

e-Restaurant: Online Restaurant Management System for Android

Dr. Vinayak Ashok Bharadi

Associate Professor, IT Department, Thakur College of
Engineering & Technology
Kandivali(E) , Mumbai, India

Vivek Ranjan, Nimesh Masiwal, Nikita Verma

UG Student, IT Department, Thakur College of
Engineering & Technology Kandivali(E) , Mumbai, India

Abstract—The simplicity and ease of access of a menu are the main things that facilitate ordering food in a restaurant. A Tablet menu completely revolutionizes the patron's dining experience. Existing programs provide an app that restaurants can use to feed their menus into iOS & Android based tablets and make it easier for the diners to flip, swipe & tap through the menu. We here aim to provide the restaurants with a tablet menu that would recommend dishes based on a recommendation algorithm which has not been implemented elsewhere. In addition to this we run the app on an Android based tablet & not on an iOS based tablet which is more expensive alternative. We use a cloud-based server for storing the database which makes it inexpensive & secure.

Keywords—*Recommendation; Tablet; menu; Intelligent; Android application; restaurant*

I. INTRODUCTION

Over the years, technology has tremendously revolutionized the restaurant industry. But much of the innovation has been with point-of-sale (POS) operations. Yet other areas of a restaurant are ripe for innovation, such as the menu.[4] Traditional restaurant service requires waiters to interact with customers directly before processing their orders. However, a high-quality recommendation service system would actively identify customers and their favorite meals and expenditure records.[1]

There is a famous saying that “People eat with their eyes”.[6] The e-Menu provides additional information about menu items and drinks than a traditional paper menu. With interactive pictures it gives additional information about the food item. Tablets are said to eliminate order-taking errors from the waiters. In the kitchen, there is less confusion as everything is now written clearly. Developers of similar applications maintain that customers who seat at tables outfitted with tablets spend about 10% more than those at other tables (“people buy more when they can do so instantly, without waiting for service”).[9] With the visuals, you know exactly what you're going to get in your plate The service goes quicker. Tablets are said to allow cutting the labor expenses. Customers feel more involved in the process. Restaurants can build their e-reputation and customer community in live.

The restaurant menu, as we know it, has evolved from its humble beginnings on carte chalkboards and imageless print to today's detailed, colorful displays. With the emergence of digital tablets and user-friendly touch screen technology

menus can move to a whole new surface. With this electronic menu, orders can be taken correctly the first time. There is no need to run back and forth to a distant terminal, because the terminal is always with the server. Every order is associated with an individual seat at the table, and orders are built one customer at a time, just like on paper, but with greater accuracy. Items can also easily be shared by the whole table, moved or modified, and noted and the cost can be calculated in real time.[10]

The Recommendation algorithm suggests dishes to the patrons based on previous orders. It makes it easier for the customer to build his/her order and also view the most popular dishes. Moreover, various dimension filters can be used according to individual preferences e.g. Price, taste, quantity, etc.

In a study earlier [1], a preliminary experiment was conducted in a restaurant, and a questionnaire survey was administered to fifteen waiters and forty-five customers. The survey result was encouraging. In addition, extensive interviews with restaurant owners were conducted and the results indicated that the proposed system is useful in reducing running cost, enhancing service quality as well as customer relationship.[9]

II. EXISTING SYSTEMS

A. Conventional Systems

Restaurant services such as making reservations, processing orders, and delivering meals generally require waiters to input customer information and then transmit the orders to kitchen for meal preparation. When the customer pays the bill, the amount due is calculated by the cashier. [1]

Although this procedure is simple, it may significantly increase the workload of waiters and even cause errors in meal ordering or in prioritizing customers, especially when the number of customers suddenly increases during busy hours, which can seriously degrade the overall service quality. [1]

B. Electronic POS Terminals

A very commonly implemented system, currently being used by numerous restaurants and chains all over the world, is the electronic point-of-sale terminal system.

Here, the servers/waiters generally take the order from the customer and head onto a terminal, where they can feed the order into a computer. The order can then be transmitted to the

kitchen automatically via the terminal through a network, or it may even be delivered manually by the server to the kitchen.

Although a huge improvement over the pen and paper still prevalent over the world, this does not have much value addition for the customer and mostly only benefits the establishment and the administration of the establishment.

C. Tablet Based Menus

With the current onslaught and popularity of touch based devices, especially the Apple iPad, it did not take long for the tablet based menus to make an entrance in to the market.

In their current state, these menus are just a glorified version of their paper based counterparts.

The workflow, even with these tablet menus, remains the same as the one mentioned in section 2 (A) or a combination of the previous two. There is very little value addition to the customer and the establishment or its staff.

The available technology is definitely not being utilized to its maximum potential.

III. PROPOSED SYSTEM

A. System Architecture

The following figure (Fig 1), demonstrates the basic architecture of our system.

Understanding the other intricacies and details of the system will become a lot easier if one goes through this figure, before diving into the rest of it.

The system will consist of the following main components: The backend, which is made up of the web server and the database, and the frontends that include both the patron frontend (delivered as a native mobile application) and the administration or the kitchen frontend (delivered as a web application).

This system is based on the very popular Model-View-Controller (MVC) architecture. MVC is most commonly used in websites, very popular and tried and tested.

None of the frontends directly “talk” to the database. They instead rely on RESTful web services that can be used to perform CRUD operations on the database.

B. System Overview

The most important components of this system are the database and the patron frontends or the tablet applications.

Providing value to both the business and its patron is an important objective, but we believe that one follows the other.

Following that belief, the customer is given a whole lot of importance.

The following figure (Fig 2) demonstrates the flow of our tablet application

The patron will be presented with a tablet, running the Android OS. This tablet will be synchronized with the database running on our centralized cloud powered servers.

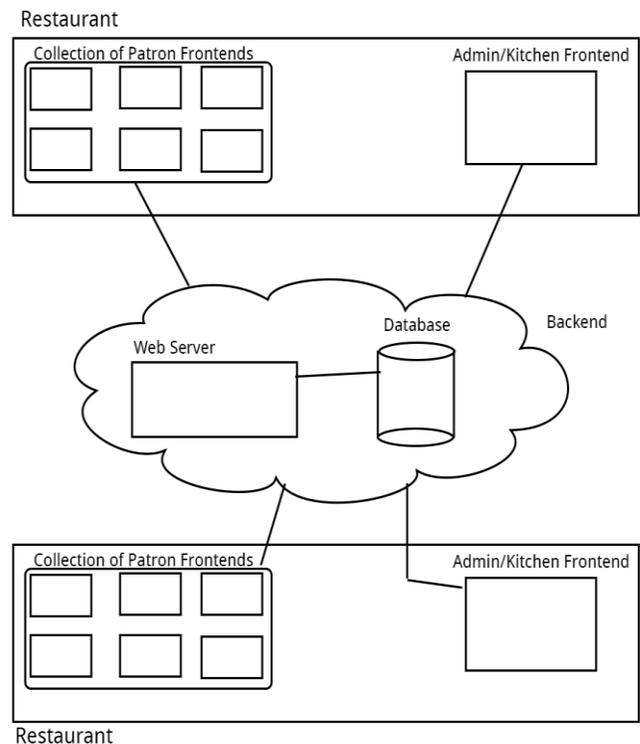


Fig.1. System architecture & the main components

The menu data, upon synchronization, is stored on locally on the tablets so that the user, i.e. patron, need not wait for the menu to be downloaded from the servers. This will allow faster access to the menu.

The user can then browse the menu however they want to, sorting the items on various dimensions like price, popularity, ratings, etc. The user can also click through to view more information about any item like nutritional information, ingredients, trivia and any other content that the restaurant administration may feel like including.

The user can also view personalized recommendations for items that they may like. This is one of the most important aspects of our system that not only enhances the customer experience but can also help increase revenue for the business.

While browsing through the menu, the customer may add items to his/her order. This process is commonly known as “building the order”. After the order is built and read, the user may go ahead and place the order. The staff will automatically and almost instantly be notified about the new order so that they can act on it.

If the establishment allows, the user may even track the status of their order so that the customer may know when to expect their food and drinks to land up on their table.

C. Recommendation Overview

The recommendation algorithm is an innovative feature that we aim to include in our menu. When most tablet menus provide the customers with only a simple menu, this system

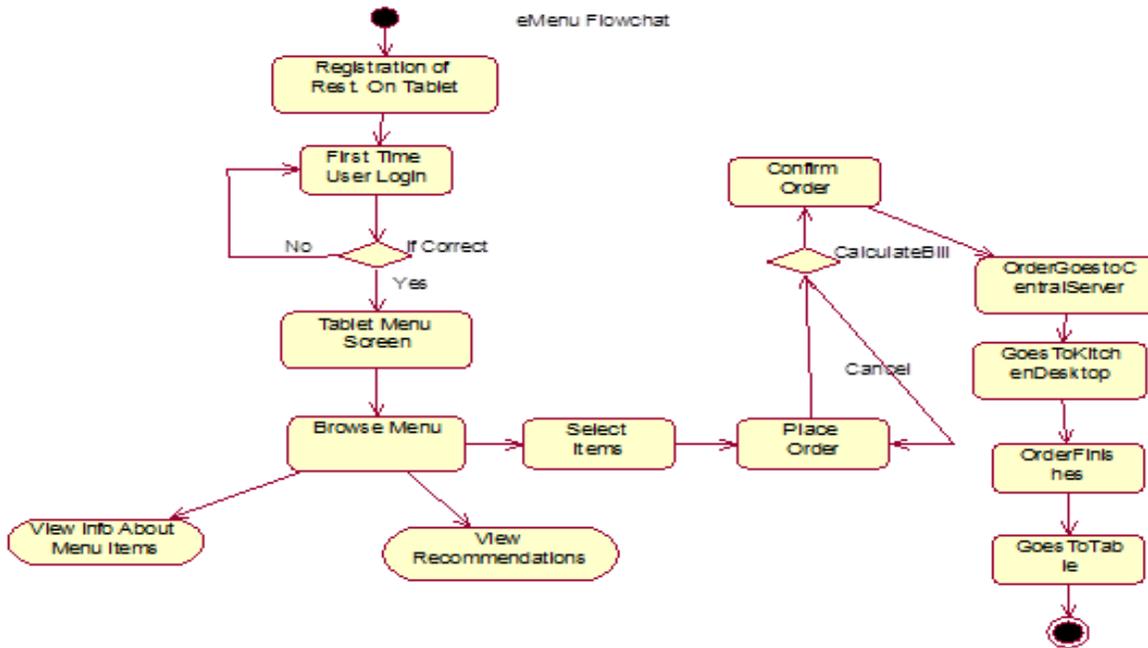


Fig.2. Flow chart of the tablet application

will provide recommendations which will make it easier to build an order considering what other customers have ordered

previously or the similarities between various dishes. Recommendation systems using sets were considered [2]. We finally decided to use the below methodology, which has been discussed in an earlier study [3].

D. Methodology For Recommendation System

The algorithm mainly has 5 parts:

1. Users - a certain number of people are made to rate individual food items.
2. Entities - the food items.
3. Value Dimensions - the categories that are formed to rate the food items e.g. Price, quality, meat content, etc.
4. Belief System - is personal to each user & allows to tell the system what ideal value they want each value dimension to have.

Fig 1: Flow chart of the tablet application

Each User rates a food item on a scale of 1 to 5 with respect to two things:

1. User’s ideal value dimension
2. The weight or the importance of that value dimension

With the food items set in place, our next task was to analyze the various attributes that were associated with each food item. We applied a food item click counter to the entire data set, which produced a list of the most viewed food item.

After listing out the Top N clicked food-items by the customers, we apply normalization to the retrieved list of food-items, to filter out redundant clicks.

We further investigate the levels of commonality that existed between various pairs of food items. Jaccard’s coefficient was used to calculate the degree of similarity.

The Jaccard index, also known as the Jaccard similarity coefficient (originally coined coefficient de communauté by Paul Jaccard), is a statistic used for comparing the similarity and diversity of sample sets.

The Jaccard coefficient measures similarity between sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets:[12]

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

Collaborative filtering-based recommender systems rely on information derived from social activities of the users, such as opinions or ratings, to form predictions or produce recommendation lists. Existing collaborative filtering techniques involve generating a user-item matrix, from which recommendation results could be derived Content-based filtering focuses on the selection of relevant items from a Content-based filtering focuses on the selection of relevant items from a large data set, things that a particular user has a high probability of liking. This involves training the data set with machine learning techniques. Clustering involves sectioning the data set into particular sets, each of which correspond to certain preference criteria. Also, typical recommendation systems output their results either as predictions, a numerical ranking value corresponding to a particular item or recommendations, a list of relevant items .The conventional approaches to computing similarity involve

the use of two popular techniques: Pearson correlation similarity given by Equation 1

$$\text{Pearson Correlation Similarity} = \frac{\sum_{j=1}^l (r_{ij} - \bar{r}_i)(r_{kj} - \bar{r}_k)}{\sqrt{\sum_{j=1}^l (r_{ij} - \bar{r}_i)^2 \sum_{j=1}^l (r_{kj} - \bar{r}_k)^2}} \quad (1)$$

and cosine similarity, given by Equation 2.

$$\text{Cosine Similarity} = \sum_j \frac{r_{ij}}{\sqrt{\sum_j r_{ij}^2}} \frac{r_{kj}}{\sqrt{\sum_j r_{kj}^2}} \quad (2)$$

We used Jaccard's coefficient in our models,

to compute similarity. by generating a list of top-N food-items clicked, then applying a re-ranking algorithm to this list. In this manner, the recommendation list becomes more refined by using the information derivable from the original data set.[11]

E. Comparison of Existing vs. Proposed systems

TABLE I. COMPARISON OF SYSTEMS

	<i>Existing System</i>	<i>Proposed System</i>
Operating Systems	iOS	Android
Communication Channel	Communication between the customer & the waiter	Communication between customer & the terminal
Customer identification	RFID used to identify customers Additional hardware required	User accounts maintained in the database server
Server location	Localized server	Centralized server
Recommendation System	Recommendations not implemented	Recommendations implemented
Exclusivity	Exclusive to every establishment	Can be extended to be used by multiple establishments

F. Technologies

The technologies being used to build the system are cutting and largely open source. Open source technologies help in keeping the costs in check, thus enabling the various establishments to use this setup without any cause of concerns with regards to the costs involved.

Amazon Web Services has a very effective and inexpensive service known as Elastic Cloud Compute which allows one to setup servers on the fly with the specifications one requires. We will be using the same (or similar) service to keep the costs down and maintain scalability.

The server uses a software known nginx. nginx [engine x] is an HTTP and reverse proxy server, as well as a mail proxy server, written by Igor Sysoev. For a long time, it has been running on many heavily loaded Russian sites including Yandex, Mail.Ru, VKontakte, and Rambler. According to Netcraft nginx served or proxied 12.81% busiest sites in February 2013.

The database uses the ever popular MySQL as its DBMS. The web services mentioned earlier are powered by a web framework called Symfony that uses PHP and an ORM called Doctrine. OAuth and SSL can and will be used to implement security.

The tablets supplied to the patrons will be running Android OS.

G. Features Overview

In this section we won't go into the detailed features of the system, but instead take a bird's eye look at the same.

1. Intuitive, Beautiful & User friendly

The end users, i.e. the restaurant customers, will have maximum interaction with this system. This interaction will mostly occur through the tablet application.

Unlike most applications that have a targeted user base, our application will be used by all & sundry. It could be used by an 8 year old or an 80 year old person, from varying cultures and background. This ensures the need for a design that is intuitive, user friendly and beautiful thus making it a pleasure to use.

2. Scalable

The system has the potential of being used by millions of users at any given time, with possible peaks at various rush hours at the establishments this may be implemented at. Thus, the system will have to be scalable and should be able to accommodate as many users and as much data as required.

3. Secure

Since private information about so many businesses and individuals will be stored in our database. We will need to ensure that the whole system - the web service, the app, the database as well as the server be secure from external as well as internal threats.

H. Assumptions

The success of the proposed system is based on the following assumptions that:

- End users should be able to use tablets.
- Network connection will always remain stable.
- Security of server is maintained.
- Tablets should work fault free.

I. Risks

The above mentioned assumptions, give rise to a bunch of risks. Those risks are mainly the following:

- Intrusion of confidential data

- Server goes down
- Tablets suffer a defect
- End users are not able to use the tablet devices

J. Advantages

Most, if not all, of the current tablet based menu systems use Apple's iPad device. This is mostly due to the fact that the current generation of iPads were the first tablet devices to feature high resolution displays and the perceived brand recognition and status of the brand that is Apple.

It is a well-known fact that Apple devices sell at a premium. So, if an establishment were to implement this system, they would unfortunately have to pass on the extra costs to their customers that they incurred in the procurement of these devices.

High resolution Android tablets are now available at extremely economical rates with varying feature sets. Since all our application needs is a Wi-Fi connection and good display, the costs of the tablet devices can be kept to a minimum.

The centralized servers can allow the sharing of customer data between restaurants, if allowed to do so explicitly by both the customer and the business, then this can help in providing better recommendations and user experience to the patrons. This is really not possible in the stand alone systems that currently exist.

A high-quality service system should be customer-centric, i.e., it should immediately recognize the identities, favourite meals and expenditure records of customers so as to provide customer-centric services. Therefore, using advanced technologies to improve service quality has attracted much attention in recent years.

In recent years, various product recommendation systems have been developed to enhance customer satisfaction and perceived value. Defined as a system which recommends an appropriate product or service after learning the customers' preferences and desires, recommendation systems are powerful tools that allow companies to present personalized offers to their customers. Extracting users' preferences through their buying behaviours and histories of purchased products is the most important element of such a system.

The mobile device-based service unit enables instant transmission of customer orders via Internet to the kitchen for meal preparation. In addition, the expenditure information can be sent to the cashier for bill pre-processing. The restaurant managers can access the database to evaluate the business status anytime and make appropriate redeployments for food materials. Notably, all ordering and expenditure information is digitized for database storage, which allows restaurant owners to consider discounts or promotion to customers based on expenditure statistics. Customers can thus appreciate high-quality service, which in turn highly promotes enterprise image and increases business revenue for the restaurant.

For our backend server, where the admin and the kitchen frontends reside, uses the Amazon Web Service Elastic Cloud

Compute (AWS EC2) servers. The advantages of using these EC2 servers are manifold and a few are listed below for reference.

- You can easily scale vertically (upgrade to a larger 2 or 4 CPU instance) or horizontally (add instances).
- You get a lot of RAM. A small instance has 1.7Gb. A large instance has 7.5Gb.
- You can more easily and cheaply leverage S3 for backup, storage, and serving of large files.
- They have excellent bandwidth.
- Less worry about hardware failure. Failures do occur, although it should occur less often than a dedicated server, and recovery is much easier.
- No CPU throttling or other usage limitations. At a shared host, it is common practice to kill long running scripts that are using significant CPU.
- Dedicated IP address. It's yours and yours alone, as long as you keep your instance running.

IV. RESULT

The following screenshots show our tablet application (earlier referred to as the patron frontend) running on a virtual android machine on a standard PC (Intel i5, Windows 7).

For our test environment we used a distribution of Android known as AndroidVM. This distribution is available in different versions for phones and tablets. We are, as visible, running the tablet version of the same. The Android version being used is 4.1.1, although the application is theoretically compatible with versions 2.2 onwards. This compatibility has not been tested

A. Testing

We performed the following Test

Test-id	Test Case	Expected Result	Actual Result	Pass/Fail
01	Does the total cost change when items are added/removed from the cart?	Yes	Yes	Pass
02	Does the total cost change when the quantity is reduced/increased?	Yes	Yes	Pass
03	Do all the images load on clicking?	Yes	Yes	Pass
04	Is it easy to navigate through all the categories e.g. Desserts/Veg/Non Veg/Cocktails,etc.	Yes	Yes	Pass

05	Can an order be cancelled within a selected time frame(eg. Time frame= 10 mins)	Yes	Yes	Pass
06	Is it possible to order more food items after specific time intervals?	Yes	Yes	Pass
07	If discount offers on specific card payments can be availed?	Yes	Yes	Pass
08	Does order get placed & customer gets billed if there is a loss in connectivity	No	No	Pass

All the above Test cases are executed in Aakash Tablet with configuration : 4.0.1 ICS , 1GB Ram, 1Gz Processor

B. Test Results Screenshots

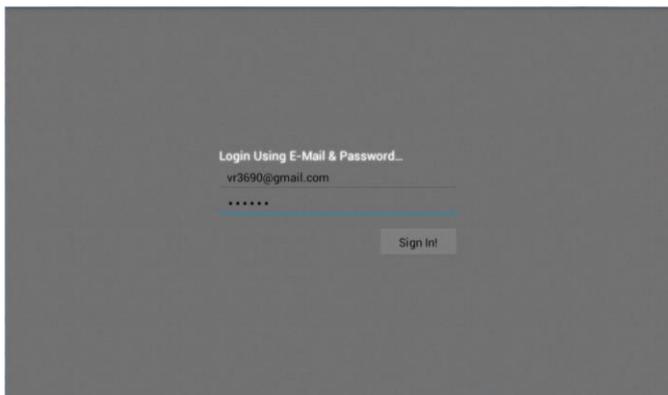


Fig.3. Login screen of Android app

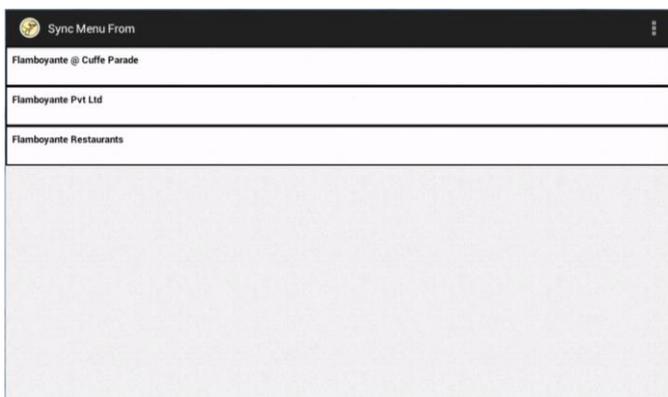


Fig.4. Choose establishment

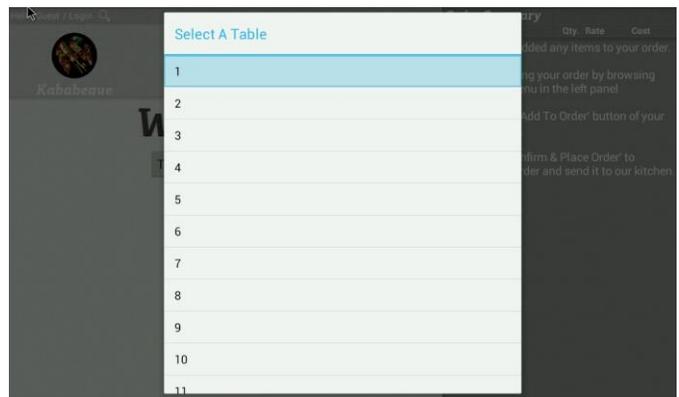


Fig.5. App Welcome Screen



Fig.6. Browsing Menu



Fig.7. Add items to order

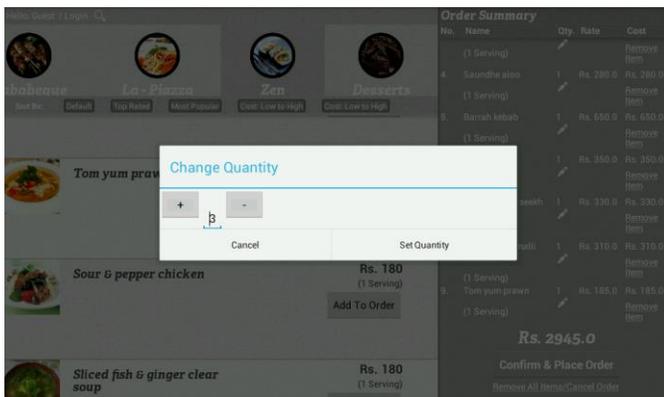


Fig.8. Edit quantity

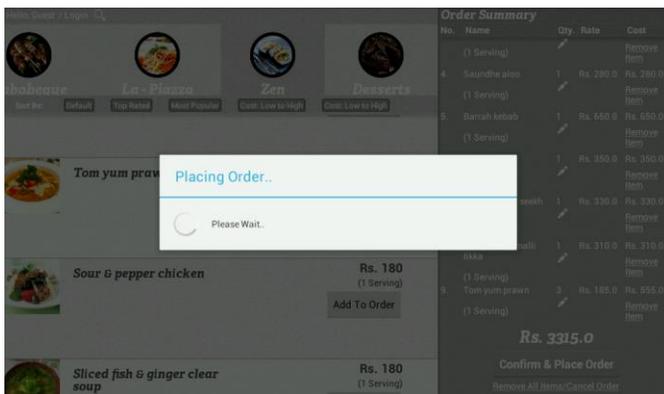
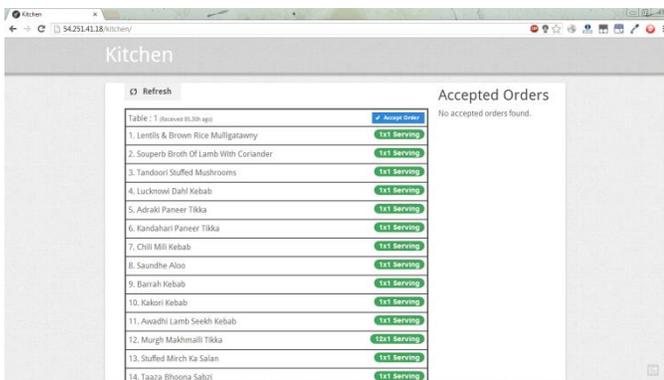


Fig.9. Place Order



Order received at kitchen

Fig 10 depicts the kitchen interface running in the latest version of the Chrome browser. The main aim of the kitchen frontend is to display the latest orders, place by the customers using the patron frontend, in a comprehensive manner. An AJAX query is fired at a pre-defined regular interval to check if any new orders have been placed and the same is displayed.

ACKNOWLEDGMENT

First & foremost we would like to thank our mentor & guide Dr. Vinayak Ashok Bharadi, for his constant support & guidance. His active cooperation & involvement have helped us through the various stages of project development.

We would also like to express our gratitude to Dr. Kamal Shah, H.O.D., Information Technology for her thoughtful recommendations & suggestions.

Last but not the least we would like to thank Dr. B.K. Mishra for extending his support.

References

- [1] Tan-Hsu Tan, Ching-Su Chang, Yung-Fu Chen, Yung-Fa Huang, Tsung-Yu Liu, "Developing an Intelligent e-Restaurant With a Menu Recommender for Customer-Centric Service", Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions.
- [2] Tomoko Kashima, Shimpei Matsumoto, and Hiroaki Ishii, "Recommendation Method with Rough Sets in Restaurant Point of Sales System", PIMECS 2010 Vol III
- [3] Ali Akhtarzada, Cristian S. Calude and John Hosking, "A Multi-Criteria Metric Algorithm for Recommender Systems", CDMTCS-400
- [4] <http://www.forbes.com/sites/tomtaulli/2012/01/13/will-tablets-kill-the-traditional-menu/>
- [5] http://www.emenuworld.com/emenu/features_22
- [6] <http://www.deplod.com/tablet/ipad/>
- [7] <http://aptito.com/Why-Aptito>
- [8] <https://www.menupad.com/features>
- [9] <http://www.waitersrace.com/tablets-and-e-menu-debates-around-the-restaurants-tables>
- [10] <http://www.touchbistro.com/features>.
- [11] <http://www.acsu.buffalo.edu/~suchismi/iRec.pdf>
- [12] http://en.wikipedia.org/wiki/Jaccard_index