Small Scale Restaurant Management with Custom Scheduling Algorithm

Anchit Agarwal - 19BCE2279, Prithish Samanta - 19BCE2261, Ritvik Kohli - 19BCE2223
Vellore Institute of Technology (VIT), Vellore

Abstract

The goal of this project is to manage a restaurant database. Over the course of this project we are going to create a website that will make it easier for consumers to order and for waiters to serve. To collect and extract data records, we will employ the concept of a relational database management system. The goal of the project is to ensure that a restaurant runs well during peak rush hours. This will speed up the delivery of meals from the kitchen to the customers' tables. To implement the above idea we have used HTML, MySQL, TAILBLOCKS FLASK, etc.

Keywords: relational database management system, MySQL, FLASK, front end, back end, custom scheduling algorithm, round robin scheduling algorithm, dynamic,
Javascript

1. Introduction

About the problem. Whenever we go to a restaurant, we must always wait until the waitress with the menu card comes to us, and then we can only give him our order. It takes a lot of time and can take over 10 minutes sometimes. This
lengthy, tedious process can both irritate customers and damage the reputation of the restaurant that no restaurant owner wants to face. Whenever we go to a restaurant, we must always wait until the waitress with the menu card comes

[2]Fully documented templates are available in the elsarticle package on CTAN.

to us, and then we can only give him our order. This process takes a lot of time and may take more than 10 minutes. This long and tedious process can anger
the customers and also spoil the restaurant's reputation, which no restaurant owner wants to face.

Need/motivation. Whenever we go to a restaurant, we must always wait until the waitress with the menu card comes to us, and then we can only give him our
order. This process takes a lot of time and may take more than 10 minutes. This

lengthy, tedious process can both irritate customers and damage the reputation of the restaurant that no restaurant owner wants to face. DBMS facilities can solve this issue easily with our website and can also provide a pleasant meal for customers.

Objectives. Our objective is to make an easy and aesthetic interface which will hide the complexity of database side programming so that anyone can use it without learning it from anyone or any prior training.

- Our website's main objective is to make sure that the customers or users are not made to wait in a restaurant for a long time just for ordering food.
- Using our website he or she can give their orders smoothly and also efficiently. The customer's responses will be prompted and updated quickly through this website and the chefs/ waiters will be able to serve our customers properly and create a wonderful experience for them.
- One of our objectives is to normalize the database created to BCNF normalized form. This helps to reduce the redundancy from the database.
- Our customised algorithm will take in inputs of burst times of all the groups and calculate waiting time. The difference which this algorithm makes in that it assigns priority in the most optimised way possible. We will display the waiting times.
- With our website, waiters will have to just give order and not take it as it is happening in an automated way, saving waiters time and making the process smoother.

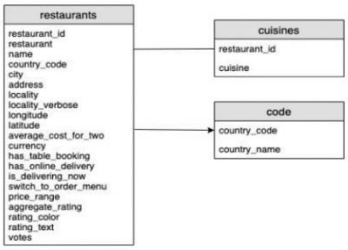
Possible techniques for proposed system.

- For front end we can use bootstrap, tailwind.css or React JS as we will be produce a fully mobile responsive website as well.
- For the back end i.e. inter connection and creation of routes, connection of our web application interface to database, and importing our custom algorithms we are Flask, node JS or nest JS can be used.
- For the performing queries and doing CRUD operations, we can use SQL server, Mongo DB, Mongose or Go.
- For automatic assigning of chefs and waiters, we can use any scheduling algorithm like SJF, Priority, round robin, etc but not FCFS as it is already used and inefficient.
- For the crowd management functionality, we can use different permutation and combinations of scheduling algorithms or we can make our own.

2. Literature Review

NAME, AUTHOR AND YEAR	METHODOLOGY	ADVANTAGES	DISADVANTAGES
L. Almukhaizeem, N. Almatar, M. I. Sarfraz and M. Sarfraz, "An Elegant and Efficient Database Design for Home-Based Restaurants," 2018	1. LOGICAL DESIGN SYSTEM INPUT: ● Products (kind, size, and price) that are necessary for the kitchen to prepare the order must be included in the system inputs. ● Automobile drivers (name and the location of the area that he should deliver the order to). ● Workers (employee name, civil ID, phone, nationality, address, employing date, basic salary, residence). ● The client (address, phone number, order, name). SYSTEM OUTPUT: ● Inform the employee at the call centre whether the product is available or not. ● Generate reports for end-users about the order (report of new orders on the kitchen screen, report of delivered order on call centre screen), about the company's returns (reports of revenues, profits, and expenses), about employees (reports of employee salaries to accounting department) to be displayed to the business owner. ● Include information on the final price, client address, and order status. 2. ER DIAGRAM - Starting with the "Log" table, which is linked to the "User" table via the User ID. The user table will link two primary tables, the "Employee" table and the "Order" table. The Employee table will identify the sort of employee (Call Center, Bakery, Accountant, Manager), which will be specified by the Employee ID as a primary key, and will also connect with two other tables, "Payroll" and "Delivery area." The payroll table will keep track of all the employee's financial details. The Delivery area table will serve as a link between the employee table and the "Invoice" table 3. SYSTEM IMPLEMENTATION The system was created from the ground up, and the final implementation was capable of completely capturing and implementing all necessary criteria for a home-based restaurant, as indicated in Table IV. MYSQL was the underlying database technology utilised to create a system that met all of the system criteria. MYSQL database server is capable of storing huge amounts of data as well as handling queries for a high number of users and orders.	1. This study suggested and built a database centric system architecture for home-based restaurants that allows all important stakeholders such as company owners, accountants, culinary staff, contact centre personnel, and delivery drivers to add, alter, or change restaurant needs in an automated manner. 2. The system may be a strategic tool since it helps a company to preserve the benefits of its existing system while it develops and evolves into other branches. 3. As the limits of a paper-based system become increasingly evident, the benefits of such a system become even more apparent.	1. In terms of hazards, there is fear that the developer's skills will not be sufficient to effectively fulfil the necessary requirements. 2. There is also worry that transitioning from a paper-based system to a computerised one would result in a loss of security for the system. 3. There are no standards for testing all the operations given by the database.

	The Windows programme enables various departments to connect to the database, therefore facilitating all procedures. The quality of the system that was developed and implemented has been favourably appraised by the end users.		
T. Fujita, H. Shimada and K. Sato, "Self-ordering system of restaurants for considering allergy information," 2014	1. Based on the criteria listed above, we seek a system that launches an application as a customer enters a restaurant to allow her to place orders. The usage flow is depicted in - NFC tags are placed on the tables by the eatery.- The customers are led to their assigned table.- They use their smartphone to activate the NFC tags. The programme that must be installed and activated (If it is not present). - Customers use the programme to place their orders. When consumers visit the restaurant, the NFC tag key is rewritten. 2. After the programme has been started, it needs to authenticate with the server to ensure that the customer is indeed inside the restaurant. This may be accomplished in a variety of methods, the most prominent of which are: Using GPS and an authentication key that is only available within the restaurant. 3. This programme, which rewrites NFC tag data when customers leave a table, requires the ability to read/write NFC tags and update the server with the most recent tag data. It reads NFC tag IDs, rewrites the keys contained in the tags, and stores the IDs and revised key data on the server. 4. The device reads the AAR from an NFC tag and launches the application, which then reads the ID and authentication key from the same tag. These parameters are used to obtain the menu from the server. User information comprises, among other things, the username, email address, password, and allergy items. The menu is shown as a picture given by the restaurant, overlaid with additional information. 5. The server hosts the following databases; "user" refers to a customer: - User data: includes user IDs, names, email addresses, passwords, tokens, and other information. 6. NFC tag data: NFC tag IDs, shop data, table IDs, authentication keys, and user IDs are all stored in this file. This database connects NFC tags to shop tables and users to NFC tags. 7. Menu data: includes the names of menu items, prices, ingredients (allergens), categories, priority, and coordinate mappings for menu pictures linked with store IDs.	1. The suggested method maintains the smartphone allergy information on its own, rather than on a server. We filter the allergen information for cooking while showing the menu smartphone. 2. This study suggested a self-ordering system that presents sophisticated, customised menus to customers via their own devices. 3. This method makes ordering very easy for both customers and waiters. Works like the pager system in US restaurants.	1. The algorithm for order management is not optimised. 2. These systems have significant installation costs. 3. Only 60% of the people who participated in the survey answered satisfaction towards this approach. Hence, it's accuracy is quite low.

C. Montgomery, H. Isah and F. Zulkernine, "Towards a Natural Language Query Processing System," 2020	1. The data for this research was gathered from Zomato, a restaurant search engine, and was made accessible on Kaggle, a public data platform. The information was extracted into CSV files and entered into MySQL. Although this solution is not database-independent, it may be converted to other databases by rewriting the mapping table and graph database to reflect the changed structure. 2. Python 3.7 was selected as the implementation language because of its clear syntax, which makes it a popular choice for most data processing and analytics jobs. There are also several NLP libraries that are Python-compatible. TextBlob2 was chosen as the NLP library since it is lightweight and includes a variety of common operations including part-of-speech tagging and lemmatization. 3. The graph database Neo4j was used to represent the schema.  Fig.3. The database schema. 4. Tokenization, mapping, and the mapping table are components of the query text analysis step. 5. The SQL queries created were limited to the following format: SELECT attributes FROM table [, table] (WHERE attribute=value [and attribute=value]) … (1) where items in curly braces occur once, elements in round brackets may occur once, and elements in square brackets may occur zero or more times. The mapped tokens were organised into three lists: tables, attributes, and attribute-value pairings, as shown below. - All tables in any mapped token will be in tables.- Attributes that are not part of an attribute-value pair will be stored in attributes. - All tokens that have been mapped to a table, attribute, and value will be in attribute=value.	1. The qualitative studies have shown that this approach is near optimal compared to existing approaches. 2. Supports multiple backend databases. 3. Converts natural language queries into SQL queries making it easier for databases to execute them and fetch the required data. 4. Uses 3 level architecture. 5.The test results were promising when it came to non complex queries (eg. many joins and group by clauses).	1. Does not support complex SQL queries. 2. For example, to enable the aggregate function and nested queries like 'how many Mexican restaurants are there in Canada?' and to allow hybrid distributed storage systems. 3. Does not use machine learning algorithms due to which the variety of natural language queries cannot be executed. 4. Artificial intelligence conversation methods may be utilised to enable interactive query processing, as well as to assist disambiguate question goals and forecast the next inquiry for efficient query processing.

Z. Mu, Z. Tan and G. Zhu, "Android-Based Order Recommender System," 2018	1. The order recommender system's system function design is analogous to the design of the building's form. Overall system design, detailed design, database design, dev tools, dev tech, and dev mode are all part of the job here. 2. My SQL database is selected. 3. The background management system interface is made up of modules such as dish management, category management, order management, and restaurant management. Users can use the relevant interface to perform operations. The primary interface is split across two Frame>s. For better management and expansion, the system is split into modules. 4. Login interface is like a big dialogue box. Users will have to login to use the App's operations. 5. After consumers log in, they will send a Post request to the server through socket. 6. The server end's background servlet invoked MySql through jdbc to ask about the restaurant details list and returned the results to the client end. 7. The client cancels the action on the menu page over the internet, which sends a request to the server end. The server end's background servlet queries the restaurant database once again for the dish list. The menu page must evaluate and bundle the returning Json data before displaying it to customers.	1. Makes management of the order automated. 2. The app manages the recommended order, which eliminates most of the inconvenience caused by artificial elements in artificial management and increases the efficiency of meal ordering. 3. Saves communication time. 4. Increases profits for merchants. 5. Reduces customer cost.	1. No security surety. 2. Which algorithm is used to assign orders and manage orders is not mentioned clearly. 3. There is also worry that transitioning from a paper-based system to a computerised one would result in a loss of security for the system. 4. There are no standards for testing all the operations given by the database.
T. Shimmura, T. Takenaka and M. Akamatsu, "Real-Time Process Management System in a Restaurant by Sharing Food Order Information," 2009	1. Each kitchen has an order verification feature by dish. When the PROCESS MANAGEMENT SYSTEM terminal gets order information from service personnel, the PMS notes the time each order was received and recalculates the total number of dishes. The PMS system sends data to the order verification system, which is installed in each kitchen. Because of overlapping units in the preparation process, this function has the potential to increase productivity. 2. Display of elapsed time and delay warning functions. The PMS monitors all orders for order-received times and estimates the elapsed time by dish till the kitchen crew finishes preparing them. An alert rings to warn personnel of the delay when the elapsed time reaches 20 minutes. Kitchen employees often inspect and comprehend the circumstances of the procedure. They assess the importance of orders and prepare them accordingly.	1. If a server misplaces an order sheet, he or she may simply look for information on the order checking system and reprint the order sheet. 2. Using this feature, employees do not have to search all tables for the clients who placed the order. 3. When the kitchen staff members begin preparing them, they will be able to identify how many of each item was ordered. Because of overlapping units in the preparation process, this function has the potential to increase productivity. 4. Makes the restaurant work faster with respect to judgement and improves with respect to memory. 5. Furthermore, the checking feature for the status of other roles assists managers in optimising staff allocations. By	1. The algorithm mentioned for finding out the priority in order management system does not cover every test case. For eg. cancellation/updating of order after it has been included in the order management system. 2. The whole model depends on the internet as POS data is updated almost every second. If the internet goes down even for a minute it can be a huge loss for business. 3. There is no mention of how secure this website/application is. If this software becomes vulnerable, then full data of the restaurant will be exposed.

	3. Checking the status of all kitchen roles' functions. Other positions' statuses can be referred to by the order checking system. When a Chief or Manager recognises a job's delay, he or she might direct other employees to assist the busy position. Using this feature, they may control the load of each job, lowering labour expenses and increasing productivity through employee mobility. 4. For a missing order sheet, use the search feature and reissue it. Unlike traditional POS systems in restaurants, PMS does not print order sheets when orders are received. Instead, order papers are produced automatically on the order checking system when a kitchen worker presses the dish sign on a screen after preparation. When the waiter brings meals to the right tables, the order sheets are used.	implementing this approach in a real-world restaurant, it was proven that the average service time was increased and complaints were significantly reduced.	4. There is no concept of "admin" in this approach. Chiefs are given little more than required privilege. This can be harmful.
M. E. Thomadakis and Jyh-Charn Liu, "On the efficient scheduling of non-periodic tasks in hard real-time systems," 2020	1. We are given a collection of n hard real-time periodic jobs I denoted by :=fi(Ci ; Ti ; Di ; ri); I = 1;::: ;n, where Ci is the maximum execution time, Ti is the request period, and Di is the execution time.ri 0 is the commencement displacement relative to the time origin t = 0, and 0 is the time origin. 2. It gives a priority and accepts Jai only if Cai = Z(rai ;dai), else it rejects Jai. 3. Original condition strong aperiodic task entrance approaches, such as static or dynamic Slack Stealing, as well as methods by Chetto et al., Schwan et al., et Stupid, all need quasi time complexity to ensure 1 stable job on-line. 4. In the next sections, we provide techniques for computing the precise idle processor capacity Z(t1; t2) on-line and in linear time.	1. Assures and arranges firm tasks for FIFO or EDF service for a one-time linear cost of (n) and (n + k), respectively, where k is the number of outstanding firm tasks. 2. This technique yields an exceptionally low actual overhead, and the consistent development pattern demonstrates the efficiency and appropriateness of the methods for scheduling challenging real-time activities, particularly in highly dynamic situations dominated by high task arrival time.	1. Slow in servicing dynamic requests. 2. This algorithm is not proved using mathematical models or graphs. 3. Works exactly like traditional methods if used in non-periodic tasks, i.e. similar results were observed. 4. Cannot be used in real life as fewer parameters are assumed. Can only be used for research and standard setting purposes.
J. Teraiya and A. Shah, "Comparative Study of LST and SJF Scheduling Algorithm in Soft Real-Time System with its Implementation and Analysis," 2018	1. A dynamic pre-emptive scheduling algorithm is the least slack time first algorithm. The job with the least amount of idle time is given the greatest priority. The slack time l is determined by the equation below. l = d – c – t. Where t is the current time.d = due date c denotes the remaining execution time.	1. Works as best scheduling in case of not overload and underload. 2. In overload, ECU% and SR% is good. 3. The outcomes are acquired and compared. According to the observations, the dynamic algorithm LST operates effectively in underload situations and can	1. LST does not work properly in case of overload. 2. SJF does not work at all in case of underload. 3. In an overload condition, ECU and SR percent fall rapidly.

	2. When a new job comes or the current work is completed, the scheduling algorithm runs and calculates the slack time for each task using Equation 1. 3. The new job with the least amount of idle time was chosen for execution. 4. When a new job arrives or the presently performing work is completed, the scheduling algorithm runs and identifies the task with the shortest execution time. The new job with the shortest execution time has been chosen for execution. 5. If a job is completed by the deadline, the system assesses its worth. If a job is completed beyond the deadline, the system receives less value from the task. 6. If a job misses its deadline in a solid real-time system, no value is evaluated, yet there is no calamity. 7. In this article, the LST and SJF algorithms are developed for use in a soft real-time system. The task's worth has been determined by the amount of time necessary to complete it.	plan the majority of the tasks when Load is 1. 4. Both of the algorithms used are easy to implement and do not require much calculation.	4. SJF is used which cannot be used in real life scenarios. As, this algorithm is heavily based on burst time, which cannot be accurately calculated for every scenario.
Vindhya Liyanage, Achini Ekanayake, Hiranthi Premasiri, Prabhashi Munasinghe, Samantha Thelijjagoda Foody – Smart Restaurant Management and Ordering System 2018	Long wait times and serving the wrong order are two classic restaurant blunders that eventually contribute to customer displeasure. The goals of this online programme of the above paper are to fix these flaws and deliver speedy and accurate customer service by creating bespoke menus for each customer based on their preferences. This concept is realised as a mobile application that incorporates cutting-edge IT principles including Business Intelligence, Data Mining, Predictive Analysis, and Artificial Intelligence. The app displays a live map of the eatery. The table's colour changes to represent the current reservation status.	Delivery speed increases due to automation. This algorithm is more optimized than the currently used one. The app displays a live map of the eatery. The table's colour changes to represent the current reservation status.	The algorithm which is used for sorting the orders and queueing them is not given properly i.e. the maths behind it or it's proof is not given. This paper does not include the method through which the most appropriate information of every ingredient present in the dish will be available. The table reservation dashboard is created using IOT, which is a little bit too much and it works on customer's location. What if the customer never gives his/her location permission?

Comparison of Scheduling Algorithms **(ARTICLE)**	1. First Come, First Serve is a data structure similar to the FIFO (First in, First Out) Queue data structure, in which the data piece added to the queue first is the one that departs the queue first. This is commonly used in batch systems. It's simple to comprehend and implement programmatically, thanks to the Queue data structure, in which a new process enters through the tail of the queue and the scheduler picks a process from the head of the queue. Purchasing tickets at a ticket counter is an excellent illustration of FCFS scheduling in action. 2. The shortest job First scheduling prioritises the process that has the lowest burst time or duration. This is the most effective method for reducing waiting time. This is commonly used in batch systems. There are two kinds of it: Non-preventive, Pre-emptive. 3. In the Quickest Job The priority of a process is typically the inverse of the CPU burst time, i.e. the greater the burst time, the lower the priority of that process. Priority scheduling does not always set the priority as the inverse of the CPU burst time; rather, it can be set internally or externally; however, the scheduling is done on the basis of process priority, with the most urgent process processed first, followed by the ones with lower priority in order. FCFS is used to run processes with the same priority. When a process is internally specified, its priority can be determined based on memory needs, time limitations, the number of open files, the ratio of I/O burst to CPU burst, and so on. 4. The Round Robin (RR) scheduling method is primarily intended for use in time-sharing systems. This technique is similar to FCFS scheduling, except Round Robin(RR) scheduling incorporates pre-emption, allowing the system to move between processes. Each process, known as a quantum, is given a set amount of time to execute. Once a process has completed for the specified time period, it is pre-empted and another process executes for the specified time period. Context switching is used to save pre-empted processes' states. This method is simple and straightforward to implement, and the most essential aspect is that it is starvation-free, as all processes receive an equal portion of CPU time. It is crucial to remember that the duration of a time quantum ranges from 10 to 100 milliseconds.	The FCFS method has no complicated logic; it simply queues process requests and performs them one by one. As a result, FCFS is quite basic and straightforward to implement. Eventually, every process will get a chance to run, so starvation doesn't occur. Throughput is enhanced because more processes can be completed in less time in SJT. Because the CPU serves each process fora set time period, all processes are assigned the same priority. Because each round robin cycle gives each process a defined time to run, starvation does not arise. There is no procedure that is overlooked. A process's priority can be set depending on memory requirements, time constraints, or user preferences. A high-end game, for example, will have better graphics, which implies that the process that updates the screen in a game will be prioritised in order to achieve greater graphics performance.	1. Because there is no pre-emption, if a process takes a long time to run, the processes at the rear of the queue will have to wait a long time before they can be processed. 2. Longer processes will have greater waiting time, and they will finally starve. 3. If the time quantum is less than what is required, the number of times the CPU switches from one operation to another rises. As a result, CPU efficiency suffers. 4. In pre-emptive priority scheduling, a higher priority process can run ahead of a lower priority process that is already running. Starvation happens when a lower priority process is kept waiting for a higher priority activity.

NAME, AUTHOR AND YEAR	METHODOLOGY	ADVANTAGES	DISADVANTAGES
A Review Paper on Online Restaurant Management System By Prof. N. M.Yawale, Prof. N. V. Pardakhe, Prof. M. A. Deshmukh, Prof. N. A.Deshmukh (2017)	The patron's dining experience is completely transformed by a Tablet menu. Existing programmes offer an app that allows restaurants to feed their menus onto iOS and Android tablets, making it easier for diners to flip, slide, and tap through the menu. We want to provide restaurants with an advanced menu display using Android phones and a tablet menu that recommends items based on a recommendation algorithm. Furthermore, we run the programme on an Android-based tablet rather than an iOS-based tablet, which is a more expensive option. We store the database on a cloud-based server, which is both cost-effective and safe. Customers who sit at tables with tablets spend roughly 10% more than those who sit at other tables, according to developers of similar apps ("they buy more when they can do so instantaneously, without waiting for service"). The following modules make up the proposed system : Module 1: Login Module Module 2: Registration Module Module 3:Add/Update/remove Menu Module 4:Account Management Module Module 5: Station Tracking Module Module 6: Place Order Module Module 7: Carting Module Module 8: Order Manage Module Module 9: Messaging Module Module 10: Logout Module	1. There are no misunderstandings or difficulties. 2. Online meal ordering will be available 24 hours a day, seven days a week. 3. The online menu is simpler and easier to utilise. 4. The number of users or clients grows. 5. It is in charge of accelerating the growth of your business on the internet.	1. The customer must login before placing an order, otherwise placing an order is not possible. 2. An internet connection is required. 3. Adding GPRS module adds to complexity.

Foody - Smart Restaurant Management and Ordering System By V. Liyanage, A. Ekanayake, H. Premasiri, P. Munasinghe and S. Thelijjagoda (2018)	A requirement analysis was performed first in order to design the system, and Foody was developed based on those needs. Users can download and register for this mobile app on their devices. It will have a pop-up menu with restaurant availability, a map, a table reservation function, a meal ordering tool, the option to see past food reviews, a personalised menu, and the ability to see meal preparation time. The admin part of the mobile app is used to manage orders for culinary personnel. Foody was built using a variety of well-known technologies, including Google Maps, sensor and signal processing, graph API, 3D Max, and Natural Language Processing. Foody uses real-time data in a map to illustrate table availability with the greatest accuracy. The table reservation status is collected using a sensor, and the status is shown using color-changing lights. Data is retrieved from Facebook using the Graph API to provide a unique food recommendation. To generate this up-to-date dataset, users' public data is mostly collected utilising user's posts, "user's likes,' and user's events.' Every food item may be viewed as a rotatable image in the 3D modelled menu, which was produced using 3D max programme. It uses a unique food ingredient API to retrieve the details from Google for the data retrieval part. It will filter the ingredients based on their fat, calorie, carbohydrate, and protein percentages. And, in order to sort the orders by priority, an algorithm is devised that takes into account order priority and preparation time. Identify the notion, tokenize the customer review sentence, and assign a score to each sentence in the Natural Language Processing (NLP) section. Then, using NLP, summarise whether each food item review is positive or bad.	1. Foody stands out among existing online restaurants with mobile apps because it is a comprehensive software that incorporates an LED lighting system, 3D modelling menu, task allocation and scheduling order, as well as sentiment analysis to summarise customer feedback and build a unique menu for each customer. 2. It also retrieves the most detailed ingredient information for each food item. 3. Even while there are many online restaurant management services, such as 'Yelp reservation,' that can check restaurant availability via a website, there is no mobile application that can simply check table availability.	1. A requirement analysis was performed first in order to create the system. 2. Users must download the app before placing orders. 3. Each device must be configured to the restaurant's Wi-Fi ID.
Design and development of Multi-touchable E-restaurant Management System By S. Cheong, Wei Wing Chiew, Wen-Jiun Yap (2010)	The MEMS overview framework, which includes the multi-touchable dining table, the counter payment module, the kitchen display module, the administrative module, and the centralised server. On top of the Zend framework, the suggested MEMS was built with PHP, MySQL, and Adobe Flash AS3 scripting. Customers order food at their table using the multi-touchable dining menu located on the dining	1. This paper described a newly created Multi-touchable E-restaurant management system built on the Zend framework, which addressed some of the issues with the PDA-based food ordering system. 2. The MEMS provides a logical work flow for restaurant personnel to digitally handle	1. Throughout the ordering process, errors and delays will occur, resulting in consumer dissatisfaction. 2. Errors are especially common when it comes to the dining menu.

	table. The menu functions as an informational platform for users to explore and study meal specifics, as well as an ordering platform enabling customers to place and manage orders digitally on the dining table. A menu list, menu items as list objects, an ordering panel, and functional buttons make up the multi-touchable dining menu. Process 1 retrieves data from the database and displays it on the screen. The menu list is a collection of PHP functions that extract data from the database and save it to an XML file. The information will subsequently be pushed into the menu item objects, which will be shown. Process 2 depicts the procedures involved in consistently saving clients' orders in the database. The ordering panel is a platform where clients can add and remove orders.	restaurant operations, from ordering to paying. 3. As part of the MEMS, a web writing system was created to allow managers to easily design, edit, and administer the multi-touchable dining menu. 4. Customers no longer have to wait for an attendant to serve them during busy hours because of the MEMS.	3. Food products are still presented on a traditional menu. Any changes to the food products, such as pricing, availability, or promotions, will necessitate manual menu changes.
Intelligent restaurant system Smartmenu By S. Pieskä, M. Liuska, J. Jauhiainen, A. Auno and D. Oy (2013)	Four software applications and two separate device types make up the intelligent restaurant system. Handheld tablet devices and desktop PCs with touchscreens are utilised to run applications. From the customer's and waiter's apps through the kitchen's and cashiers' applications, the system covers the entire order procedure of a restaurant. On Android tablets, the customer and the waiter use software. The system also contains a server that stores the restaurant's menu in a database as well as all ordering logic information. The gadgets are linked to a secure and private WLAN network. A desktop application designed expressly for this purpose can be used to construct the menu. On an Android tablet-computer, the customer's application runs. The application connects to the database server in real time and downloads the restaurant's menu. The customer can look over the menu and find out more information. Intelligent Restaurant System Smart menu with information on the items (nutrition, calories, etc.) and how to order them. The menu includes explanations, photographs, and information about the items on the menu. The programme also allows the consumer to summon the waiter with a single button push.	1. Because the extensions of the digital menu include elements that enable the waiting staff to connect better with the kitchen and cashier, the newly developed intelligent restaurant system Smart menu introduces a new way of working. 2. It's easier to serve customers quickly and accurately when they have access to accurate, up-to-date information. 3. This comprehensive approach Intelligent Restaurant System Smart menu of all orders aids restaurant staff in comprehending the entire restaurant service procedure. 4. The introduced intelligent restaurant system Smart menu creates a new way of working because the extensions of the digital menu offer features that help the waiting staff to communicate better with the kitchen and cashier. 5. Informative, up-to-date communication makes it easier to serve customers fast and accurately. 6. This holistic approach Intelligent Restaurant System Smart menu of all orders helps the employees of the restaurant to understand the whole restaurant service process.	1. The creation of intelligent restaurant systems is a difficult task. While modern technology is being used, it is important to remember that the restaurant service process must remain customer-focused. 2. Based on our observations and research, the restaurant service process will continue to emphasise human interaction. 3. Both the customer's and the restaurant's needs must be met by the intelligent restaurant system.

CORMS: AN AUTOMATED RESTAURENT MANAGEMENT SYSTEM By Roy Davis, Ninu Francis, Swathi K.Sukumaran, Swetha Jeevan E, Umesh Nair (2017)	CORMS is an integrated system that assists customers in selecting their favourite with a personal touch, that is, it assists the restaurant in creating a personal data file about the customer, which is then used in the future to refer the same customer to repeat his visit and to use the data once analysed to make the best food choice. It is a network-based central control system that can be utilised in restaurants, resorts, and other similar establishments. In the TCP/IP network system, there are three participants: clients and the kitchen manager are client participants, and the manager is the server participant. It is a tab-based application that replaces the menu list and waiters and contains new restaurant management features. Each table has a tab on which the client fills out the cart and sends the information to the primary control system (receptionist or manager) as well as the kitchen manager. Finally, the entire data is updated and transferred to the central database's primary server, allowing the restaurant to manage its current business data. Client-server networking, TCP/IP protocols, wireless connection without the internet, database, and so on are all part of the implementation.	1. It can take the place of both waiters and paper menus. 2. Customers are free to express their food preferences. 3. Restaurant operators will have the opportunity to improve based on client feedback. 4. The cook and all other staff members work within a set time restriction to save time. 5. Customers can be provided with up-to-date information about the requested food. 6. Dish cancellation can be completed in a set amount of time. 7. Such a strategy can be used to adopt a diet-based lifestyle. 8. Restaurants are becoming more customer-friendly.	1. Every Android application runs in its own process with its own user id, which the Android system generates automatically before deployment. 2. Customers may have to wait a long time to have their orders taken, to have their drinks refilled, or to receive their bill at a busy restaurant or with inattentive personnel. 3. Unnecessary waiting can lower client satisfaction and reliability, resulting in lost revenue.
Tablet PC In Restaurant By Ali Adil Ali, Hussain Falih Mahdi (2013)	1. Client/Server: This is a computer architecture that separates functions into client (Requester) and server (Provider) subsystems, with standard communication mechanisms (TCP/IP wireless and wired) to facilitate information sharing. Work on a server, responding to each client in line. 2. Access point (AP): a device that allows wireless devices to connect to a wired network through the use of radio waves (Wi-Fi). (Wi-Fi) allows users to set up a computer network without running cable throughout the building. 3. Customer touch screen (TS): connect it to the server via an access point (AP) and a protocol (TCP/IP wireless IEEE 802.11), after which the customer can select a meal by tapping on the screen and have it delivered to the kitchen and cashier. 4. Printer: A printer is a device that prints an order or a bill, and it connects to the server via a wire (USB or Wireless). 5. The network operations are based on the application layer protocol design for various purposes in the restaurant system. Command, data, and an end string are included in the package.	1. Customers do not need to ask waiters what drinks and foods the restaurant offers when employing the Tablet PC technology to tackle the problem in traditional restaurants. In the customer interface, the customer can find all of the food and drinks. 2. In addition, he or she determines the cost of each sort of food, as well as the time required to prepare it. 3. The customer orders via (TS) by customer interface, which was designed by visual basic in a user friendly way.	1. Customers cannot use their visa card to make payment. 2. System is not multi-language. 3. The customer may require help with the functionality.

Restaurant Management System By Carl Abernethy, Prof. Chris Taylor (2010)	The goal of this project is to develop a restaurant management system that combines the advantages of all existing solutions with none of their disadvantages, as well as many new features. Table 1.1 contains a list of proposed characteristics. Many existing POS (Point-of-Sale) systems are sold with the requisite expensive hardware, thus the richer software solutions are just out of reach for any firm working on a budget. Communication of data between each application. Minimum click touch screen GUI design for efficient ordering. Meal ingredient and cooking preference options. Interface to view active orders in the kitchen. Ability to add flexible discounts; calculating best price for the customer. Interface to maintain and manage the menus and associated meals. Stock control for all ingredients; reducing/increasing stock automatically. Ability to define groups of ingredients that may be used in numerous meals. Flexible meal grid design to fit any screen size. Real time waiter status alerts. User login functionality. Interface for table management and selection. Figure generation; management can view statistics in numerous forms. Automatic daily stock level alerts. Ability to define meals by images as well as text.	1. Meals with enough ingredient stock can be sold sooner with a real-time view of ingredient stock levels. 2. Kitchen and bar staff communicate directly with waiters, allowing the kitchen to alert the waiter to the need for service. 3. All orders have been logged in order to generate future queries.	1. Some of the lower priority requirements had to be scrapped. 2. GUI could have had a more appealing look and feel. 3. Perhaps a web-based GUI would have been a better option, but given the developer's lack of web development knowledge, this would not have been the best option. The design and feel of the GUI could have been better.
ONLINE RESTAURANT MANAGEMENT SYSTEM (ORMS) By HANISAH BINTI MD TAHA (2008)	Deliverables of the Project - The flow of activity will be represented in terms of customer registration module, customer online ordering and reservation module, waiter module, feedback module, menu module, and create report module, depending on the module that was recognised. Module for Customer Registration Customer information, such as personal information and other information about that customer, is stored in the customer registration module. The information was then entered into a database. Module for Customer Online Ordering and Reservations Customer online ordering and reservation module provides a form that must be completed in order-to-order food and book a table via the internet.	1. This website has system for online reservation 2. This website also contains information on this restaurant, such as a map of its location, other facilities, special events, culture and customs, arts and crafts, and a supper cultural performance. 3. After all of the inf formation has been entered, the system will display a notification indicating that the order has been placed successfully.	1. Elements that are in the database should be known. 2. Identifying suitable methodology to be used is time consuming. 3. Fast internet connection is required.

AUTOMATED FOOD ORDERING SYSTEM BY PROCESS SCHEDULING SJF, SRTF, PRIORITY By See WEN Xiang , Aiman Arriady , Abu Sayed Roman, Hamid Madni	The main objective of this project is to develop a program for shortest job first scheduling Shortest Remaining Time First and Priority scheduling which helps to improve the Food Ordering process. There are many algorithms available for CPU scheduling. But we cannot implement in a real-time operating system because of high context switch rates, large waiting time, large response time, large turnaround time, and less throughput. The proposed program improves all the drawbacks of simple shortest job first scheduling Shortest Remaining Time First and Priority scheduling architecture.	This paper improves all the drawbacks of simple shortest job first scheduling Shortest Remaining Time First and Priority scheduling architecture and improves the process and the time taken to do it.	The priority giving method is not optimised as it is only dependant on cooking time, can go into starvation. This paper mentions it is using Shortest Job First Technique in 'real time' which is ironical. This paper does not talk about cases when there are more than 2 orders placed simultaneously i.e. when arrival time is same. All these algorithms are highly dependant on cooking time which can be changed from staff to staff. Lastly, when this algorithm is actually implemented i.e. after in what intervals is not mentioned.

//9 Papers Done By Ritvik Kohli 19BCE2223//

NAME, AUTHOR AND YEAR	**METHODOLOGY**	**ADVANTAGES**	**DISADVATAGES**
Digital Ordering System for Restaurants Using Android Ashutosh By Bhargave, Niranjan Jadhav, Apurva Joshi, Prachi Oke, Prof. Mr. S. R Lahane (2013)	1. Each tablet will be provided according to the author's strategy, allowing clients to traverse the meal as needed. 2. You may also look for food products based on their name and price. 3. Prices and user ratings are also used to sort the items. This enables the buyer to locate or select a tasty dish that many people enjoy. 4. The proposed system also alleviates the issue of making adjustments to certain items if the food's quality improves as a result of low ratings. 5. The author also includes a function that shows the availability of an item during a specific time period on the menu card.	1. According to the author, acceptable reaction times for system functionality are part of the performance requirements. 2. For example, user interface displays will load in 2 seconds or less, log-in information will be confirmed in 4 to 5 seconds, and searches will return answers in 5 seconds or less.	1. One tablet will be utilised for each table, which is the biggest disadvantage/limitation of this approach. So, if the restaurant has 20 to 25 tables, the restaurant will need to buy that many tablets. 2. Any restaurant will have to purchase and maintain these tablets, which will be a financial burden.

E-Restaurant: Online Restaurant Management System for Android By Dr. Vinayak Ashok Bharadi, Vivek Ranjan, Nikesh Masiwal, Nikita Verma (2013)	1. The suggested system's primary components, according to the author, are the backend, which contains the web server and database, and the frontends, which include both the client and kitchen front ends. 2. The system is built on the Model-View Controller architecture, which is very popular. MVC is a popular website framework that has been proved and tested. 3. None of the interfaces "speak" to the database directly. Instead, they use RESTful web services to conduct CRUD tasks on databases. 4. The buyer is presented with an Android-based tablet. The database is synchronised with this tablet on centralised cloud-enabled servers. When the tablets are synced, the menu data is saved locally on the tablets, allowing for faster access to the menu. 5. The user can then scan the menu anyway, sorting items by price, popularity, ratings, and other factors. The user may also read more information about any item, such as nutrition statistics, components, trivia, and so on. The user can also see customised suggestions for things they would enjoy. The consumer experience is improved as a result of this. 6. The above-mentioned paper's author has also proposed a new method that will make it easier for customers to build orders. You do so by looking at previous orders of the customer and choosing the same dishes or dishes which are similar to those previously liked.	1. According to the author, their proposed suggestion system saves customers a lot of time when it comes to selecting a food and placing an order. 2. The devices usually present the customer with decent selections that they could appreciate by providing automatic suggestions based on past orders.	1. The author has created the website and all of the other required components, but he has not verified that the website is compatible with all devices. 2. They tested it on a few Android phones and tablets to see if it works. They haven't yet delved into the world of IOS.
RFID RESTAURANT POS SYSTEM By HORNG-LIN SHIEH* , YI-CHUN LIAO (2011)	1. The user must first log into the system before selecting the RFID to host connection port from the Comport pull-down menu. 2. The consumer can then select OK to proceed to the order menu, where the RFID tag will be read using the RFID reader. 3. Users can view images of the dishes as well as the components for each dish, by clicking on the links on the image. 4. In addition, each cook utilises an RFID tag to determine the state of prepared foods before selecting them. 5. RFID is made up of a card reader that sends identification data via radio waves. The tag can instantly recognise the signal and share data.	Based on the advantages of RFID, the study contributes to ● reducing restaurant demand for labour ● saving operating costs, ● reducing paper consumption in order to save energy and reduce carbon dioxide.	1. While the notion of employing RFID in restaurants is a nice one, the authors haven't provided any details on how the planned system will be implemented. 2. They haven't mentioned the system's results after it has been used.

Design of Restaurant Billing System (E Bill Resto) by Applying Synchronization of Data Billing in Branch Companies to Main Companies Based on Rest API By tM Mahaputra Hidayat, R Dimas Aditya, Alek Siswanto (2020)	1. The proposed system allows the E-Bill Resto Officer to welcome the Guest to the Restaurant transaction Data, whereas the Administrator is the User who may access the Web Service Data and synchronisation process of the Guest Transaction in the data database, Master or Slave Data Applications. 2. The author utilised AngularJS for this project. 3. Between the two synchronisation table applications with the identical properties, use the application as a slave data. Certain parts of the programme are designed to be comparable to the original characteristics of the existing E Bill Resto Information System, which are simply used to test the data synchronisation process. 4. Three applications have been made for the proposed system: the Master Application for the information system for restaurants replacing the existing application; the Master application for web services; and the application for slave-management databases. 5. Then the GUI was implemented to make it easier.	1. The proposed system aids in the enhancement of QoS. (Quality of Service). 2. It also aids in the reduction of packet loss when moving from one system to another or from one application to another.	1. The data is stored in local databases in the above system. While this has certain advantages, it also has a lot of drawbacks. 2. Alternatively, the author might save the data on the cloud. It is also easier to complete some computations as a result of this.
DEVELOPMENT AND IMPLEMENTATION OF AN E-RESTAURANT FOR CUSTOMER -CENTRIC SERVICE USING WLAN AND RFID TECHNOLOGIES By Ching-Su Chang, Che-Chen Kung, Tan-Hsu Tan (2008)	1. All clients should have a membership card with an RFID tag, according to the suggested scheme. The authors created two service models based on this: 2. Customers are actively detected and customer data and spending records are retrieved when a restaurant customer is entered by the server-site 3. RFID reader on the counter. The customer orders and the order will be presented on the PC screen for confirmation at the same time on the desk. After confirmation, the order is transferred via WLAN to the kitchen's back-end server. 4. If a customer enters the restaurant with an RFID membership card and proceeds directly to a table, the waiter can place an order using a handheld PDA and RFID reader. The order is presented on the PDA for customer confirmation. The order is then sent to the kitchen's back-end database through wireless internet. 5. According to the author, the idea's experimental findings were positive, and it has the potential for practical use.	1. The author claims that the aforementioned concept might be implemented in a restaurant. 2. He claims that the aforesaid method cuts down on servers' time spent taking orders and useless calculations. 3. With the RFID scanner and RFID tags, this is now done by the computer.	1. According to the author, clients will be handed RFID tags when they become members of the restaurant, and these tags will be used for a variety of functions. 2. However, this poses a dilemma. It is not a good idea to ask individuals to join only to receive an RFID tag and a membership card. 3. This procedure may not be popular with all customers, and it may have a negative impact on the restaurant's reputation.

CareDB: A Context and Preference-Aware Location-Based Database System by Justin J. Levandoski, Supervised by: Mohamed F. Mokbel (2010)	1. When Care DB receives a query, it initially sends it to the user preferences store. 2. Any contextual data unique to or relevant to a user is injected into the Question as well. 3. The query preference and context-awareness are handled by the query processing and optimization engine. 3. The query processing engine is the fundamental benefit of Care DB and this thesis. 4. The major objectives of this module are to: Integrate into an existing database system's core, couple preference evaluation with typical database operators, and provide context awareness for various sorts of preferences. 5. Support integration of context-aware query processing, which may require costly calls to third-party contextual data sources. Support efficient continuous query processing with shared performance and incremental evaluation.	1. In the database system's primary query processor, the author was able to develop place-based services, which comprise forms of preference and context. 2. He has successfully built the proposed model using Care DB in conjunction with this. 3. This is a location-based service that works based on the distance and quality of the area.	1. According to the author, the proposed system requires further development and the addition of some other parameters that may alter the system's operation.
The Development of Self-service Restaurant Ordering System (SROS)	1. When clients first visit the system, the offered notion is shown as the system's homepage.	1. The restaurant owner can lower the personnel and monthly costs by using the technology to replace various duties.	1. According to the author, the proposed system needs to have a few additional features added to it in order to increase its functionality.
M. Z. H. Noor, A. A. A. Rahman, M. F. Saaid, M. S. A. M. Ali, M. Zolkapli	1. On the following page, you'll get restaurant information as well as promotion price. 2. Customers then go to the menu page to look over the entire menu before placing an order. 3. Simply add and remove menu items as needed, and the total cost will be calculated automatically. 4. A better connection is established between the user and the server via the router and the LAN wire. 5. The order is kept in the database on the admin computer. This order is also visible on the kitchen computer used for meal preparation.	1. The system shortens the customer's wait time and lowers the likelihood of misunderstanding. 2. During busy hours, it is critical to ensure that the consumer is satisfied with the service.	1. For example, the author should include a function that allows consumers to reserve a table or dishes in advance.

Zero Food Waste: Food wastage sustaining mobile application By M.D.C.J Gunawardane, H.A.N ushpakumara, E.N.M.R.L Navarathne, Shashika Lokuliyana, K.T.I. Kelaniyage, Narmadha Gamage (2019)	1. A custom dataset is trained using the TensorFlow API for the analysis and identification of food items in a single image. 2. As a baseline model, the Mobile Net SSD was used to train the dataset. 3. Using the TensorFlow object detection model, the exact location of the object can also be provided via a box framing the object. 4. The TensorFlow Object Detect API is used to calculate the volume of food items by determining the distance between the camera and the food items, extracting data such as radius, length, and other attributes of the food items, image fragmentation techniques (k-means), and weight estimation using density tables.	1. While most apps for minimising food waste focus just on food waste, the "Zero Food Waste" App also tracks users' nutrient intake and provides healthy meal planning and suggestions for foods that are readily available at home. 2. The "Zero Food Wastage" app's intelligent chatbot makes the app more engaging than any other food waste reduction software, and the intelligent chatbot makes the app much easier to use by guiding users through the receipt.	1. The above-mentioned paper's authors make no mention of how effective their app is. 2. They did not test their application and as a result, no results were produced.
Website Development of Restaurant Management By Prof. Jadhav A.K, Nupur Ligade, Renuka Savale, Malay Chitodkar, Rushikesh Salve	The Restaurant Management website provide convenience for the customer. The restaurant management system is there to help communication between all teams within a restaurant by minimizing the probability of human error and getting a more efficient and effective information. This System set up menu online and the customers easily places the order with a simple mouse click. By using the food menu online anyone can easily track the orders, maintain customer database and improve food delivery service.	The restaurant management system helps in communicating between all the teams within a restaurant by minimizing the probability of human error and getting a more efficient and effective	The mechanism of registering table given in the paper is not clearly mentioned i.e. what algorithm or method they are using for that. It's mentioned that it is a restaurant management website, but there is no way or option to input and store customer feedbacks. Basic schema of the database is missing in this paper. It says that there needs to be a tablet on each table, if this is a website, then it can be replaced with a QR code.

3. Proposed Work

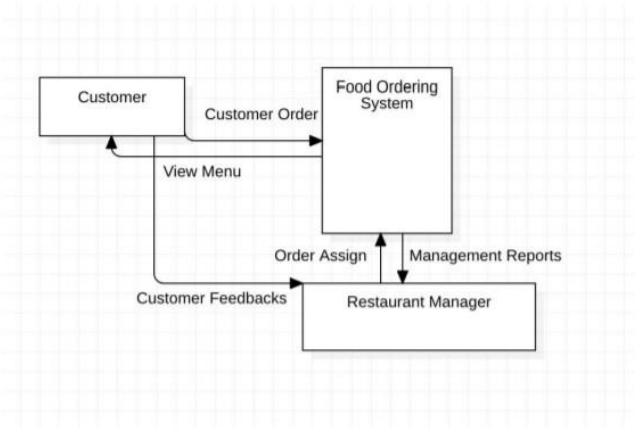
Proposal/ideas formulated from the literature analysis. Most of the papers studied for the literature review tackled a frequent problem and that is to speed up the service of the restaurant. The authors noticed that most of the restaurants really struggled to meet the high service standards because of miscommunication, laziness, poor planning, bad interfaces used in the computer systems, etc. To solve these problems, they have produced various ideas as explained in the literature review. *elsarticle* Like the authors we have also noticed the above problems in India. Most of the restaurants here still use the old ways of taking orders, which is fine but not efficient. To tackle we have proposed a system or a website which allows the customer to order their favourite dishes by looking at their images. The waiters, Chefs are also assigned to the customers in a Round Robin Fashion. Once they order their food, the order gets stored in the database and the chefs, waiters can view the orders by logging on to the website through their own log in credentials. Customers can also provide their feedback on the website once they have finished their food. The feedback can only be given once per order. The feedback in then stored in the website. New food items along with their image can be added by the manager of the restaurant at any time. At the end of the day, we can also see the summary of all the orders along with the chef details, waiter details, etc.

Architecture diagrams.

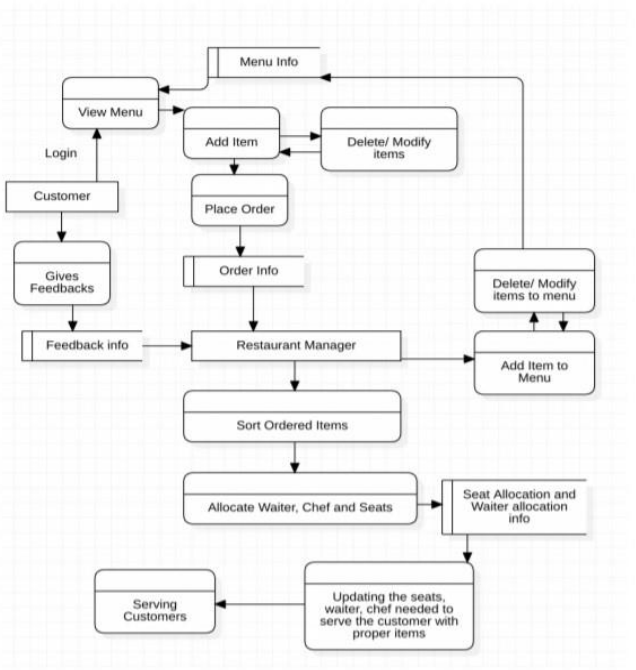
The customer will be able to view the menu on the website, select the items he/she prefers to order, place the order, and delete items after placing the order. The customer will be able to view their total bill amount to be paid. The customer will also be able to give feedback which can be used by the restaurant to change their quality of service.

The restaurant manager will be able to set a menu for his restaurant, add and remove items from the menu, assign orders to the chef and waiter. The chef and waiter can also independently view orders assigned to them on separate webpages. The restaurant manager can also view the order history and total bill amount.

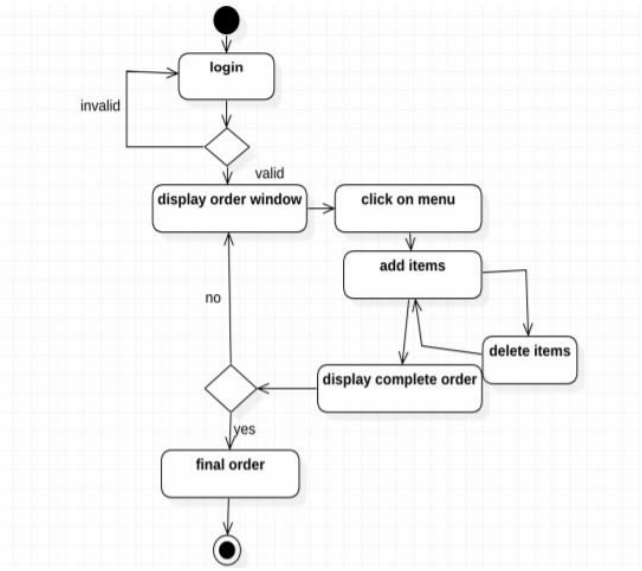
0 – LEVEL DFD



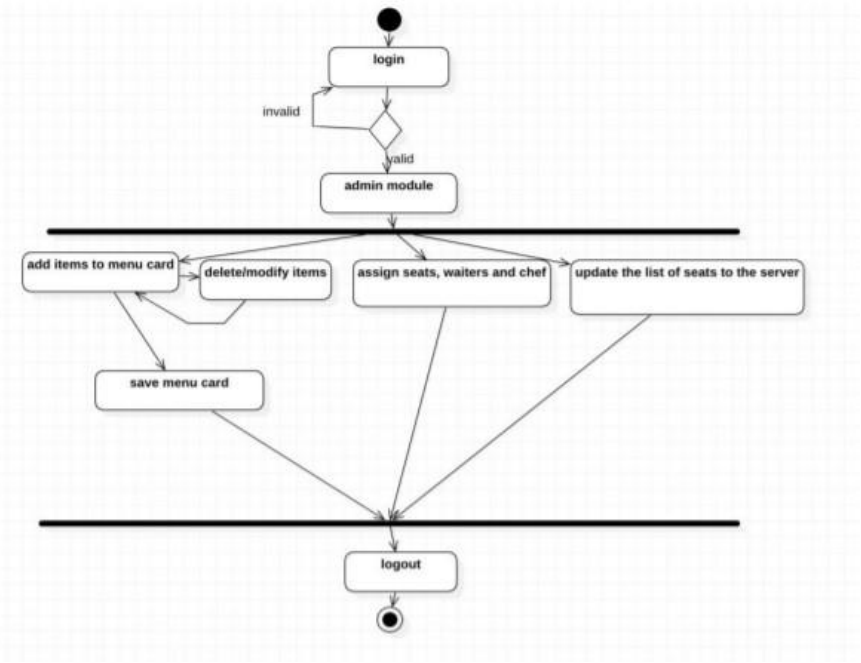
1-LEVEL DFD



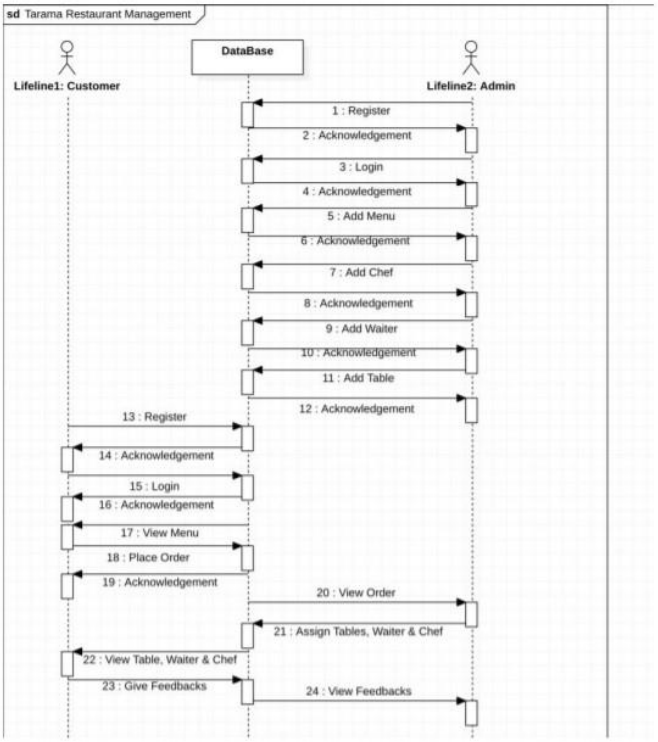
Activity Diagram for Place Order:



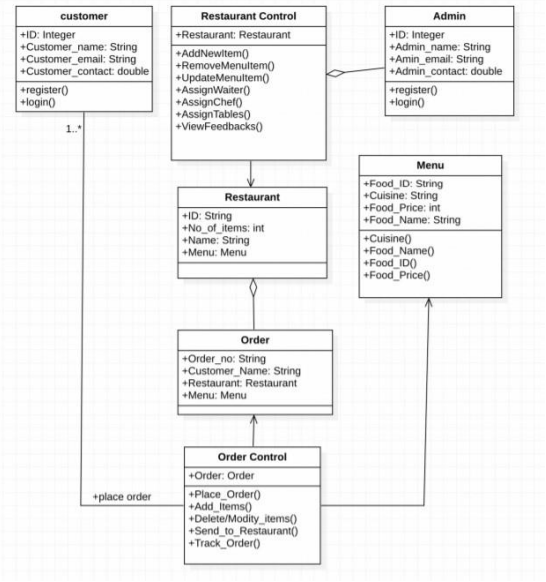
Activity Diagram for admin:



SEQUENCE DIAGRAM

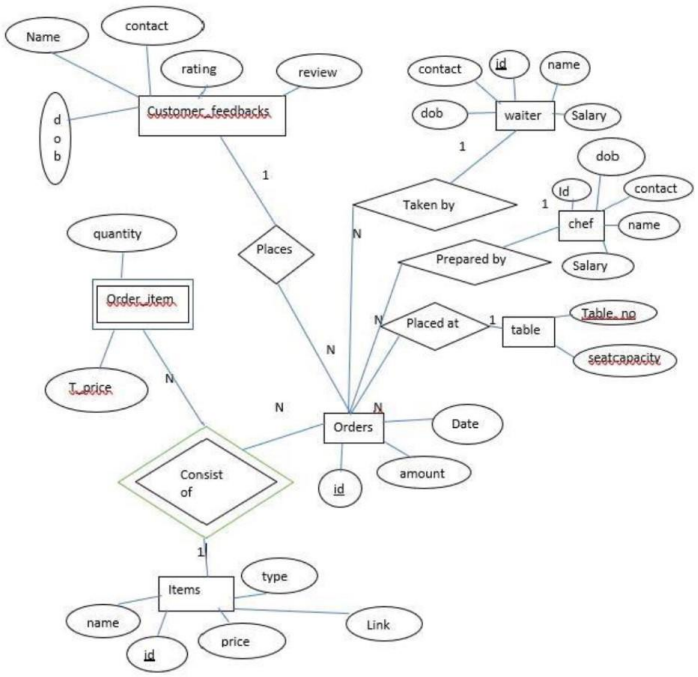


CLASS DIAGRAM

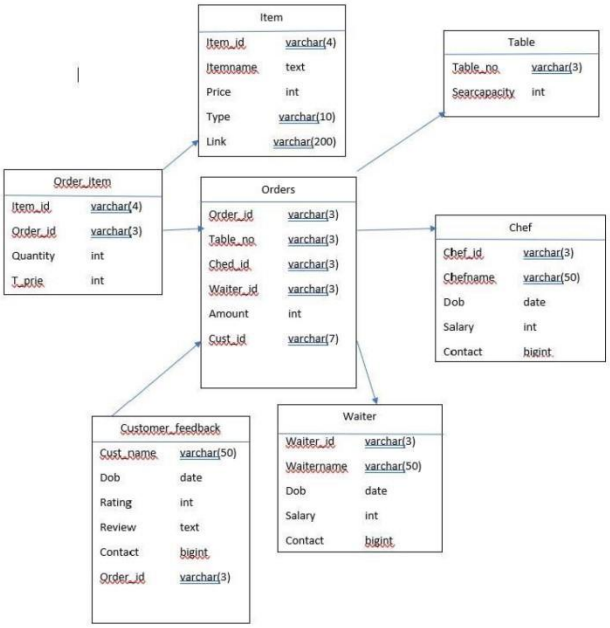


Data sets used.

ER DIAGRAM



ER TO RELATIONAL SCHEMA



Result Grid | Filter Rows: | Edit:

order_id	date	table_no	chef_id	waiter_id	amount
OI13	2020-11-01	T11	C13	W13	100
OI14	2021-05-05	T11	C11	W11	5600
OI16	2021-02-15	T11	C13	W13	150
OI17	2021-02-15	T11	C11	W11	150
NULL	NULL	NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Edit:

chef_id	chef_name	dob	salary	contact	age
C11	chef1	2000-02-21	90000	1111111111	20
C12	chef2	2000-02-02	90000	7276227324	20
C13	chef3	1999-03-03	90000	2222222222	21
NULL	NULL	NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Edit:

waiter_id	waiter_name	dob	salary	contact	age
W11	waiter1	2000-02-21	90000	1111111111	20
W12	waiter2	2000-02-22	90000	2222222222	20
W13	waiter3	1999-02-02	90000	3333333333	21
NULL	NULL	NULL	NULL	NULL	NULL

Result Grid | Filter Rows:

table_no	seat_capacity
T11	3
T12	4
T13	1
T14	5
T15	6
T16	1
T17	1
T18	1
T19	1
T20	5

Result Grid | Filter Rows:

count(order_id)	date
1	2020-11-01
1	2021-05-05
2	2021-02-15

Techniques.

- For front end bootstrap, tailwind.css is used as we will be producing a fully mobile responsive website as well.
- For the back end i.e., inter connection and creation of routes, connection of our web application interface to database, and importing our custom algorithms we are using Flask which is a micro-web framework using ’apps’ to manage routes.

- For the performing queries and doing CRUD operations, we have used MYSQL server which is a table-based data structure database.
- For automatic assigning of chefs and waiters, we have used round robin scheduling algorithm as it computes the least waiting average time amongst all the other scheduling algorithms.
- For the crowd management functionality, we have made a custom scheduling from scratch. Here we are assuming that burst time is directly proportional to number of people in a group.
- We have also made a custom bubble sorting algorithm. We basically take two arguments in the form of lists and sort the second list on the basis of first list. This is used everywhere in our 'scheduleCheck()' method.
- Our code makes combinations of groups, and selects the one combination which can fill up all (or most) of the seats vacant at that point of time. Vacancy gets cleared out as each iteration goes on, with group leaving when their status in currentStatus reaches 0.
- After that, the group is popped out of the restaurant, vacancy is freed, and the code searches for a group or a combination of groups outside the restaurant that can enter the restaurant. If it finds such a group, that group enters and sufficient changes are made in currentStatus list.
- If it cannot find any group, then it waits till sufficient vacancy is cleared so that groups can enter easily.
- In the end, we have used 'Reportlabs' to convert the final output into a pdf report.

4. Results and analysis

The round robin algorithm used for waiter and chef automatic assignment is mathematically more efficient as it's average waiting time and average turn around time is less than that of first come first serve which is used by the restaurant managers in real time. The use of Shortest-Job-First Scheduling technique will fail in real time as it is highly dependent on the burst time i.e., serving time here. Priority scheduling will also be not feasible here as this is small scale restaurant hence there is not much difference between the waiters and chef posts.

The algorithm made by us on the second part of the website proves to be more optimal and efficient than the First come first serve basis as the average waiting time and average turnaround time is less than that of our algorithm. Shortest-Job-First Scheduling technique cannot be used for real time applications as it is a very ideal situation-based algorithm i.e. burst time should be already known and then only it works. Round robin algorithm does not even suit this type of situation as only non-pre-emption algorithms can be applied the restaurant situation of ours. Hence, priority scheduling, non-pre-empted which priority is given by our special algorithm can only be applied for the above mentioned scenario.

5. Conclusion

At the end we were successful in creating a proper functioning website for a restaurant which helps in serving the customer faster and also reduces the work load on the staff members of the restaurant.

This is done by automating few of the processes with the help of our website. The website made by us successfully takes the orders of the customers and also provides these orders to the database of the admin from where the respective chefs and waiters can see what to cook and what to serve. In case the customer needs to delete an order they can do that on their own too.

The customers can also give their feedbacks on their own, once they have finished eating. The feedbacks are also stored in the database and can be viewed by the admin.

The algorithm made by us also helps in solving the problem in finding out how much time it will take for the queue to clear outside the restaurant. Our method also proves to be more optimal and efficient than the First Come First Serve, the Round Robin, and the Shortest-Job-First Scheduling algorithms.

6. References

120 [1] Gunawardane, MDCJ and Pushpakumara, HAN and Navarathne, ENMRL and Lokuliyana, Shashika and Kelaniyage, KTI and Gamage, Narmadha, Zero Food Waste: Food wastage sustaining mobile application (2019) 1–4.

121 [2] Bhargave, Ashutosh and Jadhav, Niranjan and Joshi, Apurva and Oke, Prachi and Lahane, Digital ordering system for restaurants using Android (2013) 1–7.

122 [3] Bharadi, Vinayak Ashok and Ranjan, Vivek and Masiwal, Nikesh and Verma, Nikita, eRestaurant: Online restaurant management system for android (2013) 1–6.

123 [4] Shieh, Horng-Lin and Liao, Yi-Chun, RFID restaurant POS system (2011) 1–4.

124 [5] Hidayat, M Mahaputra and Adityo, R Dimas and Siswanto, Alek, Design of Restaurant Billing System (E Bill Resto) by Applying Synchronization of Data
Billing in Branch Companies to Main Companies Based on Rest API (2020) 1–5.

125 [6] Chang, Ching-Su and Kung, Che-Chen and Tan, Tan-Hsu, Development and implementation of an e-restaurant for customer-centric service using WLAN and RFID technologies (2008) 1–6.

126 [7] Levandoski, Justin J and Mokbel, Mohamed F, Caredb: A context and preferenceaware location-based database system (2010) 1–4.

127 [8] Noor, MZH and Rahman, AAA and Saaid, MF and Ali, MSAM and Zolkapli, M, The development of Self-Service Restaurant Ordering System (SROS) (2012) 1–6.

128 [9] Gunawardane, MDCJ and Pushpakumara, HAN and Navarathne, ENMRL and Lokuliyana, Shashika and Kelaniyage, KTI and Gamage, Narmadha, Zero Food Waste: Food wastage sustaining mobile application (2019) 1–4.

129 [10] fah Almukhaizeem, Nourah Almatar , Muhammad I. Sarfraz , Muhammad Sarfraz , An Elegant and Efficient Database Design for Home-Based Restaurants(2018) 1– 6.

130 [11] Takuma Fujita, Hideki Shimada, and Kenya Sato, Self-ordering System of Restaurants for Considering Allergy Information (2018) 1–6.

131 [12] Chantal Montgomery, Haruna Isah , Farhana Zulkernine, Towards a Natural Language Query Processing System (2020) 1–6.

132 [13] ZhenhaiMu, Zhongxuan Tan, Android-based Order Recommender System(2018) 1–4.

133 [14] Shimmura, Takeshi and Takenaka, Takeshi and Akamatsu, Motoyuki,2009 International Conference of Soft Computing and Pattern Recognition (2009) 1–3.

134 [15] Thomadakis, Michael E and Liu, Jyh-Charn Proceedings 20th IEEE Real-Time Systems Symposium (Cat. No. 99CB37054) (1999) 1–3.

135 [16] Teraiya, Jay and Shah, Apurva, Comparative study of LST and SJF scheduling algorithm in a soft real-time system with its implementation and analysis (2018) 1–5.

136 [17] L. Almukhaizeem, N. Almatar, M. I. Sarfraz, M. Sarfraz, Comparison of Scheduling Algorithms in OS | Studytonight (2018) 1–6.

137 [18] XIANG, SEEWEN and AZMAN, AIMAN ARRIADY BIN and ROMAN, ABU SAYED KHUDRI, AUTOMATED FOOD ORDERING SYSTEM BY PROCESS SCHEDULING SJF, SRTF, PRIORITY (2018) 1–6.

138 [19] Yawale, NM and Pardakhe, NV and Deshmukh, MA and Deshmukh, A Review Paper on Online Restaurant Management System (2018) 1–6.

139 [20] Liyanage, Vindya and Ekanayake, Achini and Premasiri, Hiranthi and Munasinghe, Prabhashi and Thelijjagoda, Samantha, Foody-Smart Restaurant Management and Ordering System (2018) 1-6.

140 [21] Cheong, Soon Nyean and Chiew, Wei Wing and Yap, Wen Jiun, Design and development of multi-touchable e-restaurant management system (2010) 1–6.

141 [22] Piesk, Sakari and Liuska, Markus and Jauhiainen, Juhana and Auno, Antti and Oy, Delitaz, Intelligent restaurant system Smartmenu (2013) 1–6.

142 [23] Roy Davis, Ninu Francis, Swathi K.Sukumaran, Swetha Jeevan E, Umesh Nair, CORMS: AN AUTOMATED RESTAURENT MANAGEMENT SYSTEM (2017) 1–8.

143 [24] Ali, Ali Adil and Mahdi, Hussain Falih, Tablet PC in restaurant (2013) 1–4.

144 [25] Rehana Akhter, Restaurant Management System (2010) 1–81.

145 [26] Hanisah Bt. Md Taha, Online Restaurant Management System (2008) 1–27.

146 [27] Jadhav, AK and Ligade, Nupur and Savale, Renuka and Chitodkar, Malay and Salve, Rushikesh, Website Development of Restaurant Management (2018) 1–6.