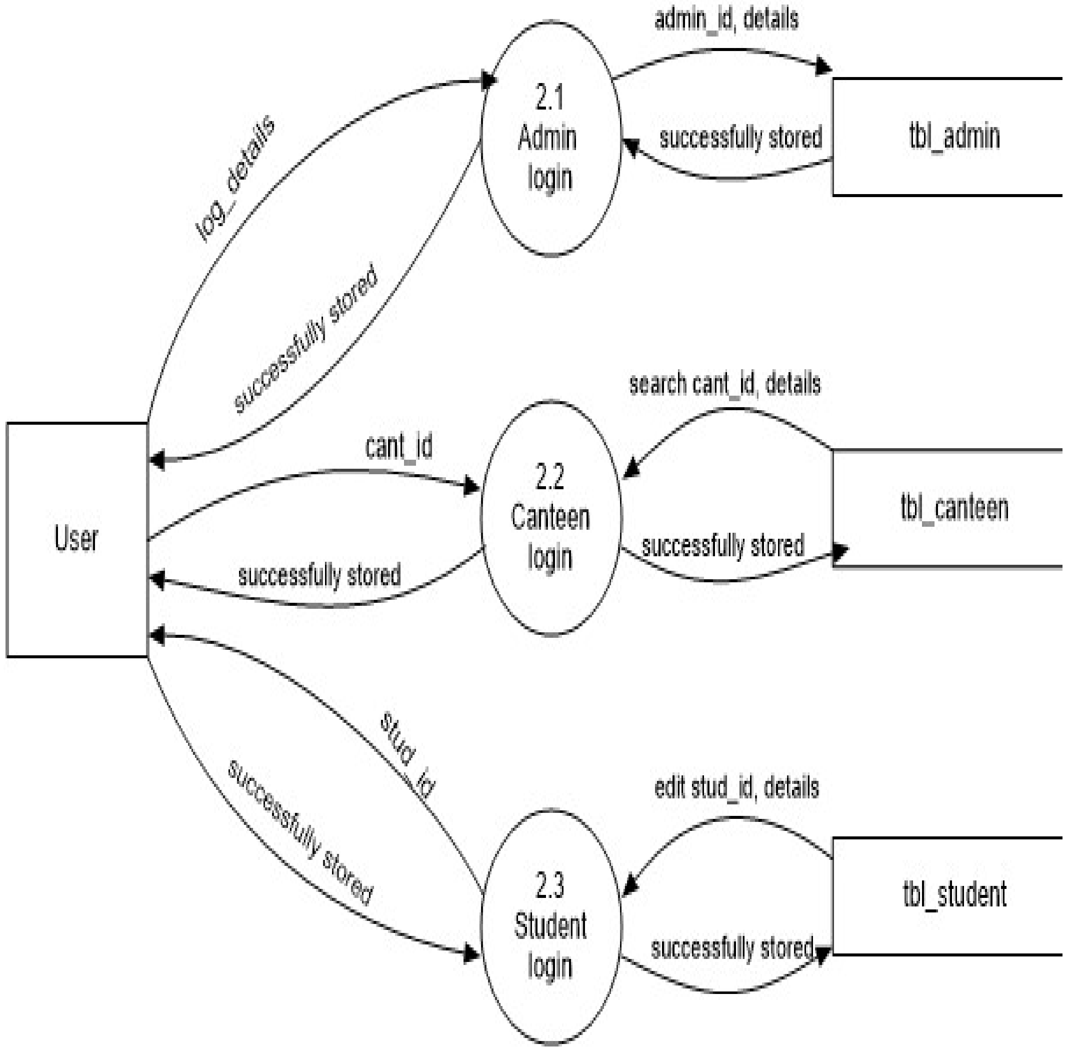
Level 1 DFD - showing all the modules in Canteen Automation System



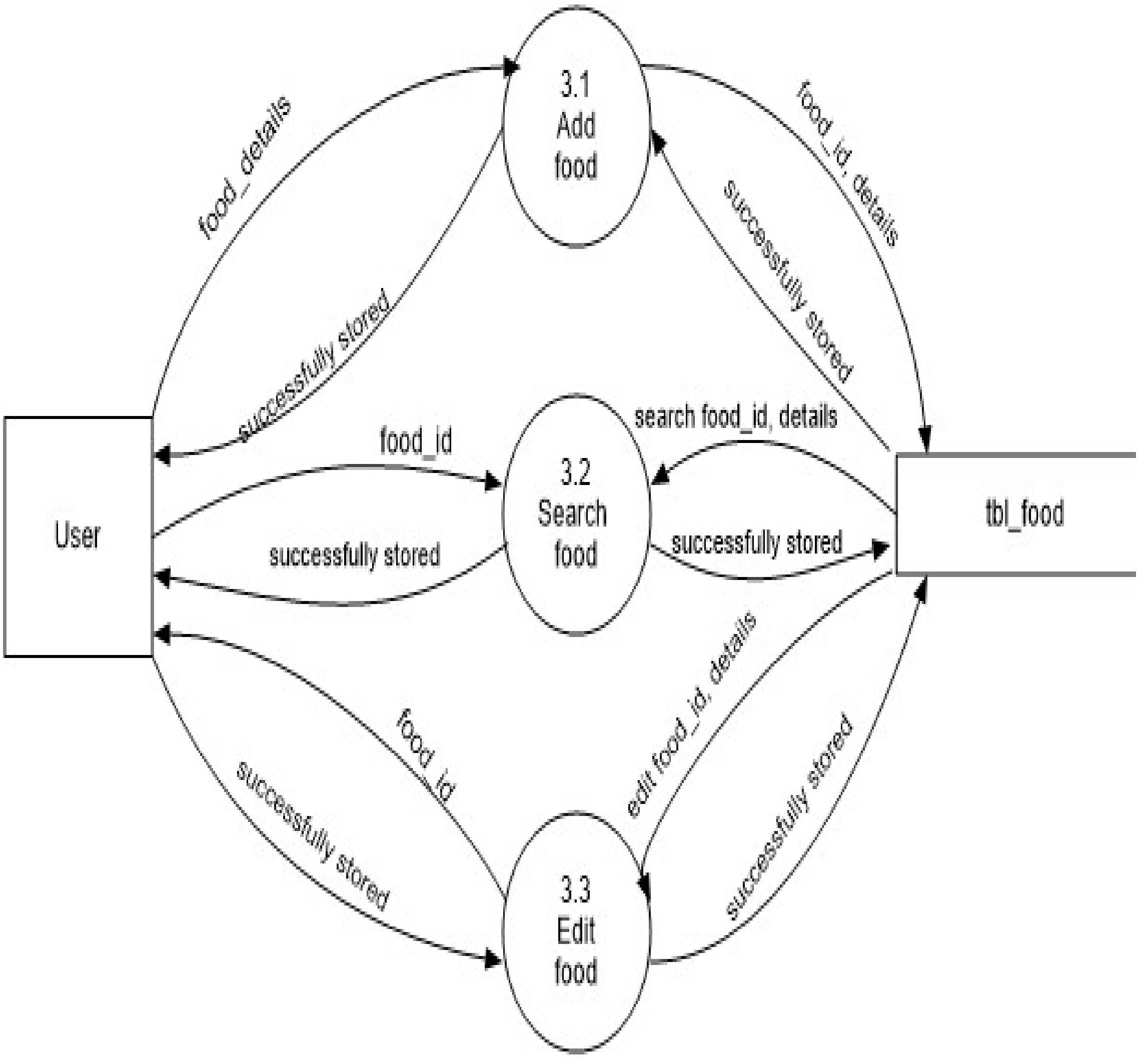
Level 2 DFD - showing registration management



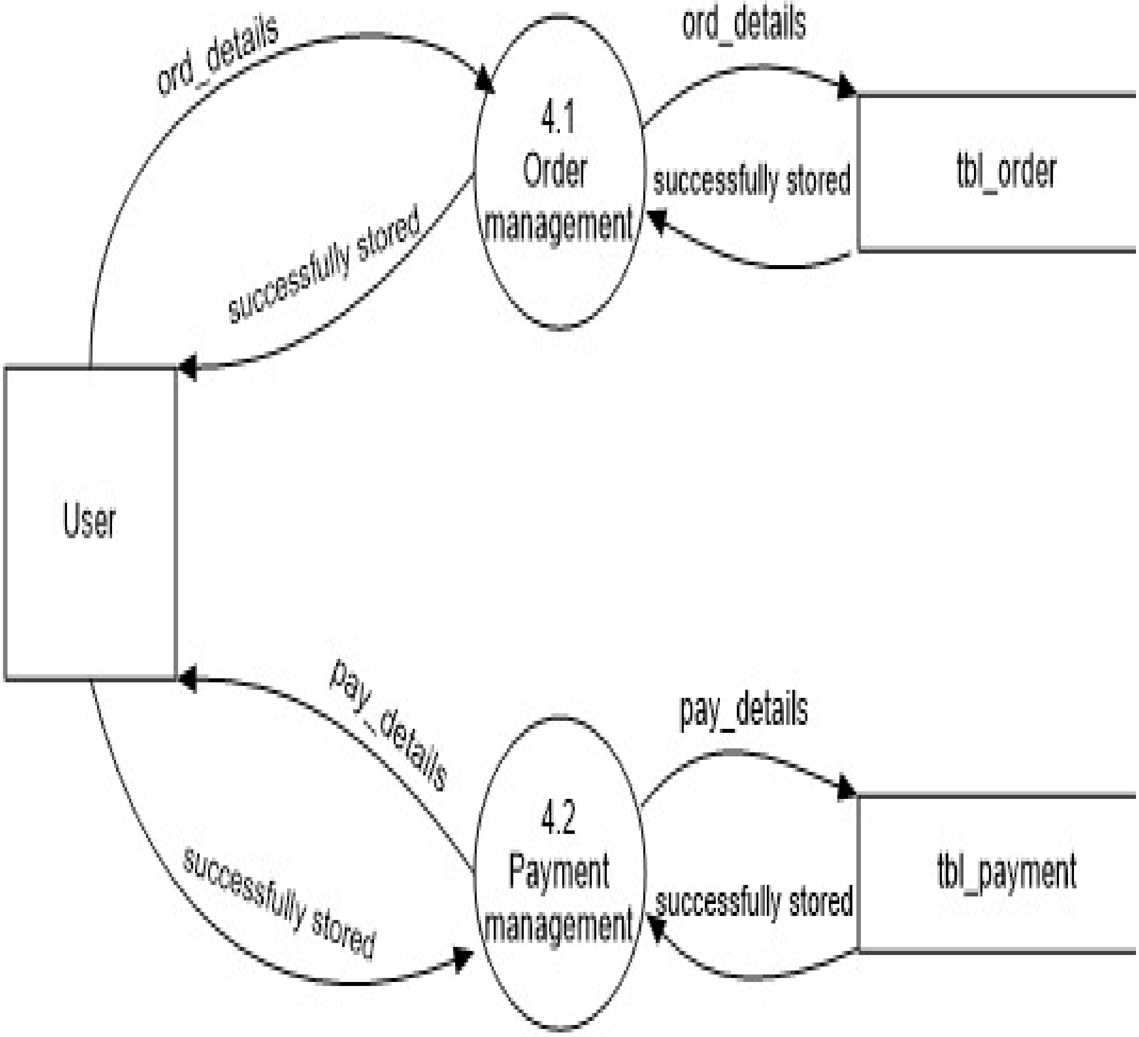
Level 2 DFD - showing login management



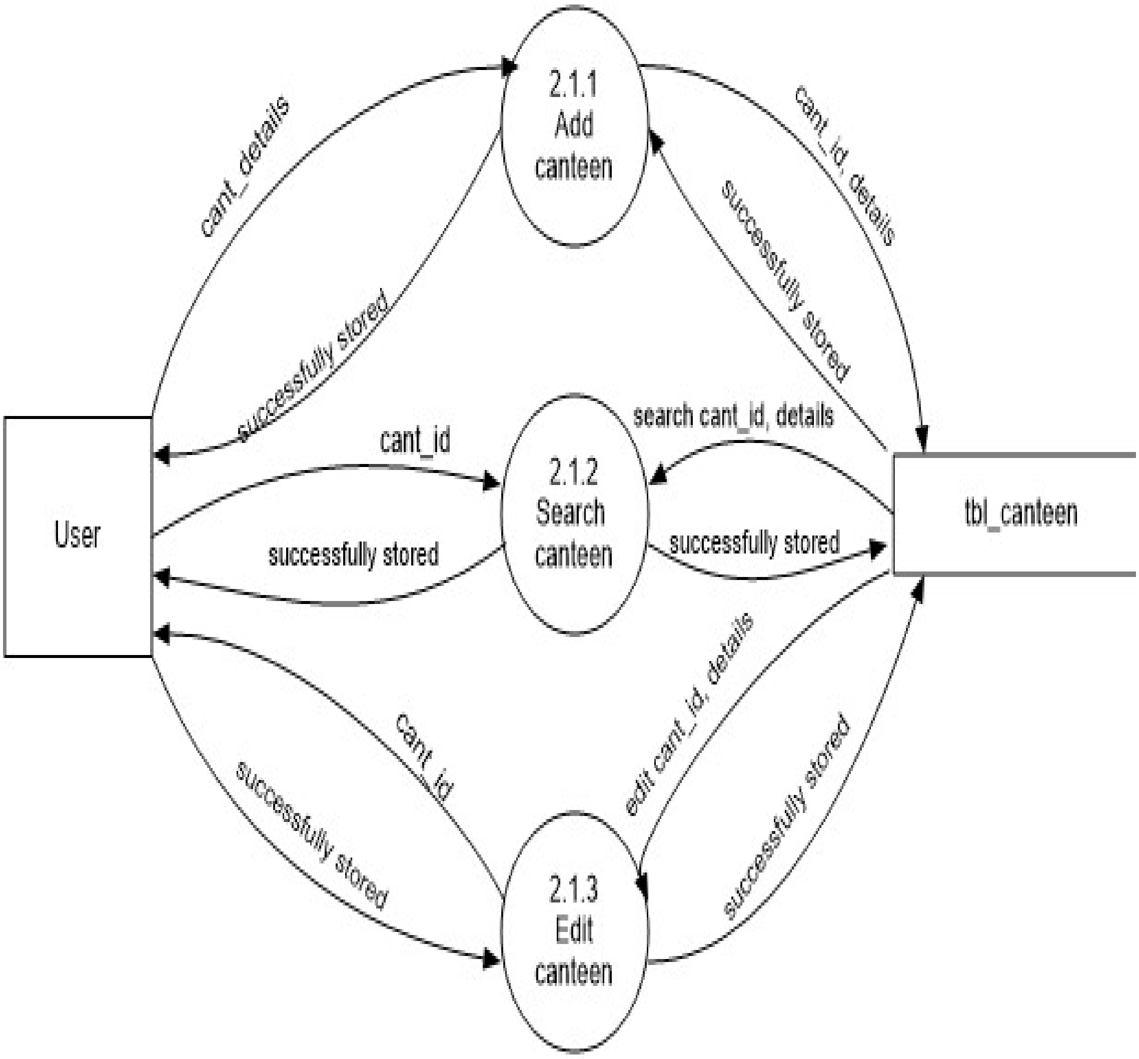
Level 2 DFD - showing food management



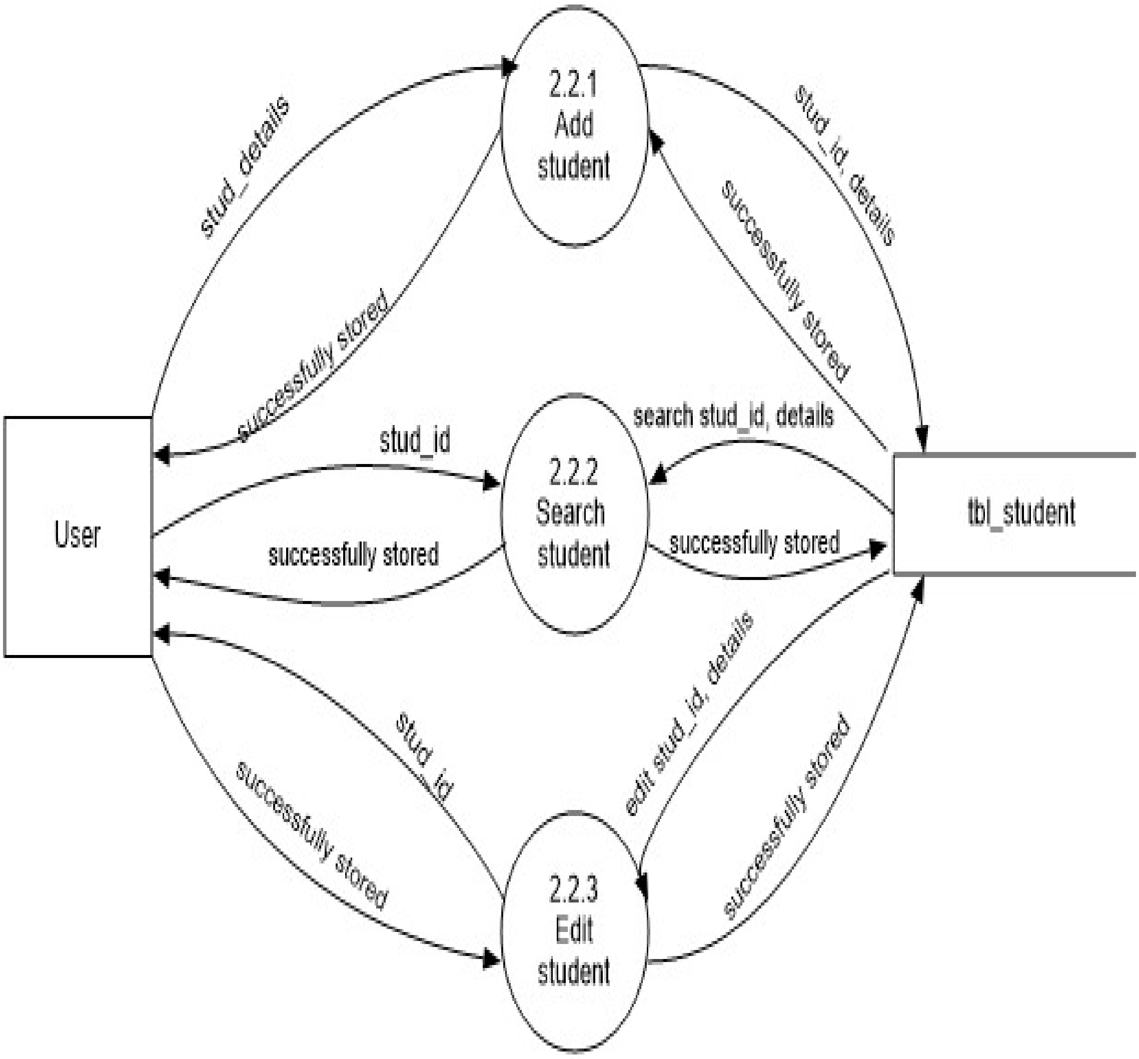
Level 2 DFD - showing sales management



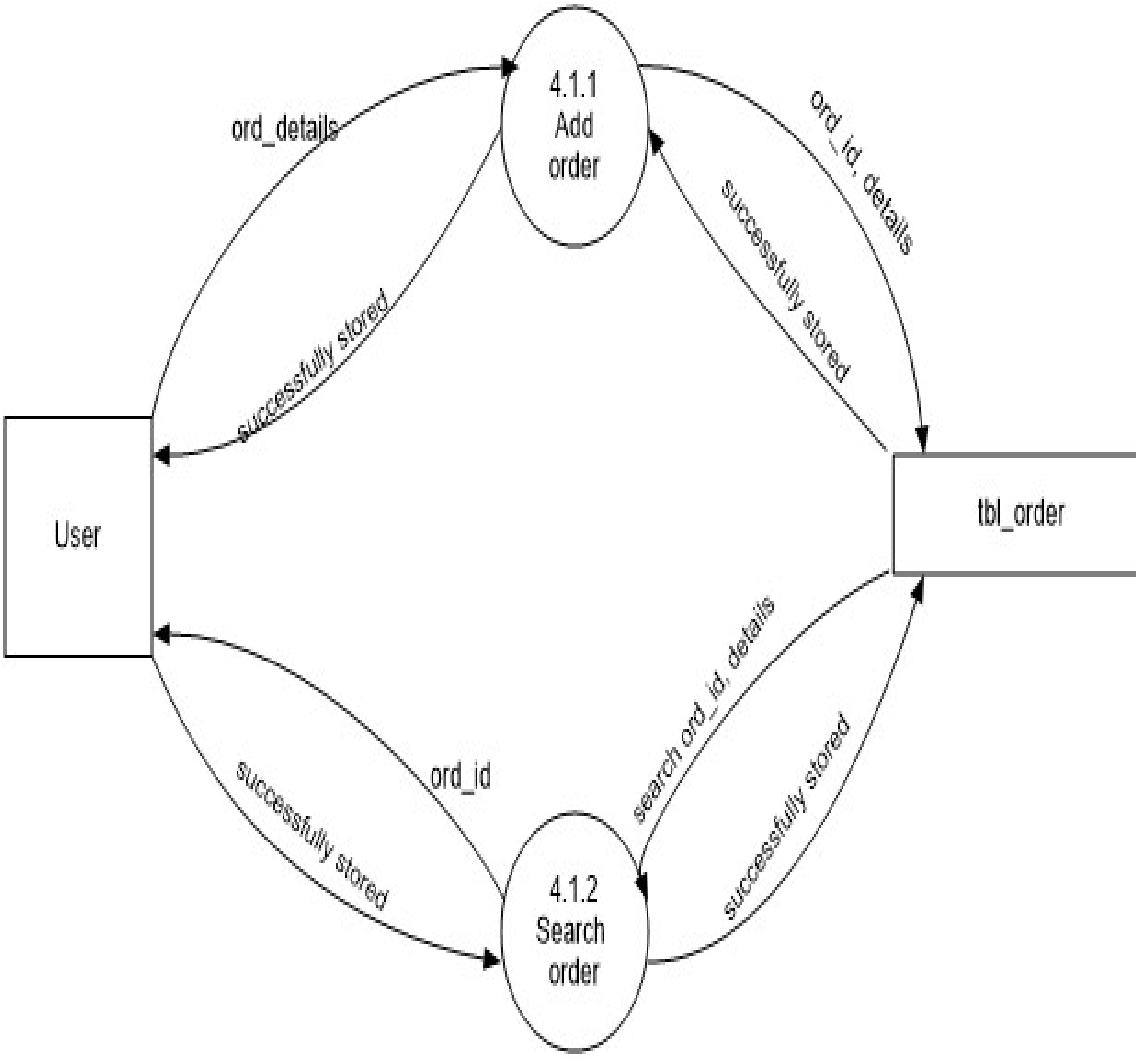
Level 3 DFD - showing canteen management



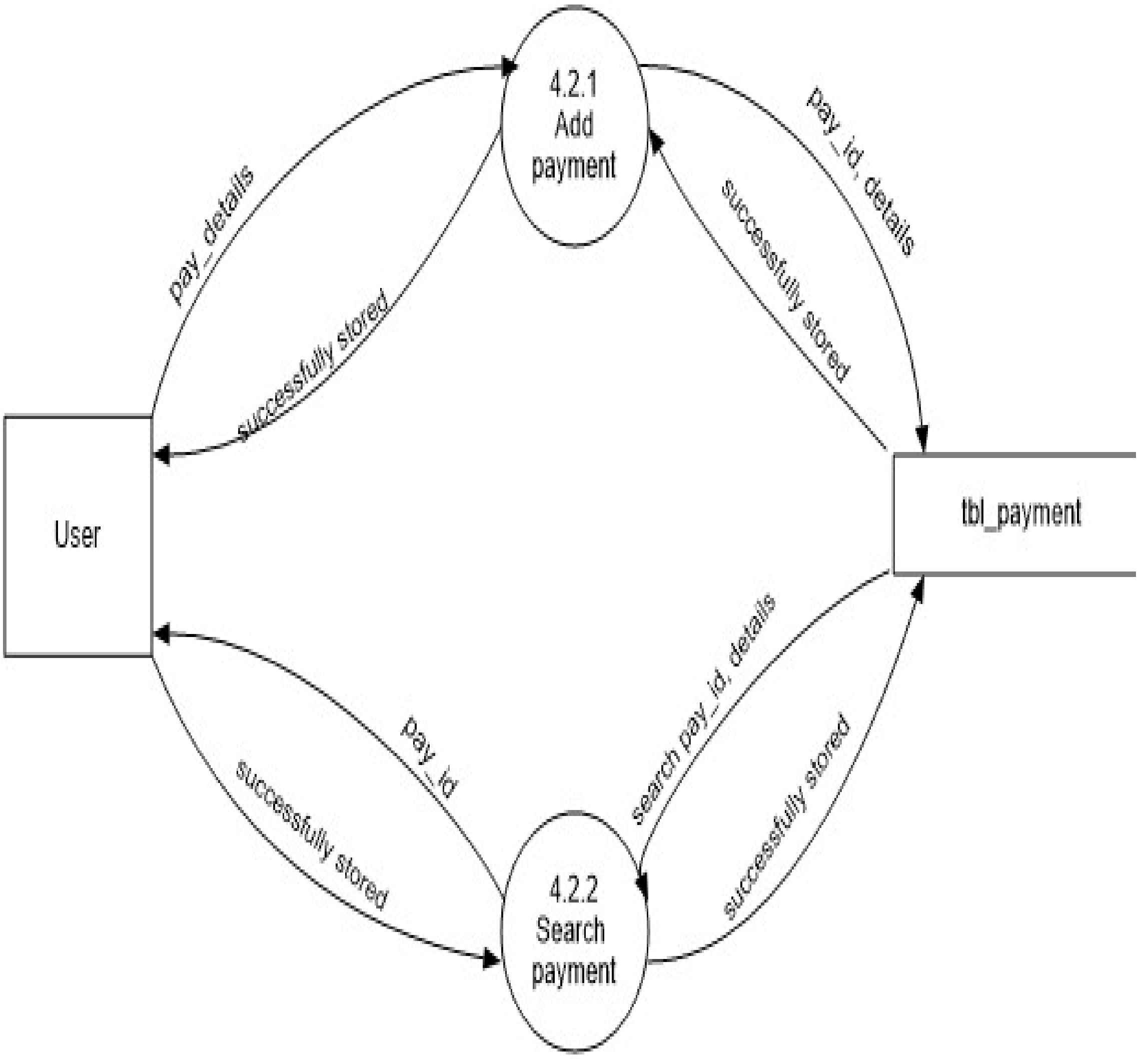
Level 3 DFD - showing student management



Level 3 DFD - showing order management

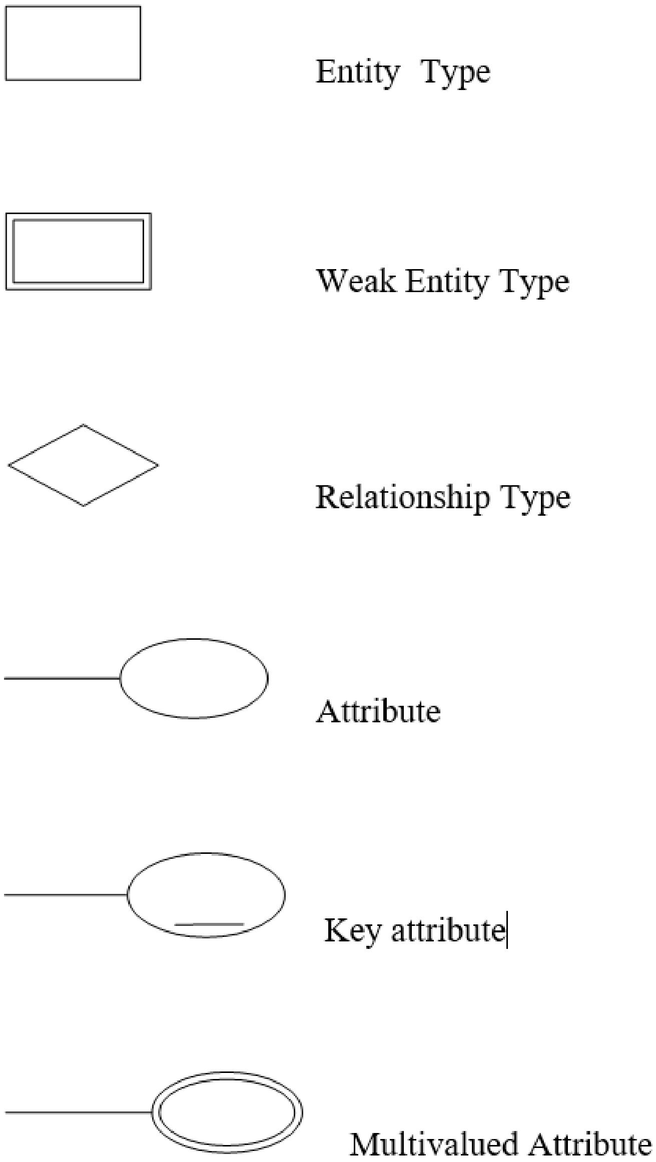


Level 3 DFD - showing payment management

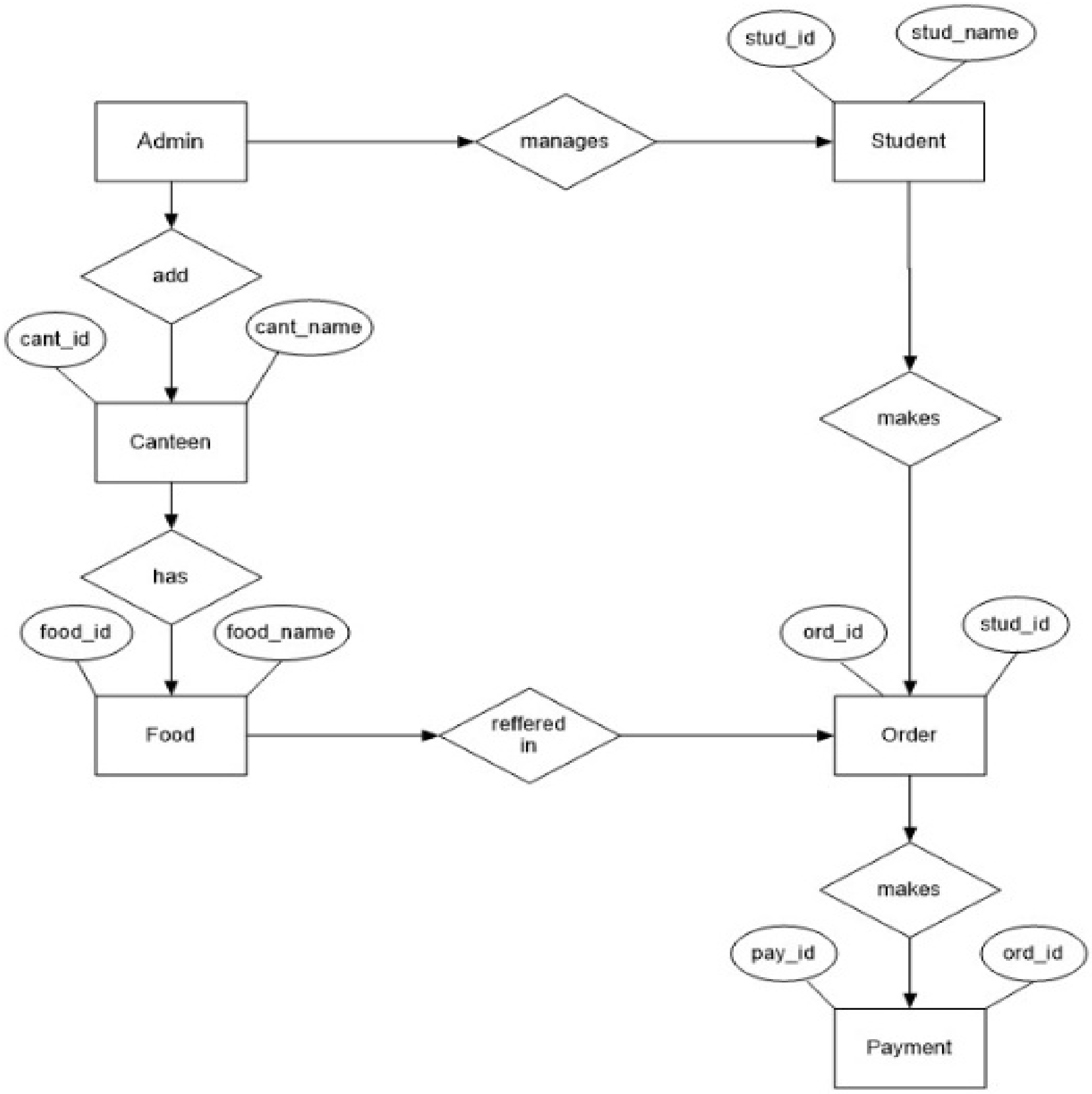


6.3 Entity Relationship Diagram

The ER model is a conceptual data model that views the real world as a construct of entities and associations or relationships between entities. A basic component of the model is the Entity-Relationship diagram, which is used to visually represent data objects. The ER modeling technique is frequently used for the conceptual design of database applications and many database applications and many database design tools employ its concepts.



ER Diagram Showing Canteen Automation System



Chapter 7

SYSTEM TESTING

7.1 White Box Testing

White box is a testing technique that examines the program structure and derives test data from the program logic/code. The other names of glass box testing are clear box testing, open box testing, logic driven testing or path driven testing or structural testing.

7.2 Black Box Testing

Black box testing, also known as Behavioural testing, is a software testing method in which the internal structure design/ implementation if the item being tested is not known to the tester. These tests can be functional or non-functional, though usually functional. This method is named so because the software program, in the eyes of the tester, is like a black box; inside which one cannot see.

7.3 Testing and validation

7.4 Testing

Testing is an import and stage in the software development life cycle. System testing is a critical element of a software quality assurance and represents the ultimate review of specification, design and coding. Importance of software testing and it simplification with software quality cannot be over me phased. Testing is one-way developers can validate the quality of a software product and verify that it fully meets the specification. During testing, the system is tested with a set of cases and checked whether the input of the program is performing as it is expected. The system tested and reviewed to ensure that the entire user requirement has being satisfied.

Testing was done throughout the system development at various stages since it is always a good practice to test the system at many different levels at various intervals that is subsystems, program modules as work progress and finally the system as a whole. If this is not done, then the poorly tested system can fail after installation. Testing is a very tedious and time-consuming job. For a test to be successful the tester should try and make the program file. Each test is designed with the intention of finding errors in the way system will process it. Though testing of a program doesn't guarantee the reliability of the system, it is done to assure that the system runs errors free.

The Testing process begins by developing a comprehensive plan to test the general functionality and special features on a variety of platform combinations. Strict quality control procedures are used. The Process verifies that the application meets the requirements specified in the system requirements document and is bug free. At the End of each testing day, the summary of completed and failed tests is prepared. And the Application is redeveloped and retested until every item is resolved. All the changes and retesting are tracked through spread sheets. Applications are not allowed to launch until all identified problem are fixed. Finally, a report is prepared at the end of testing to show exactly what was tested and to list the final outcomes. The software testing methodology is applied in four distinct phases:

- Unit Testing
- Integration Testing
- User Acceptance Testing
- Output Testing

7.4.1 Unit Testing

Developers typically do unit testing in order to trace out bugs in each module of the code. Unit testing is done in parallel with coding. It includes testing each function and procedures. Unit testing is also called as module testing. In module testing each module are tested for any possible logical error. They are also tested for specification to see if they are working as per the program should do and they are tested under various conditions. Each module is being tested thoroughly in order to discover pitfalls. Specification testing examines the specification what the program should do and how it should perform under various conditions. The testing will be done by entering data into different tables using forms. The data with less validation will be tested first. Whenever an error is encountered, an informative error message will be displayed which in forms user about the type of error. After the completion of form testing the program will be tested. The unit testing is done to identify

- No duplicate entries are present.
- To check whether it provide the required result

7.4.2 Integration Testing

Integration testing is any type of software testing that seeks to verify the interfaces between components against a software design. Software components may be integrated in an iterative way or all together (" big bang"). Normally the former is considered a better practice since it allows interface issues to be located more quickly and fixed. Integration testing works to expose defects in the interfaces and interaction between integrated components (modules). The task of the integration test is to check that components or software applications, interacts without error. Therefore, testing the data flow between 2 modules is integration testing.

7.4.3 Output Testing

After performing the validation testing, the next step is output testing of the proposed system since no system could be useful if it does not produce the required output in the specific format. The output generated or displayed by the system under consideration is tested asking the users about the format required by them. Here, the output is considered in two ways, one is on the screen and other is printed format. The output format on the screen is found to be correct as the format designed according to the user needs. For the hard copy also, the output comes out as specified by the user. Hence output testing does not result in any connection in the system.

7.5 Validation Check

The validation phase reveals the failures and the buds in the developed system. It will become to known about the practical difficulties the system faces when the operated in the true environment. Validation is the process of ensuring that user input is clean, correct, and useful. Typical validation tasks are:

- Has the user filled in all required fields?
- Has the user entered a valid email?
- Has the user entered text in a numeric field?

Form validation normally used to occur at the server, after the client had entered all the necessary data and then pressed the submit button. If the data entered by a client was incorrect or was simply missing, the server would have to send all the data back to the client and request that the form be resubmitted with correct information. This was really a lengthy process which used to put a lot of burden on the server. Most often,the purpose of validation is to ensure correct user input. Validation can be defined by many different methods and deployed in many different ways. Server-side validation is performed.

Chapter 8

CONCLUSION AND FUTURE SCOPE

8.1 Conclusion

It will overcome the issues related to traditional canteen management system and provide scale-able and reliable. It provides frequent ordering list for fast ordering.The development of Canteen Automation system involved many phases. The approach used is a top-down one concentrating on what first, then how and moving to successive levels of details.

The first phase started with a detailed study of the problems and prospects of ordering in Foods. In the course of this study, many problems were discovered to have hindered the effectiveness of the existing manual system. These problems, information needs and activities were documented and later used as the basis for system design, which immediately followed the first phase. The design phase was concerned primarily with the specification of the system elements in manner that best met the organization's business needs.

During this phase, strict adherence was made on proven software engineering principles and practices. To implement this design, a computer program was then written and tested.It is hoped that effective implementation of this software product would eliminate many problems discovered during systems investigation.

8.2 Future Scope

- All the orders placed by the students will be in digital format hence no chance of human error and tedious work.
- Canteen workers will not have to remember each and every order placed and this will make their task easy.
- Waiters don't have to manually calculate the amount of money to be paid by the customer after having food it is automatically done in the software.
- Loss or mishandling of data can be minimal with use of technology.
- Use of QR (Quick Response) Code is implemented to generate after order is placed. Admin or the canteen person need to scan the QR code to initiate the transaction and the amount will be deducted respectively.

Chapter 9

APPENDIX

9.1 Source Code

Python code

```
from django.shortcuts import render
from app1.models import *
from django.http import HttpResponse,JsonResponse
from random import random
from django.core.files.storage import FileSystemStorage
import random
from django.views.decorators.csrf import csrfexempt
from .models import*
import pyqrcode
import png
from pyqrcode import QRCode
import random
import requests
import cv2
@csrfexempt
def Student_reg(request) :
name = request.POST.get(”name”)
phone = request.POST.get(”phone”)
addr = request.POST.get(”addr”)
batch = request.POST.get(”batch”)
uname = request.POST.get(”usnm”)
pswrd = request.POST.get(”psw”)
print(name, phone, addr, batch, uname, pswrd)
response_data =
try :
ob = Student_table(Name = name,Phone = phone,Address = addr,Batch = batch,Username =
```

$uname, Pswrd = pswrd)$

$ob.save()$

$response_data['msg'] = ”yes”$

$except:$

$response_data['msg'] = ”no”$

$returnJsonResponse(response_data)$

$@csrf_exempt$

$defDelete_canteen(request):$

$cid = request.POST.get(”cid”)$

$response_data =$

$try:$

$ob = Canteen_table.objects.get(id = int(cid))$

$ob.delete()$

$response_data['msg'] = ”yes”$

$except:$

$response_data['msg'] = ”no”$

$returnJsonResponse(response_data)$

$@csrf_exempt$

$defUpdate_student(request):$

$cid = request.POST.get(”cid”)$

$name = request.POST.get(”name”)$

$phone = request.POST.get(”phone”)$

$addr = request.POST.get(”addr”)$

$batch = request.POST.get(”batch”)$

$uname = request.POST.get(”usnm”)$

$pswrd = request.POST.get(”psw”)$

$print(name, phone, addr, batch, uname, pswrd)$

$response_data =$

$try:$

$ob = Student_table.objects.get(id = int(cid))$

$ob.Name = name$

$ob.Batch = batch$

$ob.Phone = phone$

$ob.Address = addr$

$ob.Username = uname$

$ob.Pswrd = pswrd$

$ob.save()$

$response_data['msg'] = ”yes”$

$except :$

$response_data['msg'] = ”no”$

$returnJsonResponse(response_data)$

$@csrf_exempt$

$defCheckLogin(request) :$

$uname = request.POST.get(”uname”)$

$pswrd = request.POST.get(”pswrd”)$

$utype = request.POST.get(”utype”)$

$print(uname, pswrd)$

$if(utype == ”Canteen”) :$

$try :$

$ob = Canteen_table.objects.get(Username = uname, Pswrd = pswrd)$

$data = ”msg” : ”Canteen”$

$returnJsonResponse(data, safe = False)$

$except :$

$data = ”msg” : ”no”$

$returnJsonResponse(data, safe = False)$

$else :$

$try :$

$ob = Student_table.objects.get(Username = uname, Pswrd = pswrd)$

$data = ”msg” : ”Student”$

$returnJsonResponse(data, safe = False)$

$except :$

$data = ”msg” : ”no”$

$returnJsonResponse(data, safe = False)$

$@csrf_{e}xempt$

$defgetallstudents(request) :$

$resplist = []$

$respdata =$

$ob = Student_{t}able.objects.all()$

$resplist = []$

$respdata =$

$foriinob :$

$data =$

$data[”cid”] = i.id$

$data[”cname”] = i.Name$

$data[”oname”] = i.Batch$

$data[”phno”] = i.Phone$

$data[”addr”] = i.Address$

$data[”usernm”] = i.Username$

$data[”pswrd”] = i.Pswrd$

$resplist.append(data)$

$respdata[”data”] = resplist$

$print(respdata)$

$returnJsonResponse(respdata, safe = False)$

$@csrf_{e}xempt$

$defAdd_{f}ood(request) :$

$uname = request.POST.get(”uname”)$

$fname = request.POST.get(”fname”)$

$fdes = request.POST.get(”fdes”)$

$ftype = request.POST.get(”ftype”)$

$fprc = request.POST.get(”fprc”)$

$ftme = request.POST.get(”ftme”)$

$fdate = request.POST.get(”fdate”)$

$response_data =$

$try:$

$ob = Food_table.objects.get(Cusername = uname, Date = fdate, Fname = fname)$

$response_data[′msg′] = ”no”$

$except:$

$ob = Food_table(Cusername = uname, Date = fdate, Fname = fname, Finfo = fdes, Price =$

$fprc, Time = ftme, Ftype = ftype, Status = ”1”)$

$ob.save()$

$response_data[′msg′] = ”yes”$

$returnJsonResponse(response_data)$

$@csrf_exempt$

$defCanteen_view_food(request):$

$uname = request.POST.get(”uname”)$

$fdate = request.POST.get(”fdate”)$

$resplist = []$

$respdata =$

$ob = Food_table.objects.filter(Cusername = uname, Date = fdate)$

$resplist = []$

$respdata =$

$foriinob:$

$data =$

$data[”fid”] = i.id$

$data[”fname”] = i.Fname$

$data[”fdes”] = i.Finfo$

$data[”fprc”] = i.Price$

$data[”ftme”] = i.Time$

$data[”ftyp”] = i.Ftype$

$if(i.Status == ”0”):$

$fsts = ”Notavailable”$

$else:$

$fsts = ”Available”$

$data[”fsts”] = fsts$

$resplist.append(data)$

$respdata[”data”] = resplist$

$print(respdata)$

$returnJsonResponse(respdata, safe = False)$

$@csrf_{e}xempt$

$defDisable_{f}ood(request):$

$fid = request.POST.get(”fid”)$

$response_{d}ata =$

$try:$

$ob = Food_{t}able.objects.get(id = int(fid))$

$fsts = ob.Status$

$if(fsts == ”0”):$

$ob.Status = ”1”$

$ob.save()$

$else:$

$ob.Status = ”0”$

$ob.save()$

$response_{d}ata['msg'] = ”yes”$

$except:$

$response_{d}ata['msg'] = ”no”$

$returnJsonResponse(response_{d}ata)$

$@csrf_exempt$

$defDelete_food(request):$

$fid = request.POST.get(”fid”)$

$response_data =$

$try:$

$ob = Food_table.objects.get(id = int(fid))$

$ob.delete()$

$response_data['msg'] = ”yes”$

$except:$

$response_data['msg'] = ”no”$

$returnJsonResponse(response_data)$

$@csrf_exempt$

$defFood2cart(request):$

$fid = request.POST.get(”fid”)$

$sname = request.POST.get(”sname”)$

$fqty = request.POST.get(”fqty”)$

$today = date.today()$

$d1 = today.strftime(”print(”d1 = ”, d1)$

$response_data =$

$try:$

$ob = Bucket_table.objects.get(Foodid = fid, Sname = sname, Date = d1)$

$response_data['msg'] = ”exist”$

$except:$

$ob = Food_table.objects.get(id = int(fid))$

$fname = ob.Fname$

$fprc = ob.Price$

$cname = ob.Cusername$

$total = int(fqty) * int(fprc)$

$ob1 = Bucket_table(Foodid = fid, Fname = fname, Quantity = fqty, Price = total, Date =$

$d1, Sname = sname, Cusernm = cname)$

$ob1.save()$

$response_{d}ata['msg'] = "yes"$

$returnJsonResponse(response_{d}ata)$

$@csrf_{e}xempt$

$defGetmybucket(request):$

$sname = request.POST.get("sname")$

$print(sname)$

$resplist = []$

$respdata =$

$ob = Bucket_{t}able.objects.filter(Sname = sname)$

$cantlist = []$

$foriinob:$

$cname = i.Cusernm$

$if(cnamenotincantlist):$

$cantlist.append(cname)$

$print("canteenlist ==> ", cantlist)$

$if(len(cantlist)! = 0):$

$forkincantlist:$

$obx = Bucket_{t}able.objects.filter(Cusernm = k, Sname = sname)$

$obc = Canteen_{t}able.objects.get(Username = k)$

$Canteen = obc.Cname$

$items = ""$

$prices = ""$

$total = 0$

$qty = ""$

$forjinobx:$

$items+ = j.Fname + ","$

$prices+ = j.Price + ","$

$total+=int(j.Price)$

$qty+=j.Quantity+”,”$

$resplist.append(data)$

$print(resplist)$

$respdata[”data”]=resplist$

$print(respdata)$

$returnJsonResponse(respdata,safe=False)$

$fromdatetimeimportdatetime$

$@csrf_exempt$

$defConfirm_order(request):$

$cname=request.POST.get(”cname”)$

$sname=request.POST.get(”uname”)$

$today=date.today()$

$now=datetime.now()$

$d1=today.strftime(”d2=now.strftime(”print(”d1=”,d1)$

$response_data=$

$print(sname)$

$print(cname)$

$try:$

$obc=Canteen_table.objects.get(Username=cname)$

$canteen=obc.Cname$

$obx=Bucket_table.objects.filter(Cusernm=cname,Sname=sname)$

$resptxt=”====================”+canteen+”=================$

$”$

$resptxt+=”Date:”+d1+””$

$resptxt+=”ItemQuantityprice”$

$items=””$

$prices=””$

$total=0$

$qty = ””$

$forjinobx :$

$tval = ””$

$items+ = j.Fname+”,”$

$tval+ = j.Fname+””$

$prices+ = j.Price+”,”$

$total+ = int(j.Price)$

$qty+ = j.Quantity+”,”$

$tval+ = j.Quantity+””$

$tval+ = j.Price+””$

$resptxt+ = tval$

$j.delete()$

$resptxt+ = ”GrandTotal : ”+str(total)$

$print(resptxt)$

$number = random.randint(111111,999999)$

$print(number)$

$StringwhichrepresentstheQRcode$

$scode = str(number)$

$GenerateQRcode$

$url = pyqrcode.create(scode)$

$qrpath = ”qrimages/”+sname+d2+”.png”$

$Createandsavethepngfilenaming”myqr.png”$

$url.png(qrpath,scale = 6)$

$obs = Purchase_table(Cusernm = cname,Fooditems = items[: -1],Quantity = qty[: -1],Price =$

$prices[: -1],Total = str(total),$

$Date = d1,Sname = sname,Secretid = scode,Qrpath = qrpath)$

$obs.save()$

$print(”saved”)$

$response_data['msg'] = ”yes”$

$response_{d}ata[”fname”] = ”Bill” + d2 + ”.txt”$

$response_{d}ata[”bill”] = resptxt$

$exceptExceptionase :$

$print(e)$

$response_{d}ata['msg'] = ”no”$

$returnJsonResponse(response_{d}ata)$

$@csrf_{e}xempt$

$defStudent_{m}y_{o}rders(request) :$

$sname = request.POST.get(”sname”)$

$print(sname)$

$resplist = []$

$respdata =$

$ob = Purchase_{t}able.objects.filter(Sname = sname, Status = ”Orderd”)$

$forjinob :$

$fname = j.Fooditems$

$fqty = j.Quantity$

$ftotal = j.Total$

$date = j.Date$

$impath = j.Qrpath$

$cname = j.Cusernm$

$obc = Canteen_{t}able.objects.get(Username = cname)$

$cantname = obc.Cname$

$status = j.Status$

$withopen(impath, ”rb”)asimg_{f}ile :$

$b64_{s}tring = base64.b64encode(img_{f}ile.read())$

$data[”imgstr”] = b64_{s}tring.decode(”utf - 8”)$

$resplist.append(data)$

$print(resplist)$

$respdata[”data”] = resplist$

$print(respdata)$

$returnJsonResponse(respdata, safe = False)$

$@csrf_exempt$

$defStudent_my_history(request):$

$sname = request.POST.get(''sname'')$

$print(sname)$

$resplist = []$

$respdata =$

$ob = Purchase_table.objects.filter(Sname = sname)$

$forjinob:$

$fname = j.Fooditems$

$fqty = j.Quantity$

$ftotal = j.Total$

$date = j.Date$

$impath = j.Qrpath$

$cname = j.Cusernm$

$obc = Canteen_table.objects.get(Username = cname)$

$cantname = obc.Cname$

$status = j.Status$

$data =$

$data[''fname''] = fname$

$data[''ftotal''] = ftotal$

$data[''fdate''] = date$

$data[''cname''] = cantname$

$data[''status''] = status$

$resplist.append(data)$

$print(resplist)$

$respdata[''data''] = resplist$

$print(respdata)$

$returnJsonResponse(respdata,safe=False)$

$@csrf_exempt$

$defCanteen_my_orders(request):$

$sname=request.POST.get(”sname”)$

$print(sname)$

$resplist=[]$

$respdata=$

$ob=Purchase_table.objects.filter(Cusernm=sname).order_by('-Status')$

$forjinob:$

$fname=j.Fooditems$

$fqty=j.Quantity$

$ftotal=j.Total$

$date=j.Date$

$impath=j.Qrpath$

$cname=j.Sname$

$obc=Student_table.objects.get(Username=cname)$

$stname=obc.Name$

$status=j.Status$

$data[”fqty”]=fqty$

$data[”ftotal”]=ftotal$

$data[”fdate”]=date$

$data[”status”]=status$

$resplist.append(data)$

$print(resplist)$

$respdata[”data”]=resplist$

$print(respdata)$

$returnJsonResponse(respdata,safe=False)$

$@csrf_exempt$

$defCanteen_my_orders_by_date(request):$

$sname = request.POST.get(”sname”)$

$date = request.POST.get(”date”)$

$print(sname)$

$print(date)$

$resplist = []$

$respdata =$

$ob = Purchase_table.objects.filter(Cusernm = sname, Date = date)$

$forjinob :$

$fname = j.Fooditems$

$fqty = j.Quantity$

$ftotal = j.Total$

$date = j.Date$

$impath = j.Qrpath$

$cname = j.Sname$

$obc = Student_table.objects.get(Username = cname)$

$stname = obc.Name$

$status = j.Status$

$data[”fname”] = fname$

$data[”fqty”] = fqty$

$data[”ftotal”] = ftotal$

$data[”fdate”] = date$

$data[”cname”] = stname$

$resplist.append(data)$

$print(resplist)$

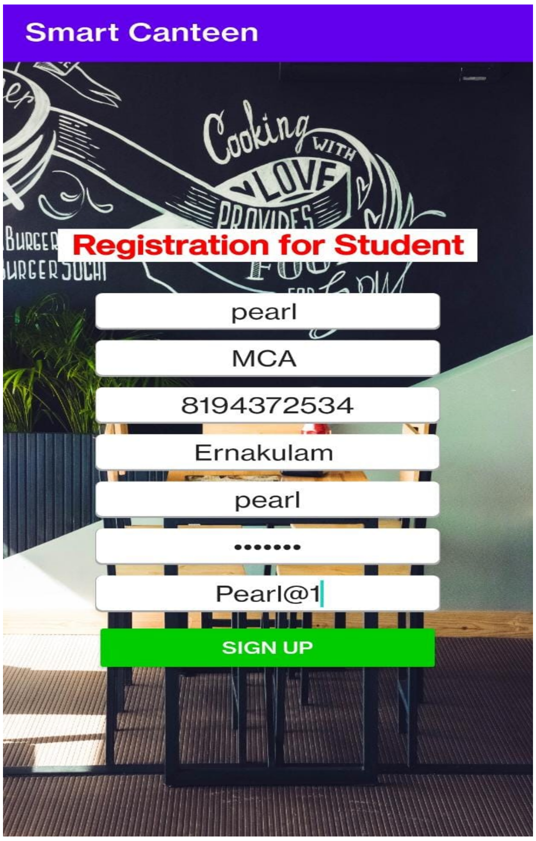
$respdata[”data”] = resplist$

$print(respdata)$

$returnJsonResponse(respdata, safe = False)$

9.2 Screenshot

Student Registration Page



Smart Canteen

Registration for Student

pearl

MCA

8194372534

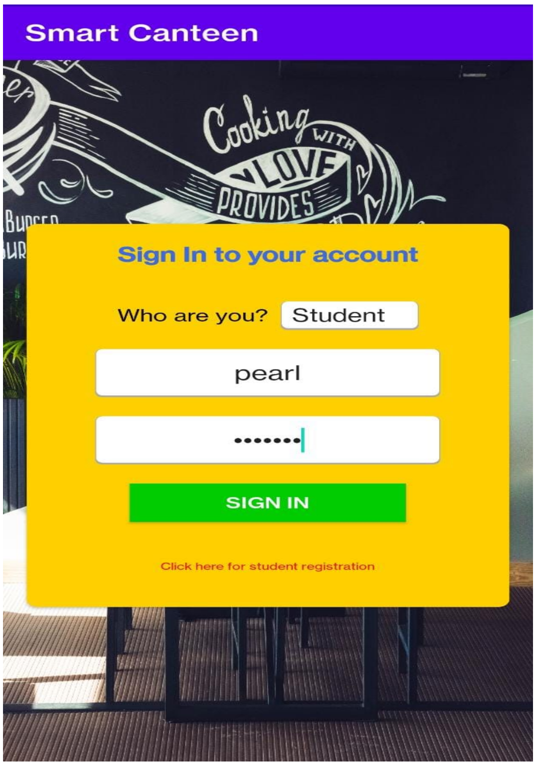
Ernakulam

pearl

•••••••

Pearl@1

Student Login Page



Smart Canteen

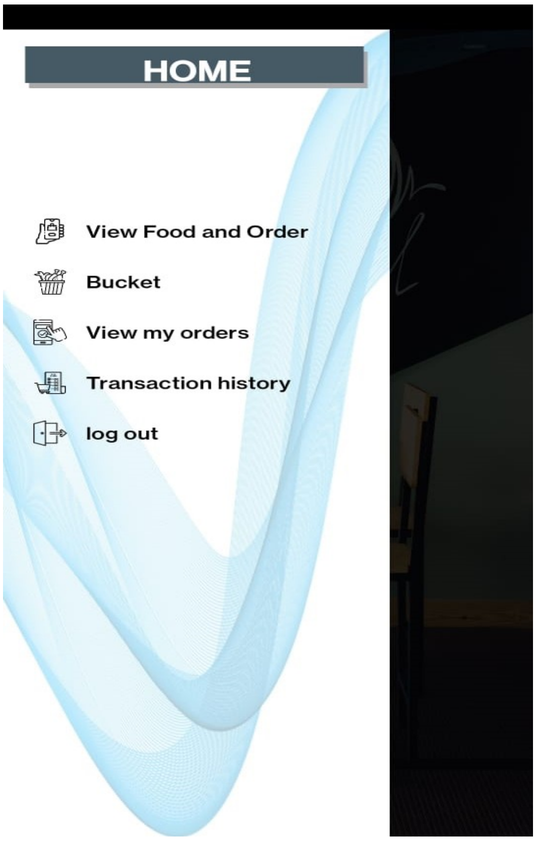
Sign In to your account

Who are you? Student

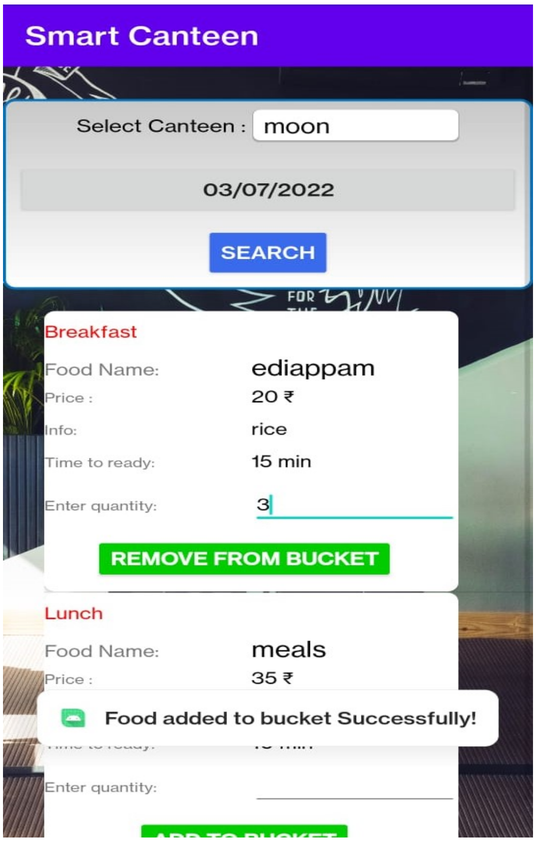
pearl

•••••••

Student Home Page



Student viewing menu by selecting the date and ordering the food



Smart Canteen

Select Canteen : moon

03/07/2022

SEARCH

Breakfast

Food Name: ediappam

Price : 20 ₹

Info: rice

Time to ready: 15 min

Enter quantity: 3

REMOVE FROM BUCKET

Lunch

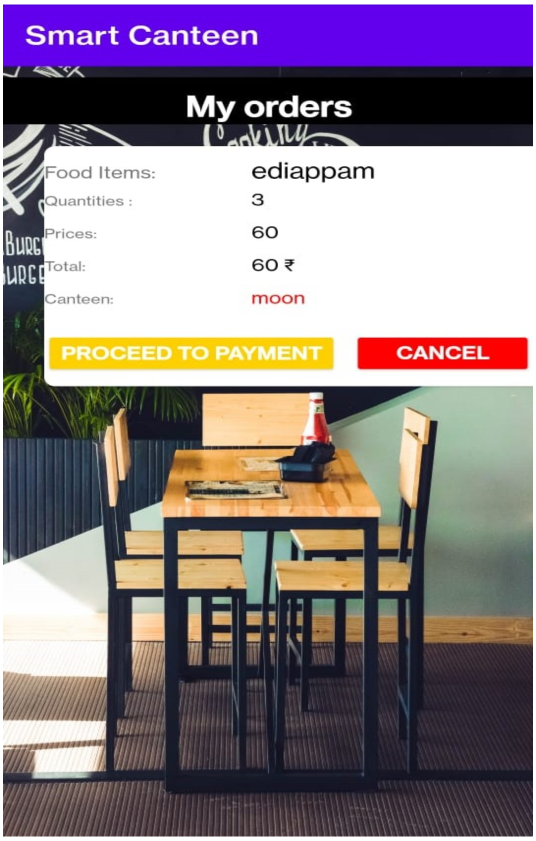
Food Name: meals

Price : 35 ₹

Food added to bucket Successfully!

Enter quantity:

Student viewing order from cart



Smart Canteen

My orders

Food Items: ediappam

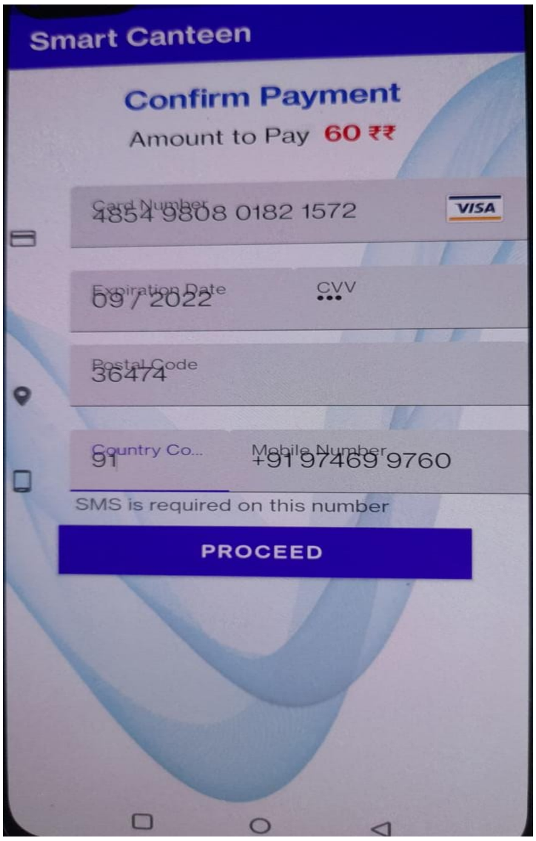
Quantities : 3

Prices: 60

Total: 60 ₹

Canteen: moon

Student payment method



Smart Canteen

Confirm Payment

Amount to Pay 60 ₹₹

Card Number: 4854 9808 0182 1572 VISA

Expiration Date: 09 / 2022

CVV: •••

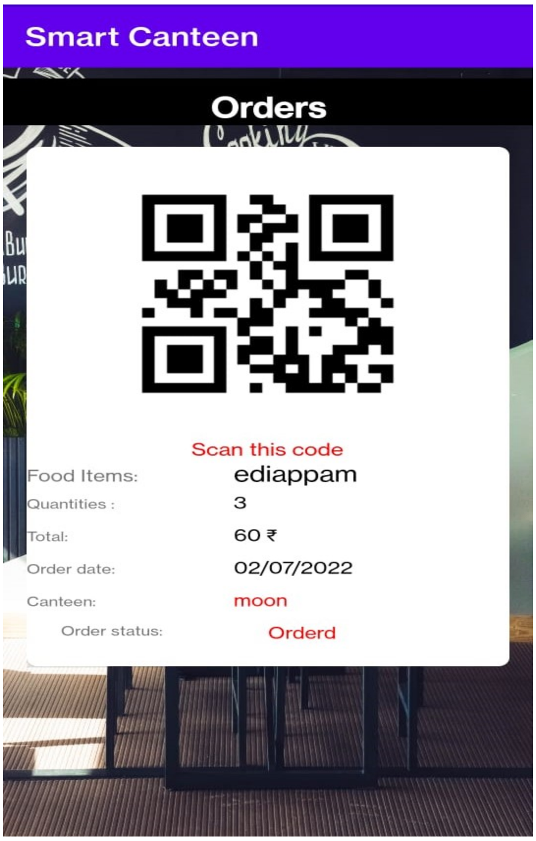
Postal Code: 36474

Country Co...: 91

Mobile Number: +91 97469 9760

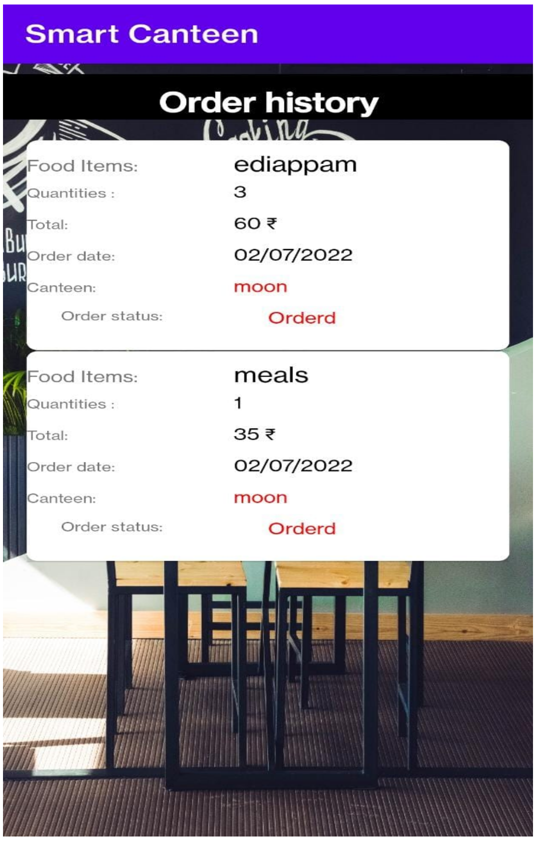
SMS is required on this number

Student viewing order



Food Items: ediappam
Quantities : 3
Total: 60 ₹
Order date: 02/07/2022
Canteen: moon
Order status: Orderd

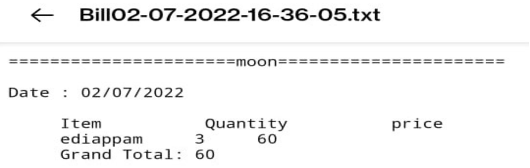
Student transaction history



Food Items: ediappam
Quantities : 3
Total: 60 ₹
Order date: 02/07/2022
Canteen: moon
Order status: Orderd

Food Items: meals
Quantities : 1
Total: 35 ₹
Order date: 02/07/2022
Canteen: moon
Order status: Orderd

Bill generation



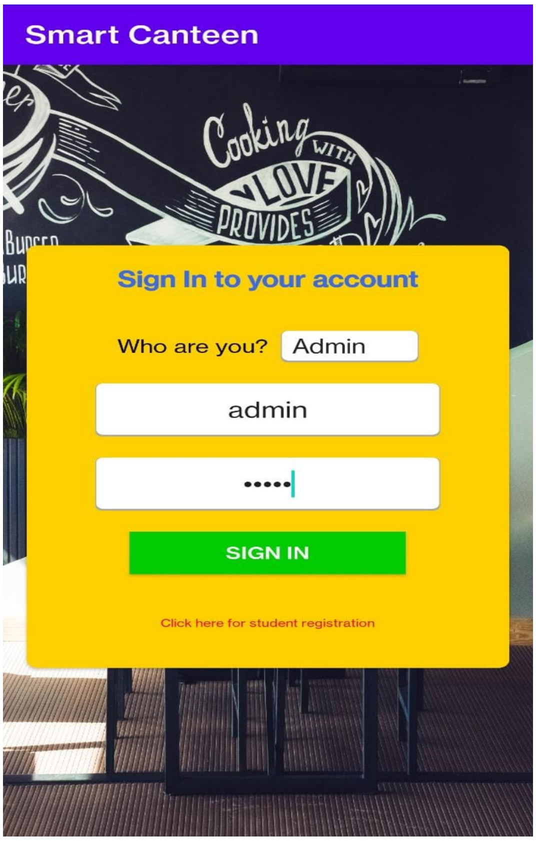
← Bill02-07-2022-16-36-05.txt

```
======================moon======================

Date : 02/07/2022

     Item          Quantity          price
     ediappam     3     60
     Grand Total: 60
```

Admin login page



Smart Canteen

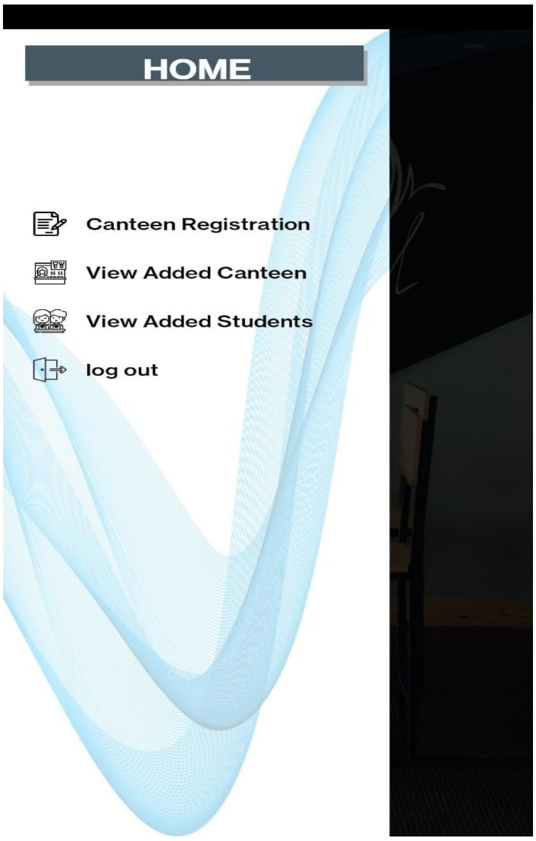
Sign In to your account

Who are you? Admin

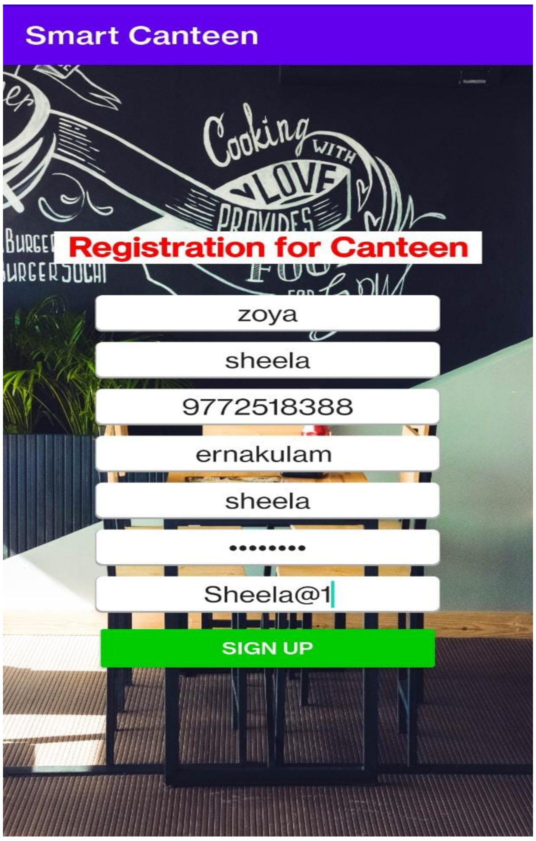
admin

•••••

Admin home page



Admin registering for canteen



Smart Canteen

Registration for Canteen

zoya

sheela

9772518388

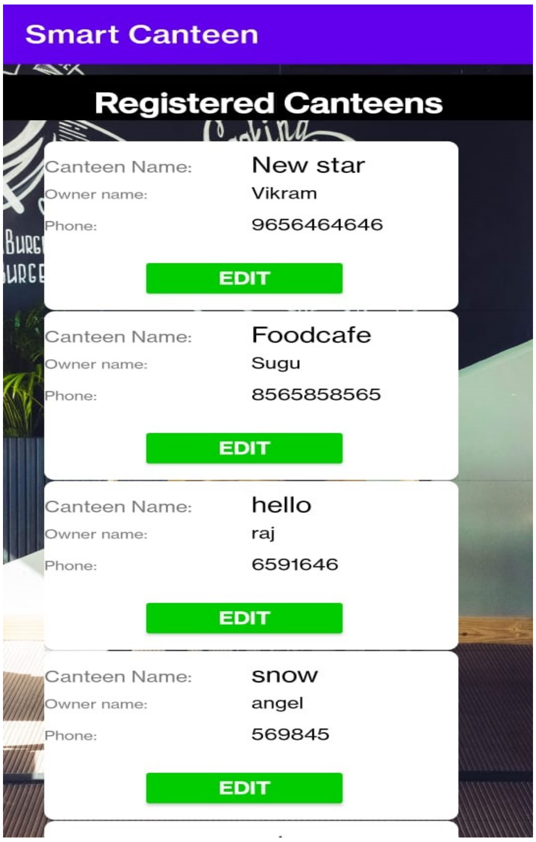
ernakulam

sheela

••••••••

Sheela@1

Admin viewing registered canteens



Smart Canteen

Registered Canteens

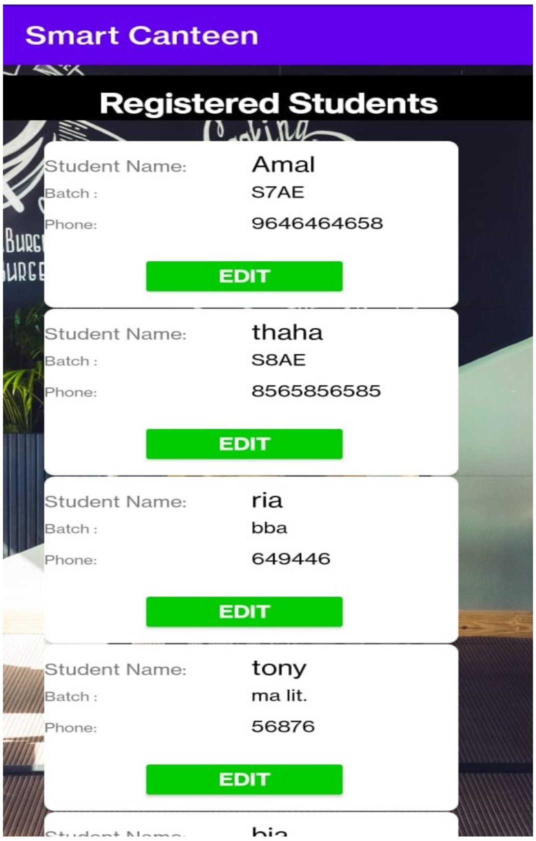
Canteen Name:	New star
Owner name:	Vikram
Phone:	9656464646

Canteen Name:	Foodcafe
Owner name:	Sugu
Phone:	8565858565

Canteen Name:	hello
Owner name:	raj
Phone:	6591646

Canteen Name:	snow
Owner name:	angel
Phone:	569845

Admin viewing registered students



Smart Canteen

Registered Students

Student Name: Amal
Batch : S7AE
Phone: 9646464658

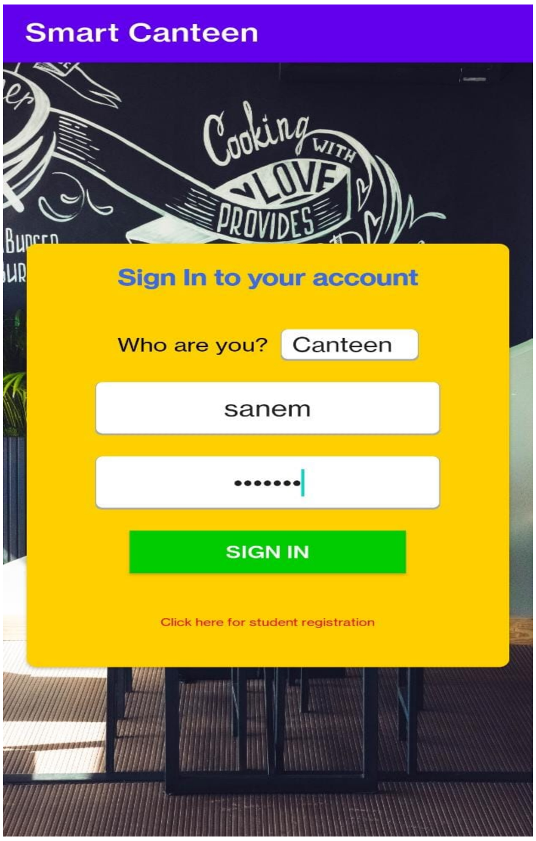
Student Name: thaha
Batch : S8AE
Phone: 8565856585

Student Name: ria
Batch : bba
Phone: 649446

Student Name: tony
Batch : ma lit.
Phone: 56876

Student Name: bia

Canteen login page



Smart Canteen

Sign In to your account

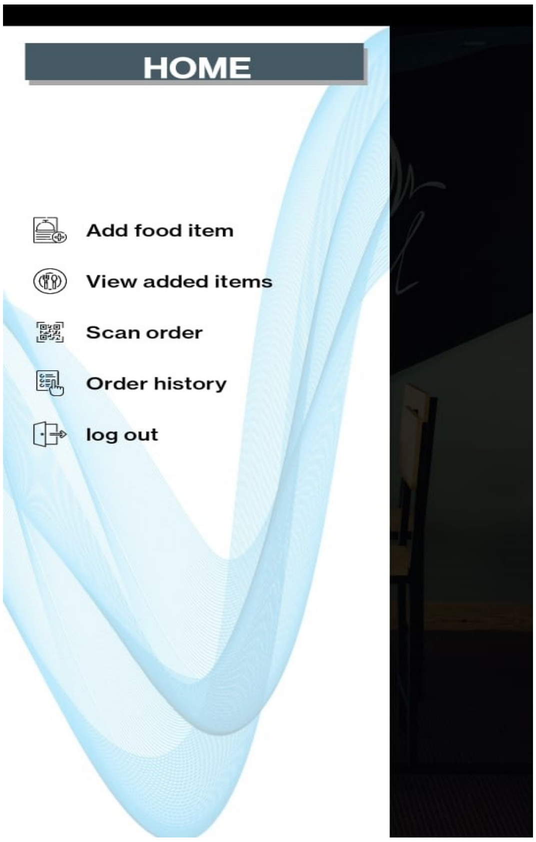
Who are you? Canteen

sanem

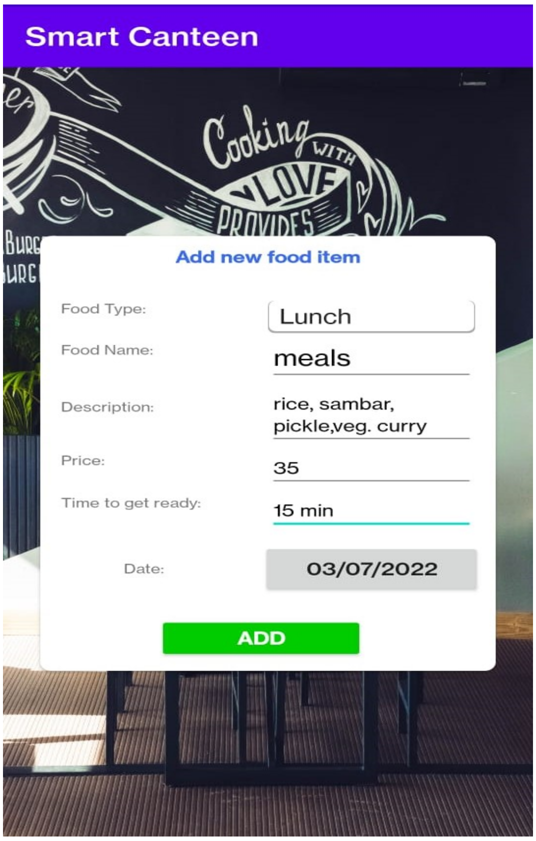
•••••••

SIGN IN
Click here for student registration
Cooking WITH LOVE PROVIDES

Canteen home page



Canteen adding food



Add new food item

Food Type: Lunch

Food Name: meals

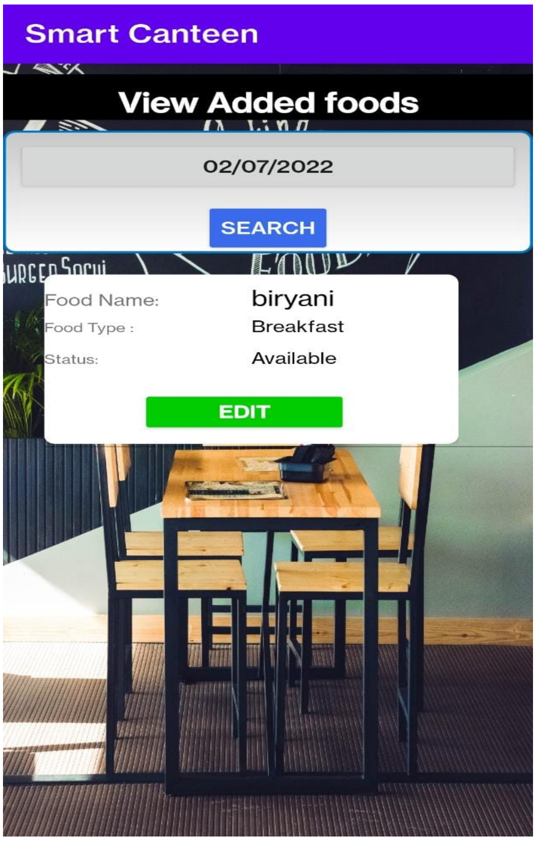
Description: rice, sambar, pickle,veg. curry

Price: 35

Time to get ready: 15 min

Date: 03/07/2022

Canteen viewing added food items



View Added foods

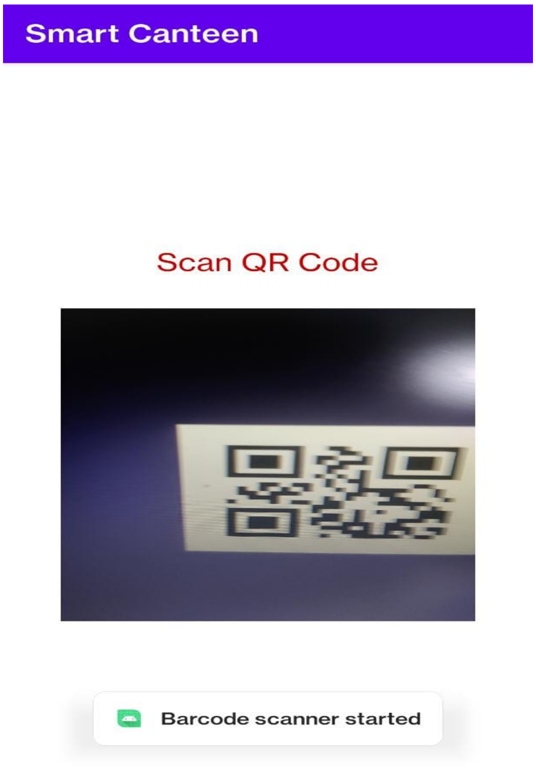
02/07/2022

Food Name: biryani

Food Type : Breakfast

Status: Available

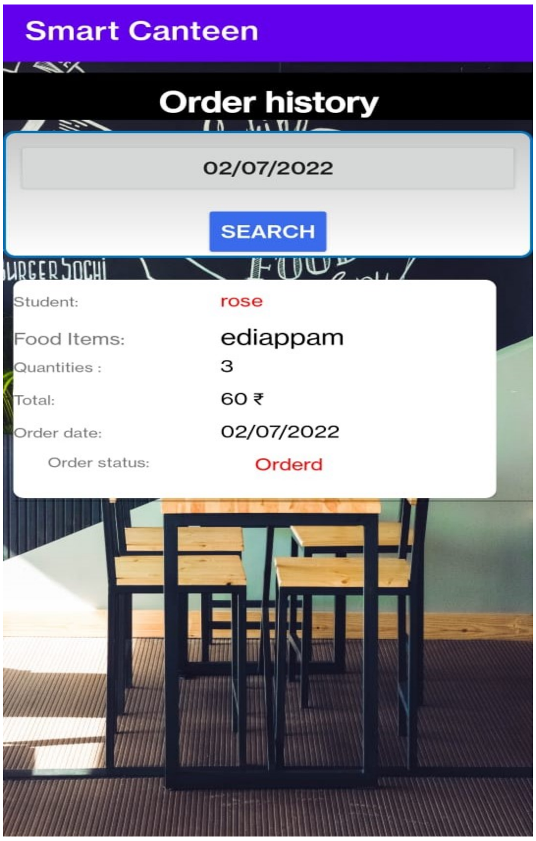
Canteen scanning QR



Scan QR Code

Barcode scanner started

Canteen viewing order history



Order history

02/07/2022

Student: rose

Food Items: ediappam

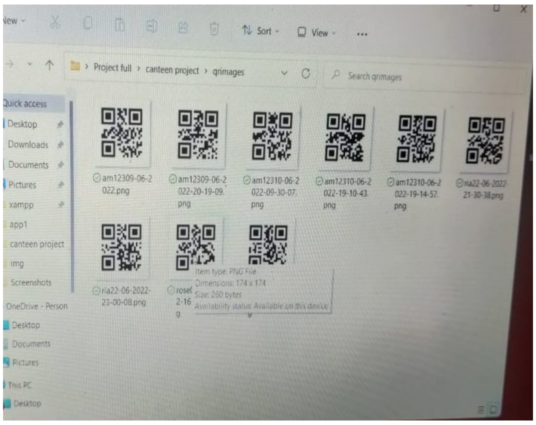
Quantities : 3

Total: 60 ₹

Order date: 02/07/2022

Order status: Orderd

QR generated for canteen


New
Sort
View
Project full › canteen project › qrimages
Search qrimages
Quick access
Desktop
Downloads
Documents
Pictures
xampp
app1
canteen project
img
Screenshots
OneDrive - Person
Desktop
Documents
Pictures
This PC
Desktop
am12309-06-2022.png
am12309-06-2022-20-19-09.png
am12310-06-2022-09-30-07.png
am12310-06-2022-19-10-43.png
am12310-06-2022-19-14-57.png
ria22-06-2022-21-30-38.png
ria22-06-2022-23-00-08.png
rose
2-16
g
Item type: PNG File
Dimensions: 174 x 174
Size: 260 bytes
Availability status: Available on this device

Chapter 10

REFERENCES

[1] Misbah Dalal, Zaid Barmare, Zehra Rizvi, Dr Ashfaq Shaikh. ”ANDROID BASED CANTEEN MANAGEMENT SYSTEM”. International Research Journal of Engineering and Technology (IRJET) Volume: 08 Issue: Apr 2021.

[2] Mrs.A.Gowthami,Ms.T.Banupriya,Ms.E.Vadivukkarasi. ”MOBILE APPLICATION FOR CANTEEN AUTOMATION SYSTEM USING ANDROID”. International Journal of Advanced Research in Computer Engineering Technology (IJARCET) Volume 9,Issue 3,March 2020.

[3] Sanil Sharma, Pranchal Jain, Rinshi Jain Roshni Gupta. ”Canteen Automation System”. GYAN GANGA INSTITUTE OF TECHNOLOGY SCIENCES,(DEC-2019).

[4] Monik Shah, Shalin Shah, Mohd Danish Shaikh, Kaustubh Tiwari. ”Canteen Automation System”. International Research Journal of Engineering and Technology (IRJET) Volume: 05 Issue: Jan-2019

[5] SUBHADEEP DAN, SHUBHAM SAURABH, ANKIT KUMAR MISHRA. ”Automated Canteen Management System”. BIRLA INSTITUTE OF TECHNOLOGY, Volume: 07 Issue: March 2021.

[6] Bisong-Abang Anthonia Obanghe. ”Automated Canteen Ordering system”. Baze university, Abuja [December 2020]