CHAPTER 1

INTRODUCTION

Daily life turning digital is fascinating all the way. Things are getting better in their level of accessibility on a technical mode. And here it is to present one bit of technicality on a learner's view. Chat bots! Running the digital market outside successfully and smoothly.

A chatbot is a computer program designed to simulate conversation with human users, especially over the world wide web. It's not at all a new fact to state that the chatbots have been garnering all the buzz these days, especially in 2017-18. As a matter of fact, you see them in abundance in Facebook Messenger and also, these bots are deployed on websites for better interaction with the guests or customers.

In reality, the purpose of a chatbot is to support and scale business teams in their relations with customers and redefine the existing guest experience. By doing this a restaurant business can save loads of money which is why many business owners are adopting this technology and replacing the conventional human live chat agents. And given the fact that these bots can be placed in myriads of platforms like Facebook Messenger, Slack, Telegram, mobile application, SMS based or on your own website gives you the potential to reach a bigger audience without digressing.

Till now restaurants have been using apps for all the work which are not user friendly. But with increase in use of chat bot many restaurants are thinking of using chat bot app instead off traditional apps. Restaurant chat bots are the modern-day equivalent to a concierge, and with one of the latest bots, your customers no longer need to make a call to reserve a table, wait for staff to attend to them or wait in line for tables to free up. Restaurants don't need to have a exclusive service executive for the customers either.

In restaurants, the consumer desire for the speed and accuracy of information which is greater than ever! An entire generation (millennial) of guests prefer not to pick up the phone to reception or talk to a human being. And also, they're not accustomed to waiting for someone to answer a simple question or to wait for something to be distributed. Here, chatbots can provide a single product solution, which addresses this type of guest across multiple different scenarios in multiple languages with very limited resources required from the hotel point of view therefore delivering a better experience whilst saving on staff costs. Many popular hotel chains are also

incorporating chatbots to redefine their customer experience by engaging the guest in a better and effective way. It promotes saving of time on a scale by engaging bots to the process and staff to their work as soon as possible. This leads in fast preparation of food material and on table serving as well. It proves to be advantageous for the humans in all the senses. Undoubtedly it gives a way to less human to human interactions, but quicker work done along with the system going digital with the customers or say eaters. This way it hikes up the impression of the business on the crowd by increasing in demand and sort of makes it profitable enormously likable to popularity. It fascinates people and grab their attraction too on a real high level.

Restaurant bot is a chat bot that will be used for faster interaction with the customers. For this purpose, a mobile app will be used. The mobile app will contain the chat bot. Using the restaurant bot will be very beneficial for restaurants as it will save the time of waiters. Also, it is much more user-friendly than other apps. Customer will order the dishes using the bot. For all the chat it will use NLP and separate the meaningful information from it. The final order item will go to the admin. Chat bots in restaurants need to be perfectly synchronized with the marketing and other customer-oriented efforts. Bots can parallelly serve as an intelligence gathering tool which assists a restaurant in understanding their customers.

Chatbots can easily replace all the existing mobile apps for hotels. Chatbots stay with the guest from pre-booking to post-stay and provide hyper-personalized recommendations and solves query real time.

CHAPTER 2

LITERATURE REVIEW & RELATED WORK

The first chatbot ever was developed by MIT professor Joseph Weizenbaum in the 1960s. It was called ELIZA. You'll read more about ELIZA and other popular chatbots that were developed in the second half of the 20th century later on. In the year 2009, a company called WeChat in China created a more advanced Chatbot. Since its launch, WeChat has conquered the hearts of many users who demonstrate an unwavering loyalty to it. It is a highly thriving social media platform. Through its platform, it has made it easy to create very simple chatbots. It has grown to be an example of the most favored ways for marketers and employers to reduce the work they do as they interact with customers online. In 2018 after bots for messenger started the use of chat bot in business platform which is growing rapidly.

Today, 27% of consumers are interested in artificial intelligence support tools. The idea of using human language in communication with machines arose in the early '50s. However, at that time, people could not yet imagine the machines that could actually react or work like humans. In the past few decades, though, things have changed significantly. People still have unrealistic expectations about artificial intelligence, but we can say that humanity has moved a step closer to interacting with machines. Today, AI technology is used to provide virtual assistance in a range of different industries, including healthcare, business, education, and finance.

1.4 billion people are using chatbots. The advancement of AI has provided humanity with a stack of useful tools and resources. Chatbot growth has been prominent across a number of industries, to the point where 1.4 billion people now use them on a fairly regular basis. So, we might have a new answer to the question: What are chatbots? More than ever, they're the way companies help customers ask questions or resolve problems quickly and easily.

Chatbots can answer 80% of the standard questions. Chatbot market statistics show that one of the reasons this technology is becoming more and more popular is that chatbots can answer most questions users might throw at them. It's still important to have some trained customer support professionals for more complicated questions. But for everyday issues, a chatbot service reduces costs and speeds up response time. This allows customer service agents to work on more challenging tasks and take a big-picture approach.

In 2017, 34% of consumers preferred to communicate with artificial intelligence in an online retail situation. Why? Because consumers like to get the information they're asking for immediately. Bots are the best at providing fast and reliable service; they're always on and they're programmed to answer the most common questions immediately, which gives them an advantage over email or social media.

67% of customers used chatbots in the past year. With two-thirds of customers having used chatbots in the past year, this technology has become mainstream. We live fast, online lives that are dominated by our phones and tablets. For the most part, that works in our favor. But sometimes we need help, and we need it as quickly as possible. This is when chatbot customer support steps in. Chatbot statistics for 2019 show that people are adjusting well to automated support.

64% of internet users say 24-hour service is the best feature of chatbots. Why do we love chatbots? First, we like their quick response and prompt service. Second, we want answers at any time of the day and we enjoy their accessibility. This is why chatbots are so popular. As many as 64% of internet users see round-the-clock support as the biggest benefit, according to the newest chatbot industry statistics provided by Drift.

37% of people use a customer service bot to get a quick answer in emergencies. Among customers who use chatbots, 37% use them for getting answers in case of an emergency. It's also interesting to note that 35% use chatbots to resolve problem and the same percentage do so to get detailed answers or explanations.

55% of people who use chatbots would expect them to provide instant responses and answers to simple questions. Customer service trends show that chatbots' benefits are numerous and customers are aware of this. Indeed, using a customer support chatbot gets you instant answers and asks you short, direct follow-up questions that are easy to understand. For speed and convenience, chatbots provide the perfect solution.

Restaurant chat bots are the modern-day equivalent to a concierge, and with one of the latest bots, your customers no longer need to make a call to reserve a table, wait for staff to attend to them or wait in line for tables to free up. Restaurants don't need to have an exclusive service executive for the customers either. Bots can be programmed to carry out a myriad of tasks ranging from answering FAQs, making a reservation, ordering food or processing payment. The bots can carry out these tasks in manner similar to a service executive, difference being — it can execute

round the clock with zero downtime. Numerous task-oriented chat bots have been developed for commercial and recreational purposes.

That apart, while the staff focuses on preparing and serving food, chat bots can engage with the customers by answering questions related to open and close times, reward points or whether if the restaurant is open on a public holiday. The use cases of chat bots in restaurants rely heavily on the kind of experience restaurants want to offer their visitors.

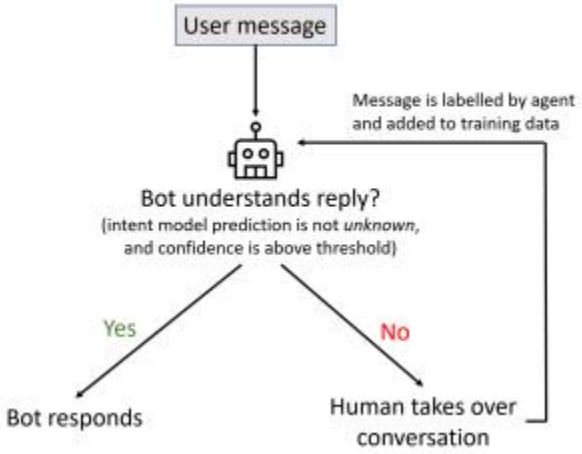


Fig 2.1- Flow of Bot and Human Conversation

Till now restaurants have been using apps for all the work which are not user friendly. But with increase in use of chatbot many restaurants are thinking of using chatbot app instead off traditional apps.

CHAPTER 3

PROBLEM STATEMENT

Today when we visit restaurants a lot of time gets wasted in manual work. Also for restaurant owners it is getting harder and harder to find good waiters. Sometimes many restaurant owners has to opt for child labours which is against the law. Also it is a time consuming process. So the process needs to get digitalized. For this mobile apps can be used. But mobile apps are not so customer friendly and it will be difficult for customers to interact on them and use them. Many times such applications are little non-understandable to the users, which creates difficulty for them while using it. We need a better user friendly solution which is a Chatbot.

To improve the accuracy & speed in restaurants by digitalizing the interaction between customers and restaurants.

CHAPTER 4

SYSTEM ANALYSIS AND DESIGN

System Analysis is the process of gathering and interpreting facts, diagnosing problems and the information to recommended improvements on the system. System analysis is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of the any system development process. The system is studied to the detail and analysed. The system plays the role of the interrogator and dwells deep into the working of the present system. The system is viewed as a whole and the input to the system are identified. The outputs from the organizations are traced to the various processes. The conclusion is an understanding of how the system functions.

4.1 Existing System

Traditional method that is commonly been used in hotels is by taking the customer's orders. The food ordering system was entirely a manual process which involved waiters, pen and paper. The waiter had to note down orders from customers, take these orders, update them in records and again make bill. Even though this system is simple it may involve human errors in noting down the orders. There are many reasons leading to the feeling of dissatisfaction including being entertained late in terms of order taking by the waiter and meals serving.

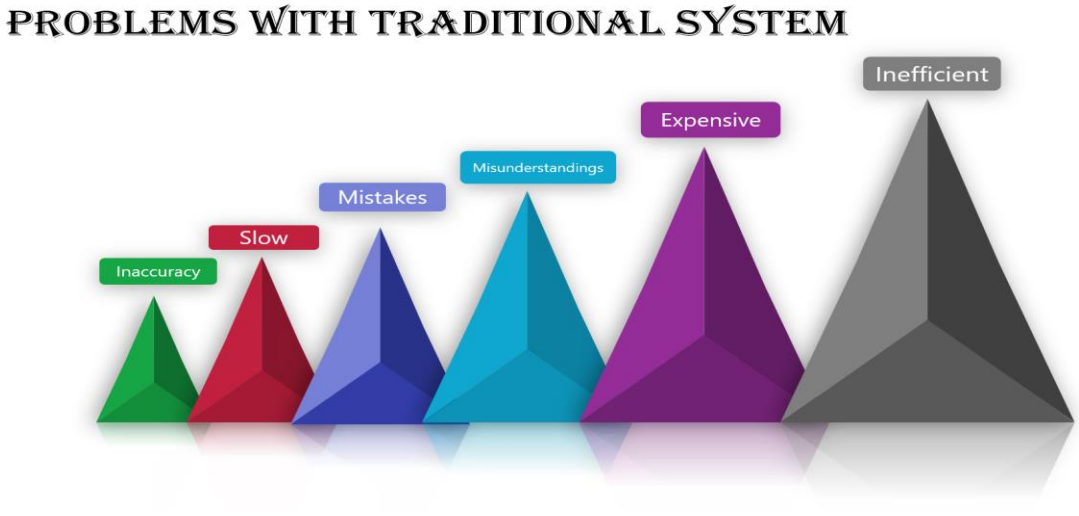


Fig 4.1 Graph of Problems with Traditional System

As these system involves manual work the process becomes slow. Also customers may need to wait for waiters to take their orders. Even if the no of waiters are more it increases human cost. Also waiters may need to wait if customers order is not decided. This affects the overall working of system making it work slow and inefficient.

4.2 Proposed System

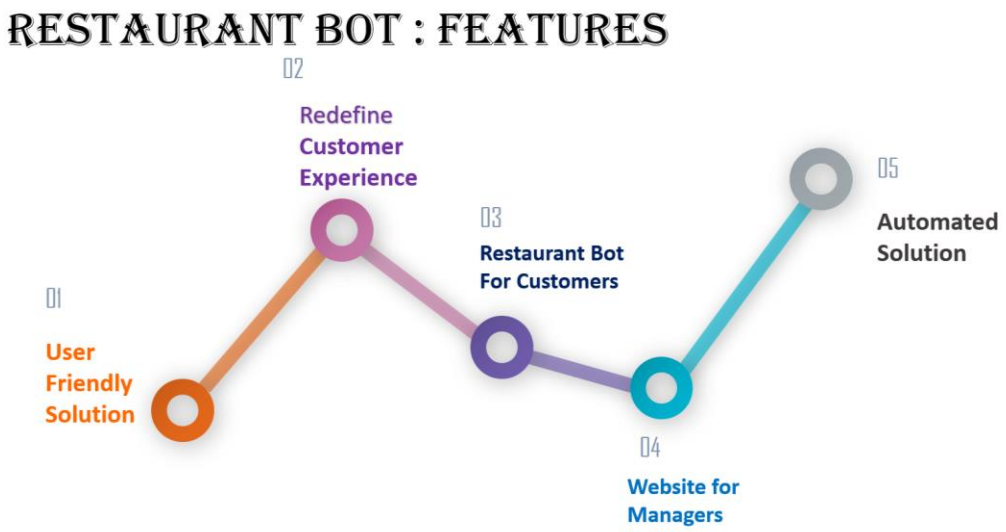


Fig 4.2 Features of Restaurant Bot

The food restaurant with automated food ordering system will be equipped with a user-friendly Chabot, and software for completing the process at the backend. For this system there will be a system administrator who will have the rights to enter the menu with their current prevailing prices. The system administrator can enter anytime in the system by a secured system password to change the menu contents by adding or deleting an item or changing its price. Now when the customer enters the restaurant, customer will place his order with the help of the restaurant bot app, right from the selection of menu items, confirming the order and viewing offers. The customer will select from the food options according to his choice and the system will display the payment amount customer has to make once finished with the order. This system increases quality and speed of service. This system also increases attraction of place for large range of customers. Implementing this system gives a cost-efficient opportunity to give your customers a personalized

service experience where they are in control choosing what they want, when they want it – from dining to ordering to payment and feedback.

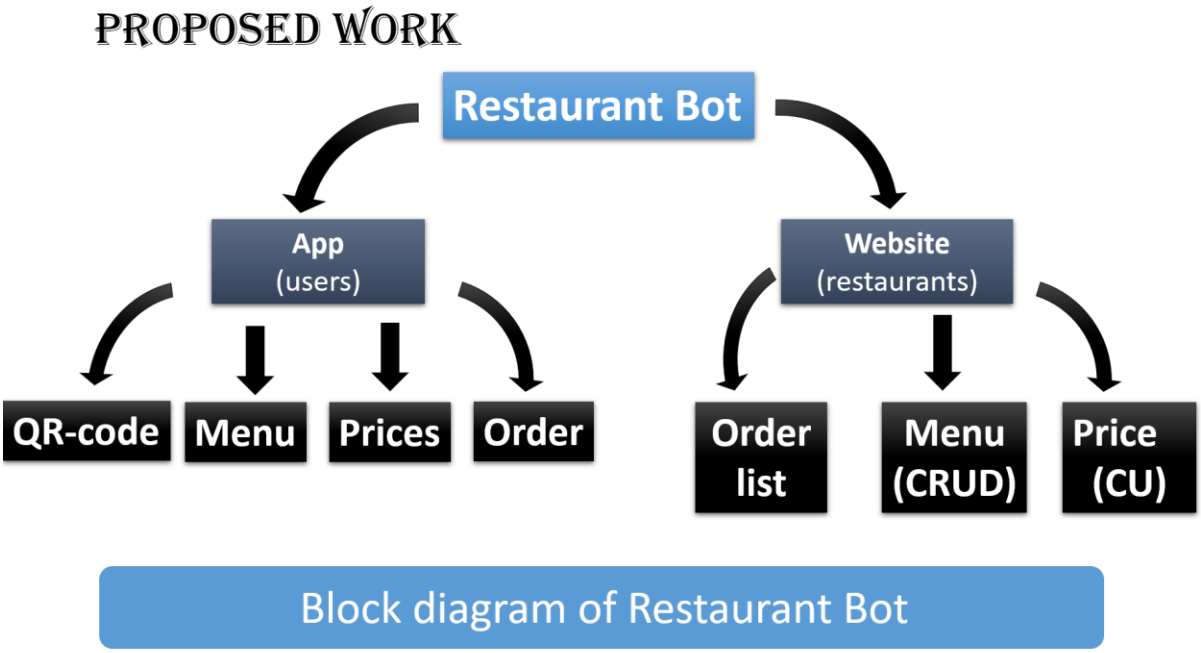


Fig 4.2.1 Block Diagram of Restaurant Bot

This system is restaurant independent. Any restaurant can use its service. Food ordering could increase efficiency for restaurants and caterers by saving time, reducing human errors and by providing higher quality customer service. With the combination of simple design and readily available emerging communications technologies, it can be concluded that this system is an attractive solution.

4.3 System Design

The two design objectives continuously sought by developers are reliability and maintenance.

4.3.1 Reliable System

Reliable system analysis we are concerned with the construction of a model (life distribution) that represents the times-to-failure of the entire system based on the life distributions of the components. There are two levels of reliability. The first is to meet the right requirements. A careful and systems study is needed to satisfy the aspect of reliability. The second level of

systems reliability involves the actual working delivered to the user. At this level, the systems reliability is interworking with software engineering and development. There are three approaches to reliability.

- Error avoidance**:** Prevents errors from occurring in software.
- Error detection and correction**:** In this approach errors are recognized whenever they are encountered and correcting the error by effect of error, by the system does not fail.

  Error tolerance: In this approach errors are recognized whenever they occur, but enable the system to keep running through degraded perform or by applying values that instruct the system to continue process.

4.3.2 Maintenance

The key to reducing need for maintenance, while working to do essential tasks.

- More accurately defining user requirement during system development.
- Assembling better systems documentation.
- Using more effective methods for designing, processing, and login and communicating information with project team members.
- Making better use of existing tools and techniques.
- Managing system-engineering process effectively.

4.3.3 Diagrams with the flow of data

Data Flow Diagram:-

This technique uses graphical tools called Data flow Diagram (DFD) to depict the information flow through the system and the transformation that are applied to it between the input and output stages.

A DFD may be used to represent a system or software at any level of abstraction; the lowest level of abstraction is presented by the source code. DFD may be partitioned into levels that represent increasing information flow and functional details. Level 0 DFD is called a functional system model or context model and represents the entire software elements as a single-bubble with input and output data indicated by incoming and outgoing arrows respectively.

The level is then partitioned to reveal more details upon exploding the context diagram with TOP LEVEL DFD, which is nothing but the representation of the major function in the form of bubbles. In this DFD the external and data stores being used are also shown.

Lower level DFD do not show the external entities. The diagram does not explicitly indicate the sequence of processing rather it depicts the information flow. Initially the functions' that the system was required to perform are identified. Concentrating on each function the subtasks to be performed are further identified.

Following figures shows the data flows in our system.

Data Flow Diagram For Application:

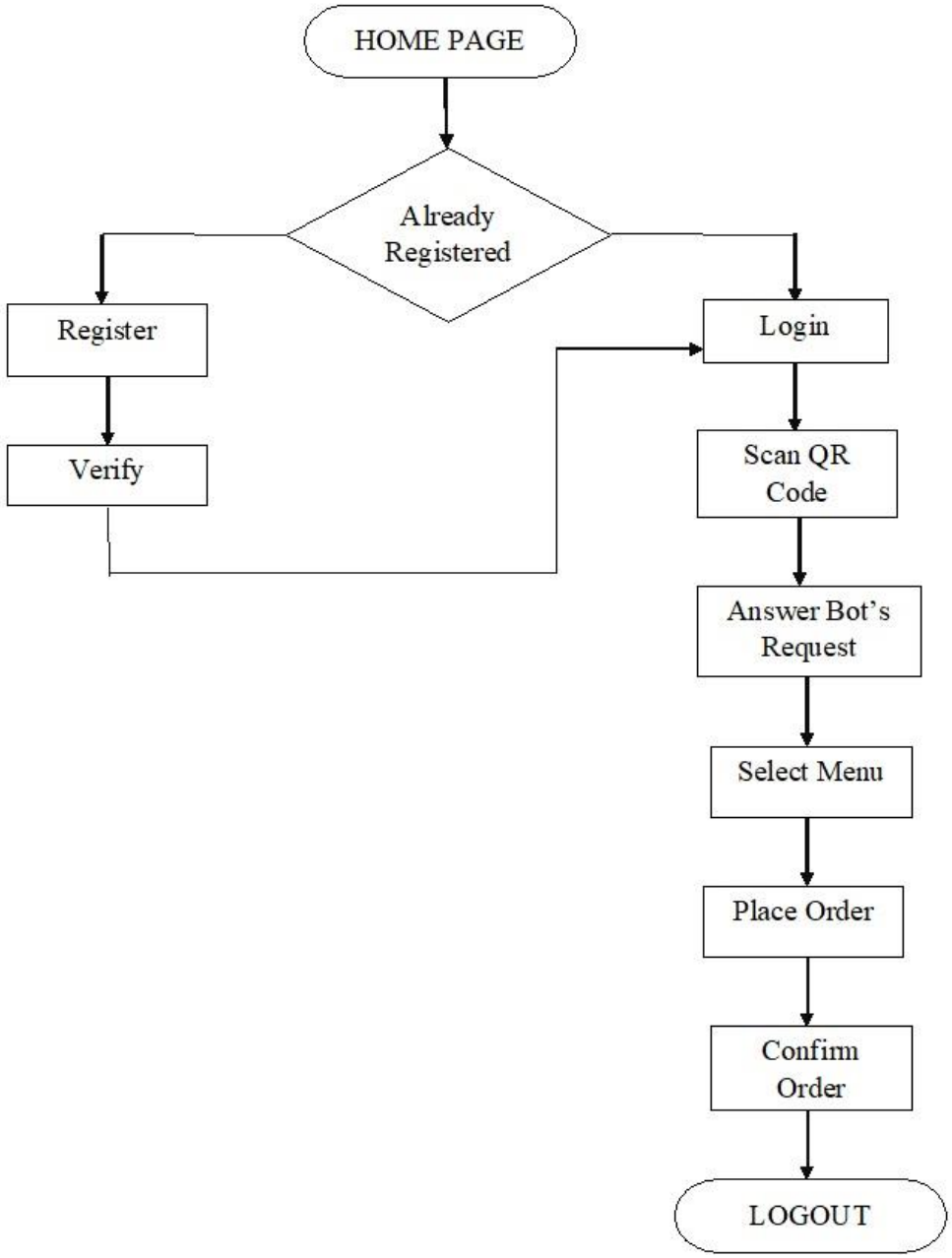


Fig 4.3.3(a) Data Flow diagram for Restaurant Bot Application

Data Flow Diagram For Website:

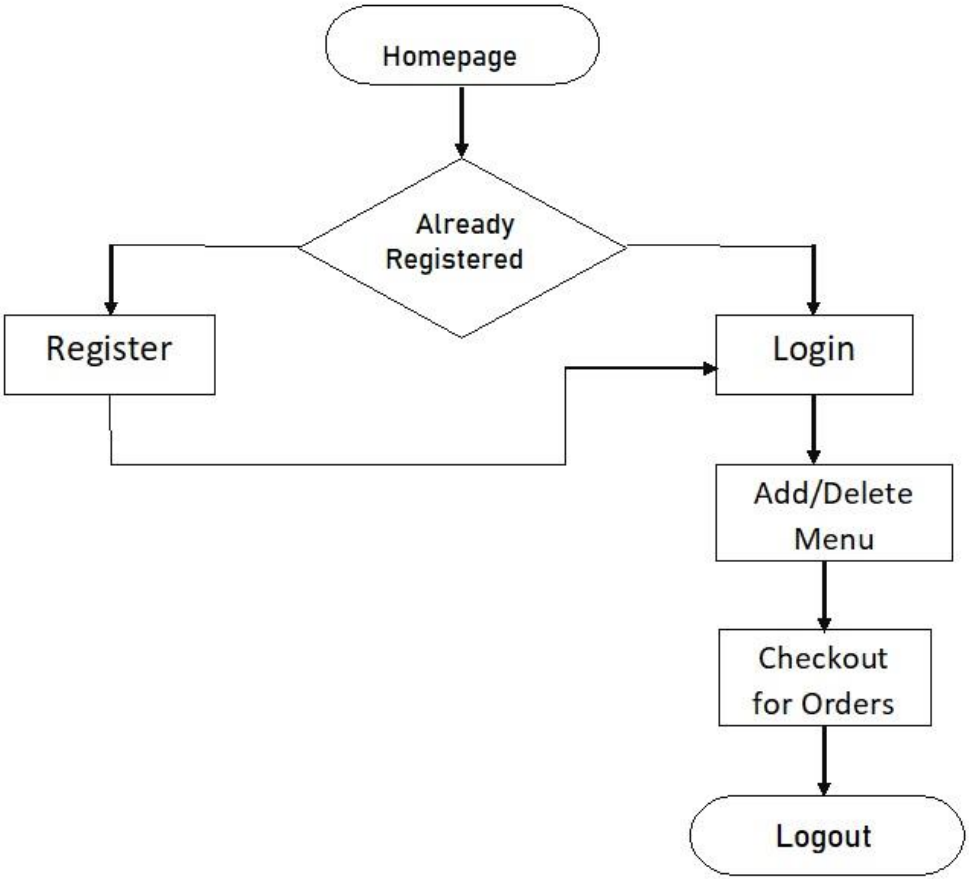


Fig 4.3.3(b) Data Flow diagram for Website

Sequence Diagram:-

UML Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.

Following figure shows the sequence diagram for our system.

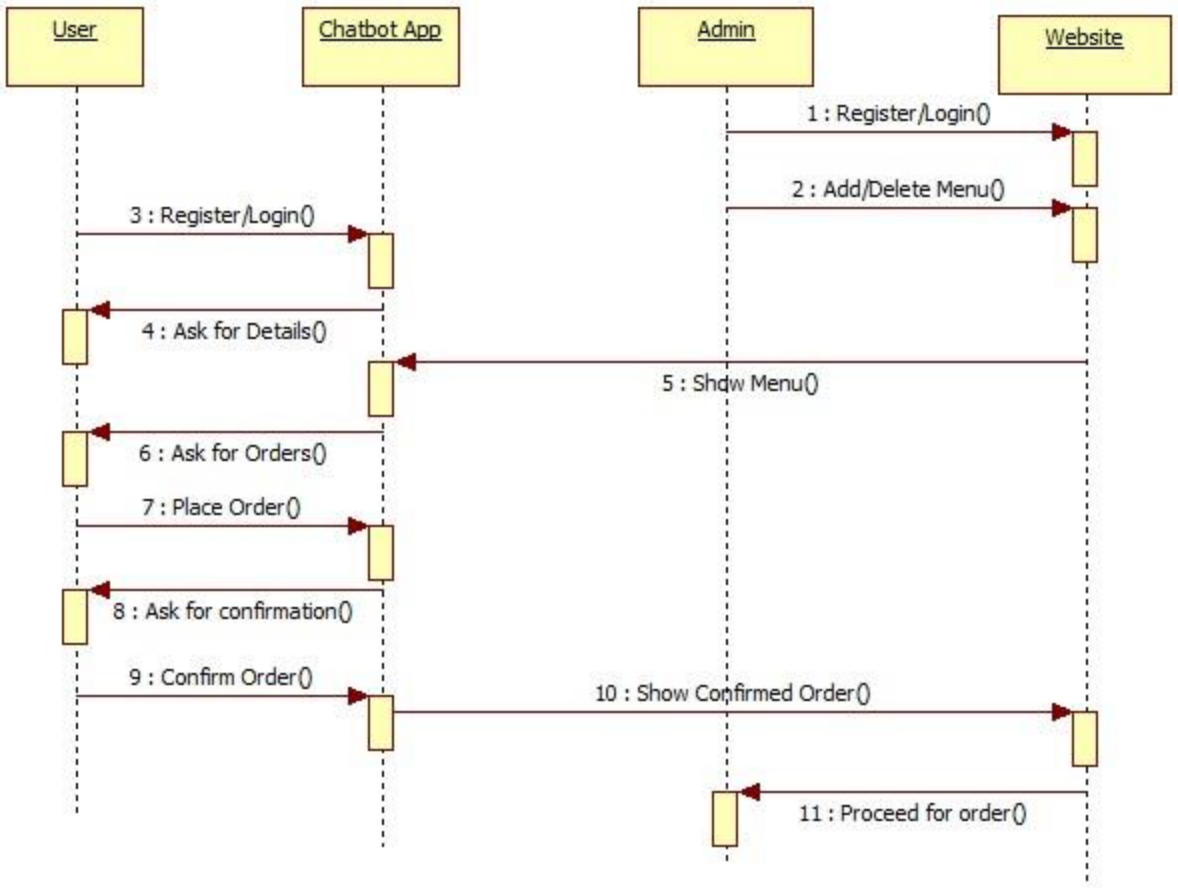


Fig 4.3.3(c) Sequence diagram for interaction between bot, website, user and restaurant.

Use Case Diagram:-

A **use case diagram** at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different **use** cases in which the user is involved.

Following figure shows use case diagram for our app and website:

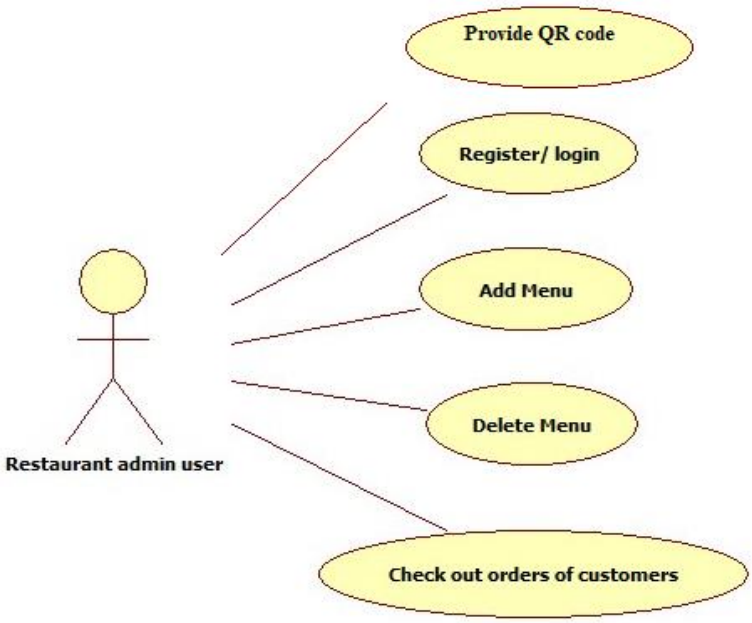


Fig 4.3.3(d) Use case Diagram for Restaurant's Admin User

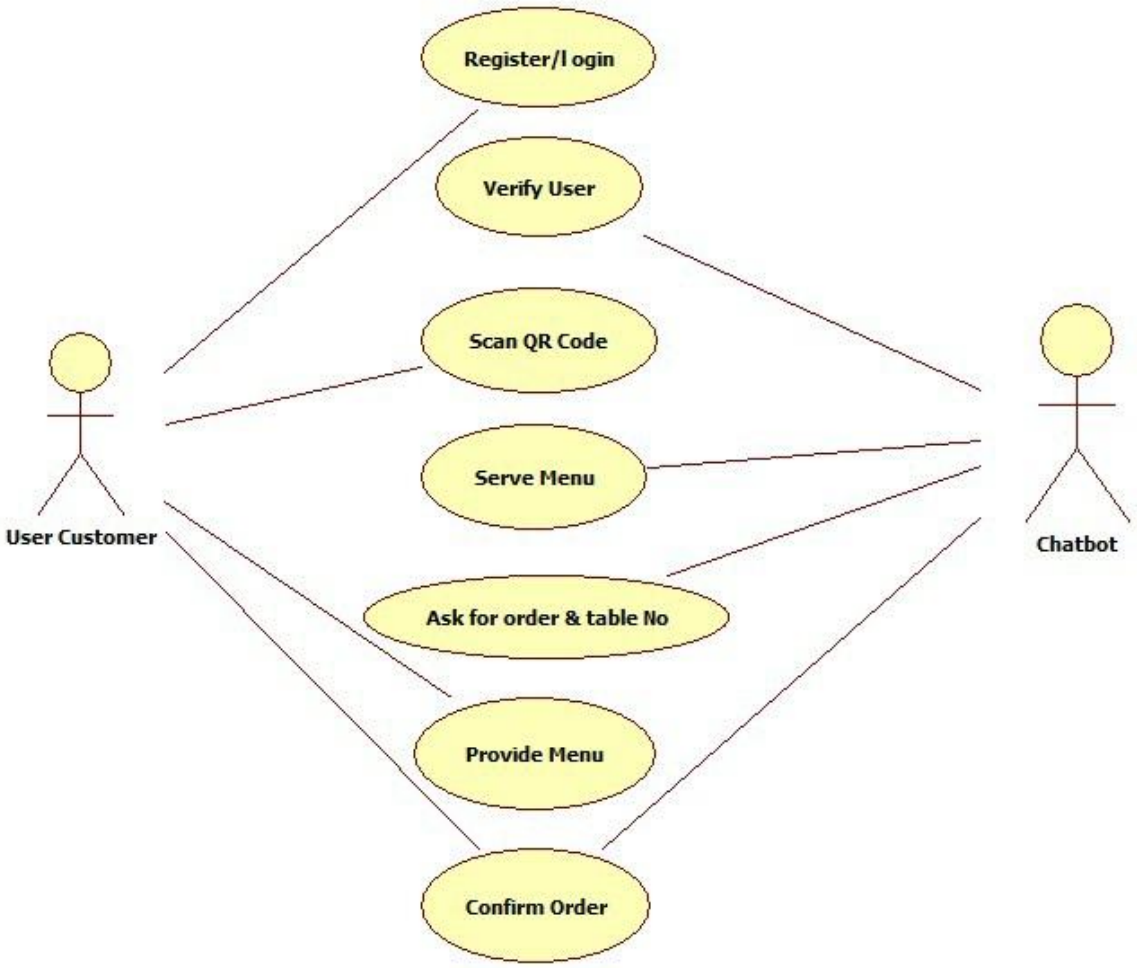


Fig 4.3.3(e) Use case diagram for The Restaurant Bot App

4.3.4. Modules:

- Welcome Screen
- Phone Authentication
- QR Code Scan
- Menu
- Order
- Chatbot

Welcome Screen: Welcome Screen welcomes the customer with a message and request for sign up.

Phone Authentication: Customer sign up need to add his/her mobile number for the authentication or the verification of the user.

QR Code Scan: This is use for scanning your native Restaurant QR Code from the scanning camera of your device.

Menu: Menu consists of list of Items with respectively prices.

Order: The restaurant will display message with your authenticated name and ask you for which table allocated you.

Chatbot: All these above modules are embedded in Chatbot. A Chatbot is a computer program designed to simulate conversation with human users. Basically, it is use for the interaction between Customer and restaurants.

CHAPTER 5

TECHNOLOGIES USED FOR APP DEVELOPMENT

5.1 Flutter

Flutter is Google's UI toolkit for building beautiful, natively compiled applications for mobile, web, and desktop from a single codebase.

Flutter is an open-source UI software development kit created by Google. Flutter is Google's mobile UI framework for crafting high-quality native interfaces on iOS and Android in record time. Flutter works with existing code, is used by developers and organizations around the world, and is free and open source. It is used to develop applications for Android, iOS, Windows, Mac, Linux, Google Fuchsia and the web.

Flutter is a cross-platform development framework built by Google using the Dart programming language (which is also built by Google). This means that a programmer could use Flutter to develop applications on iOS, Android, desktop, and the web (currently in beta) using a single codebase. Flutter is not the first to create a cross-platform solution. There are quite a few out there, with the most popular being React Native, PhoneGap, Ionic, and Xamarin. Flutter is different because it uses Dart, which compiles ahead of time (AoT). This means it compiles to native code without a bridge. Instead, Flutter uses its own rendering engine which makes it run faster.

Why Flutter ?

1. Faster Code Writing

For developers, Flutter means faster & more dynamic mobile app development. We can make changes in the code and see them straight away in the app! This is the so-called Hot reload, which usually only takes (milli)seconds and helps teams add features, fix bugs and experiment faster. Hot reload is also very comfortable in developer-designer cooperation when we want to improve or experiment with an app's look and check the effects on the spot. In other words, with Flutter, your designer or tester can work together with a developer on the UI, making changes – for example, "Put it 2 pixels right" or "Make the animation faster" – and see them immediately.

2. Single Code for multiple Platforms

Developers write just one codebase for your 2 apps – covering both Android and iOS platforms. Flutter doesn't depend on the platform, because it has its own widgets and designs. This means that you have the same app on two platforms. Yet what's important is that, if you want to differentiate your apps – it's possible.

3. Less Testing

If you have the same app for 2 platforms, it means less testing! The Quality Assurance process can be faster. Because of one codebase, the developers write automatic tests only once. What's more, Quality Assurance specialists have less work to do, because they have only one app to check. Of course, if your apps have some differences, they need to be tested on both platforms.

4. Efficient working of Apps

Flutter apps work in a smooth and fast way, without hanging and cutting while scrolling. Flutter is designed in such a way that it will never hang out.

5. Personalized and Customized according to users interest

Flutter is designed to make it easy to create your own widgets or customize the existing widgets. Here you can browse a catalog of Flutter's widgets and view, for example, Material Design widgets and Cupertino widgets.

6. Similar User Interface on all devices

Your new app will look the same, even on old versions of Android and iOS systems. There are no additional costs for supporting older devices. Flutter runs on Android Jelly Bean or newer, as well as iOS 8 or newer.

5.2 Dart

Dart is an open source, purely object-oriented, optionally typed, and a class-based language which has excellent support for functional as well as reactive programming.

Dart is a client-optimized programming language for apps on multiple platforms. It is developed by Google and is used to build mobile, desktop, server, and web applications. Dart is an object-oriented, class-based, garbage-collected language with C-style syntax. Dart can compile to either native code or JavaScript.

Why Dart ?

1. Dart is really Flexible

Dart is a very flexible programming language in that you can write the code and then run it anywhere without any limitations whatsoever.

Mobile apps written in Dart with Flutter are cross-platform native apps; so they can run on both Android, iOS (like React Native, Xamarin, etc.). You can even write web apps and that code can run on any browser. From a developer's perspective, the language's flexibility and portability matter a lot. After all, who doesn't want to be able to write some code and run it on as many platforms as possible?

Coding in Dart feels natural once you are familiar with the general Object Oriented principles. Still, fully functional programming is 100% achievable in Dart. Asynchronous programming with

async/await and “Future” in Dart is consistent and harmonious. If you are a unit tester, then good news for you, because Dart has built-in support for unit testing; no need to add new libraries or frameworks.

2. Dart is pretty easy to learn.

Learning a language takes time, effort and patience. It’s not just about learning the language, but also its ecosystem, the terminologies related to it, getting the proper tools and SDKs for the language, and then moving on to the popular frameworks and libraries available for that language

3. Dart is strongly statically typed and supports type inference.

Type inference refers to the automatic detection of the data type of an expression in a programming language. The ability to infer types automatically makes many programming tasks easier, leaving the programmer free to omit type annotations while still permitting type checking. This makes it easier for developers to transition to Dart regardless of their programming background.

4. Dart’s syntax is simple

The Dart syntax seems extremely familiar even at the first shot. Even if you haven’t seen Dart code before then too it can be easily understood without straining. The Dart language is well structured, so if you already know C, Java, or C# then Dart is going to be very easy.

5. Dart is a robust language.

Dart as a language, is very robust. Having been created by Google, its primary purpose was to leverage C-based Object Oriented Programming languages like C#, and Java. As it is also a general-purpose programming language, it compiles fast and is concise.

6. Dart solved a lot of problems, efficiently.

Dart can be compiled both AOT and JIT. Flutter took advantage of this fact, as using JIT compilation speeds up development (through Hot Reloading and stuff like that) and AOT compilation gives better optimization during release time. Part of the reason Flutter makes devs more productive is that it has Hot Reloading enabled by default. If you are a Java or a Swift dev and you have checked out React Native before, then you know how helpful the Hot Reload feature is. Hot Reloading saves a lot of time, and thus boosts productivity.

5.3 Dialogflow

An end to end developer platform for building natural and rich conversational experiences.

Why Dialogflow ?

1. Delivers natural and rich conversational experiences.

Built-in natural-language processing (NLP) feature imparts artificial intelligence (AI) to the chatbot, thereby enabling it to process the natural language (received through chat or voice) and carry forward the conversation in a natural way. It?s this cognitive capability that distinguishes the AI chatbot from the normal chatbot that operates on simple pre-defined rules as coded into it. A chatbot developer integrates this natural language processing capabilities to applications, services, and devices. The output data can be in the voice or chat format as per the requirement.

2. Understands what users are saying about machine learning

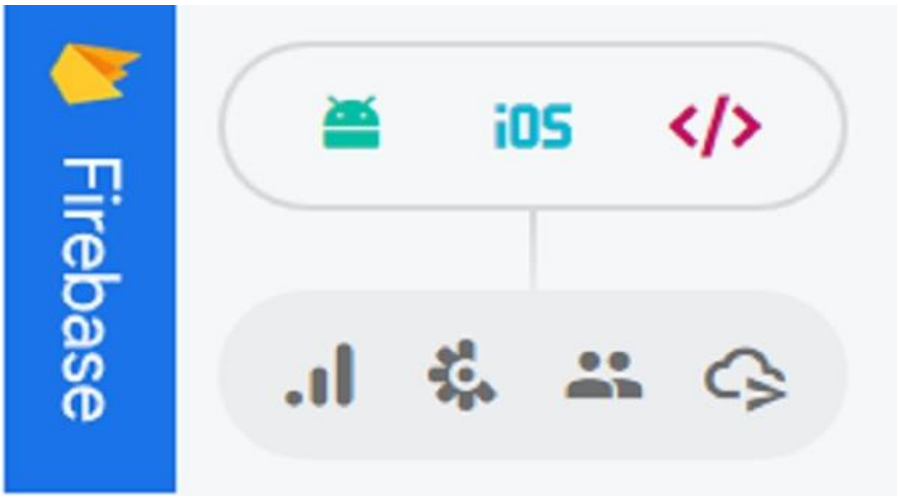
Machine learning makes Dialogflow intelligent enough to predict the hidden intention expressed in the natural input language. A Dialogflow chatbot can map the user's query with the database available with its backend server. The mechanism of mapping is called as Intent. Interestingly, it does so, applying several permutations and combinations. This is a remarkable feature as users ask the same thing in multiple ways, so it's the responsibility of the chatbot to understand and serve them correctly.

3. Offers cross-device support

Dialogflow helps with creating a device-antagonistic chatbot. Thus, it engages with users on wearables, phones, cars, speakers and other smart devices. So, businesses can connect with their prospects or customers anywhere, anytime.

5.4 Firebase

Firebase is Google's mobile application development platform that helps you build, improve, and grow your app and websites.



The Firebase is a Backend-as-a-Service (BaaS) that offers the developers a wide spectrum of tools and services to develop high-quality apps at a much faster pace.

Now, if we were to define the BaaS, it is a cloud computing service model using which the web app and mobile app developers can connect their applications with backend cloud storage and APIs rendered by the backend applications.

Firebase Functionality :

Authentication — user login and identity

Realtime Database — realtime, cloud hosted, NoSQL database

Cloud Storage — massively scalable file storage

Cloud Functions — "serverless", event driven backend

Firebase Hosting — global web hosting

ML Kit — SDK for common ML tasks

Why Firebase ?

1. Synchronizing data with real-time database: With Firebase you can sync the offline and online data through NoSQL database. This makes the application data available on both offline and online states of the app. This boosts collaboration on the application data in real time. Here are some of its benefits.

2. Real-time: Unlike the so-called HTTP requests that work to update the data across interfaces, the Real-time Database of firebase syncs data with every change thus helping to reflect the change in real time across any device in use.

3. Offline: As Firebase Real-time Database SDK helps save your data in local disk, you can always access the data offline. As and when connectivity is back, the changes are synced with the present state of the server.

4. Access from multiple devices: The Firebase Real-time Database allows accessing application data from multiple devices and interfaces including mobile devices and web.

5. Splitting and scaling your data: Thanks to Firebase Real-time Database, you can split your data across multiple databases within the same project and set rules for each database instances.

6. Fast and Secured Web Hosting: The benefit of Firebase Hosting allows you to set-up a single-page, a mobile landing page, web page or progressive web page with ease. It also helps to deliver the content rapidly anywhere. The developers can deploy the web apps as well as static content at CDN (Content Delivery Network).The Firebase hosting also automatically configures the free SSN certificate for custom domains. You can now deploy a local directory to the web with just a single command

7. Firebase Authentication: Nowadays, most of the apps have the login facility and the developer aims to simplify and secure it better. Therefore, the support of Firebase Authentication is there to do that task with an easy sign-in process.

It also provides identity solution for the emails, passwords and other important apps such as Facebook, Twitter or Instagram. The Firebase UI is also flexible, customized and drop-in dealing with the UI flow of the users. There is no compromise from the security point of view.

8. Send Notifications and Messages to Targeted Audiences: The Firebase Cloud Messaging offers you an opportunity to send notifications and messages to your targeted audiences for free across all devices and platforms with the help of battery-efficient connection.

So, if you are interested in sending the push notifications to a specific group of people based on demography and their behavior. In addition, you can choose your own particular time for sending the message, which is more convenient. The developers will be pleased to note that they do not require coding to send the notifications.

CHAPTER 6

TECHNOLOGIES USED FOR WEBSITE DEVELOPMENT

6.1 Hypertext Markup Language (HTML)

HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as<img /> and<input /> directly introduce content into the page. Other tags such as<p>surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

6.2 Cascading Style Sheets (CSS)

CSS is a style sheet language used for describing the presentation of a document written in a markup language like HTML.CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.The name cascading comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

6.3 Bootstrap

Bootstrap is a front-end framework that is developed to support creating dynamic websites and web applications. It is one of the most preferred front-end frameworks as it aids an easy and fast processing to develop a website. It supports all major browsers and fast loading responsive web pages.

Bootstrap consists of HTML and CSS-based design templates for various interface components and is aimed to ease web development. By updating the CSS, you can adapt to modern trends quickly. The developers should concentrate more on interaction components as the bootstrap itself will take care of standard views of data, which can be altered later if you wish to.

Bootstrap is compatible with almost all the latest version browsers such as Internet Explorer, Google Chrome, Opera, Firefox, and Safari. It supports the responsive web design and dynamically adjusts the layout of web pages by considering the characteristics of the device used.

Why Bootstrap?

1.Speed of Development

The speed of development is one of its major advantages. If you want to develop an application or a website promptly, it is imperative to consider using Bootstrap. It helps to save your coding effort by offering less CSS functionality and pre-built blocks of code rather than structuring code from the scratch. Ready-made themes of Bootstrap will help achieve your needs through a faster route.

2. Responsiveness

Bootstrap is equipped with responsive layout and 12-column grid system that help dynamically adjust the website to a suitable screen resolution. The ‘responsive utility classes’ feature of Bootstrap enables you to hide / show a certain section of content for a particular screen size.

3. Consistency

Consistency was the fundamental principle behind the introduction of Bootstrap. It ensures the ultimate consistency regardless of designer/developer, who is working on it. Moreover, the results work uniformly across various browsers and the output remains same.

4. Customizable

Bootstrap facilitates abundant customization and helps developers in designing tailor made websites, according to their specifications. It has the facility to select any feature that is actually needed to create a customized website. With this feature, one can get rid of what they do not require.

6.4 Javascript

JavaScript often abbreviated as JS, is a programming language that conforms to the ECMAScript specification. JavaScript is high-level, often just-in-time compiled, and multi-paradigm. It has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions.

Javascript here is going to provide connection with firebase and all the database functions are performed using it, also conditional testing.

Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it for client-side page behaviour, and all major web browsers have a dedicated JavaScript engine to execute it.

As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working

with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM). However, the language itself does not include any input/output (I/O), such as networking, storage, or graphics facilities, as the host environment (usually a web browser) provides those APIs.

6.5 Firebase

As discussed previously firebase the realtime backend as a service is used for backend.

CHAPTER 7

RESULT

The project works in number of steps discussed below with each ones resulting output:

7.1 Restaurant Registration:- To enable the service of chatbot in restaurant first restaurant should register it on the website. As soon as the registration is completed, a specific QR code of the restaurant is generated which is sent via email to restaurants. Restaurants will print that QR code and will keep it on the tables so that users will be able to access it. The registration page of website asks for restaurant name, its address, email address and password as shown below. The registered data goes on the firebase database.

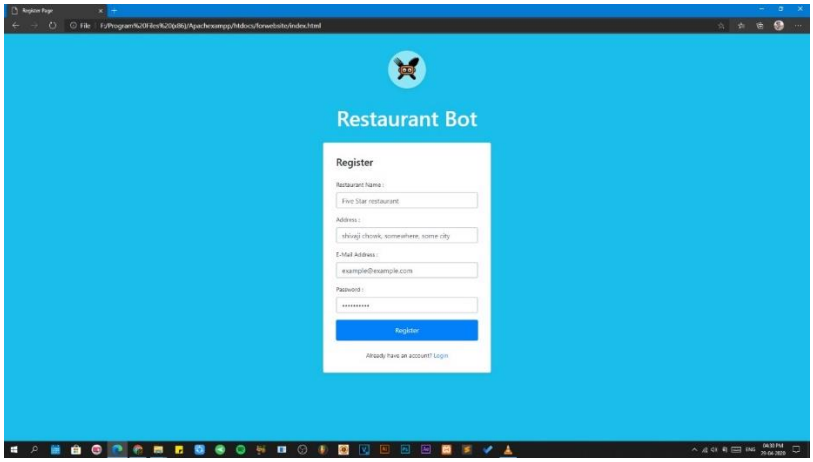


Fig : Website Registration Module

7.2 Restaurant Login Page:- Once the registration is complete the restaurant admin will move to the login page which asks for user name and password.

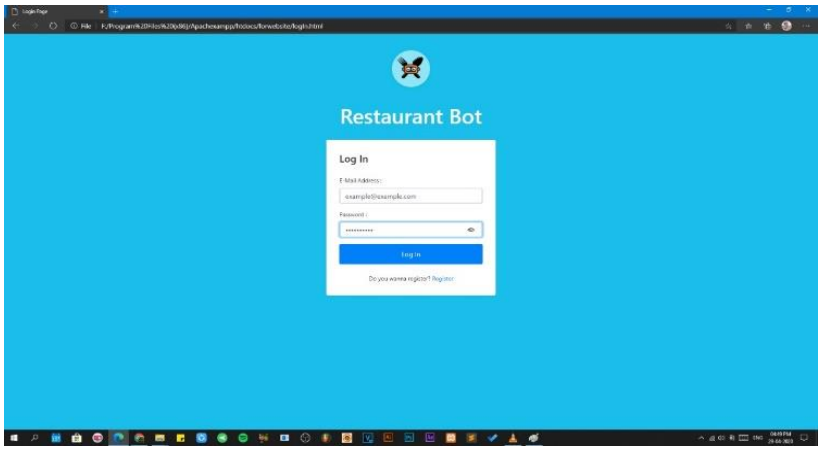


Fig: Website Login Module

7.3 Restaurant Homepage:- After login the Restaurant admin will move to home page from where he will have access to number of functionalities of restaurant. The default home page shows list of upcoming orders. The QR code page shows the QR code which will be used by customers to scan it. Manager can provide QR code to users by downloading it by right clicking on it and putting it on tables. Also, it can be simply printed out. By clicking on manage menu he can add/delete items. The items ordered by the customers will be displayed on the home page. Following image shows the just registered restaurant's home page which consists of customer's name, phone no., table no. and order list.

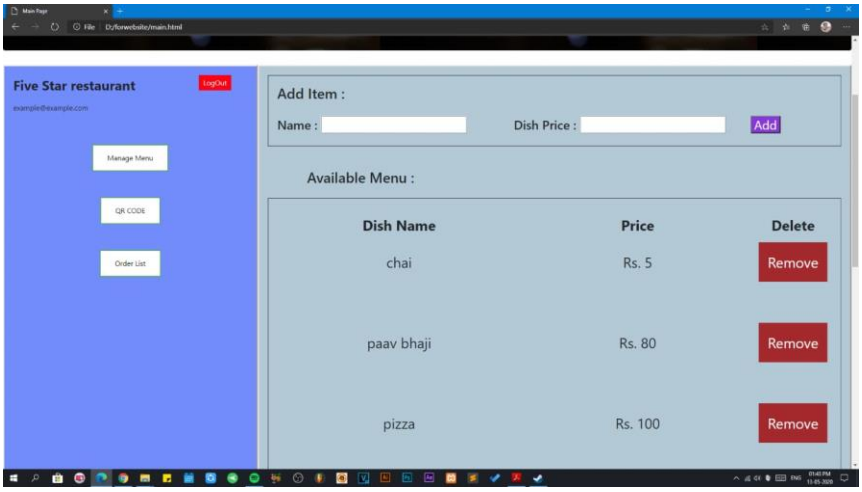


Fig : Restaurant Homepage

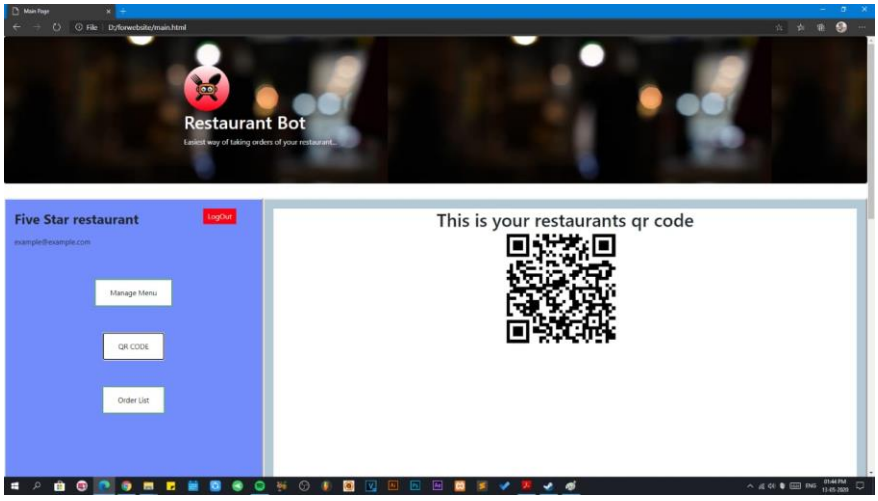


Fig : QR code scanner module

7.4 Customer Registration:- Customer will be provided with a restaurant bot app. As the app is developed using google flutter the app will be available for both android as well as iphone users. As soon as the user enters the app bot will show registration page, As the customer clicks on the sign up button the customer will be shown registration screen as-

Fig : Welcome page of Restaurant Bot

Fig : Sign Up page of Restaurant BOt

7.5 Customer Verification:- When the customer is done, he/she will verify it by clicking Verify button. Now this customers info is saved in the google firebase and it will send a verification code on customers mobile number as-

Fig : Verification Page

Then the customer will verify by entering the OTP. After that he will move to scan page.

7.6 Scan Page:- On this page customer will scan the QR code on the table by clicking on the scan.

Fig : Scanner Module

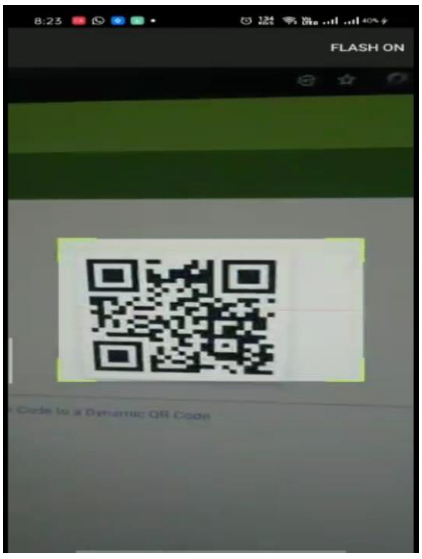


Fig : Scanning of QR Code

7.7 Chatting with Restaurant Bot:- Once the customer scans the code the bot will welcome the customer. Now Bot and customer can chat with each other in realtime. This conversation will be managed by our trained google Dialogflow. Dialogflow will filter out information in it and will send it to google firebase. Google firebase will send that to the restaurant website. All this will done in fraction of a minute with the help of realtime technologies.

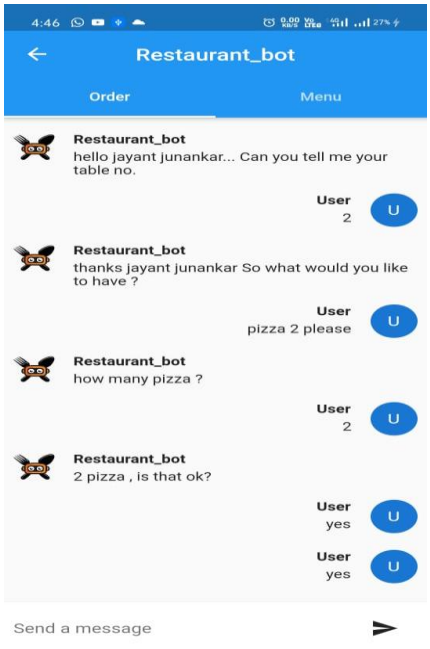
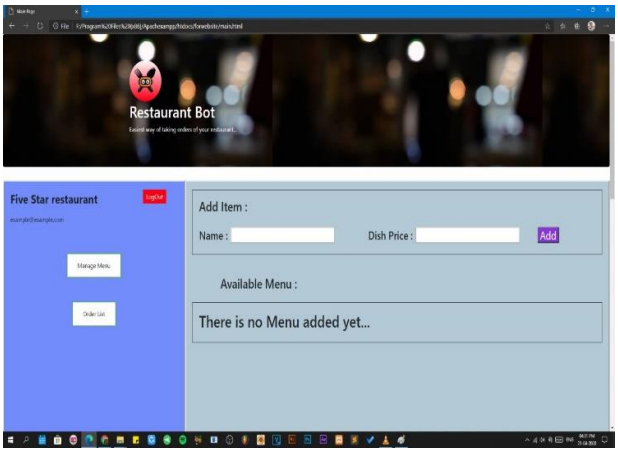
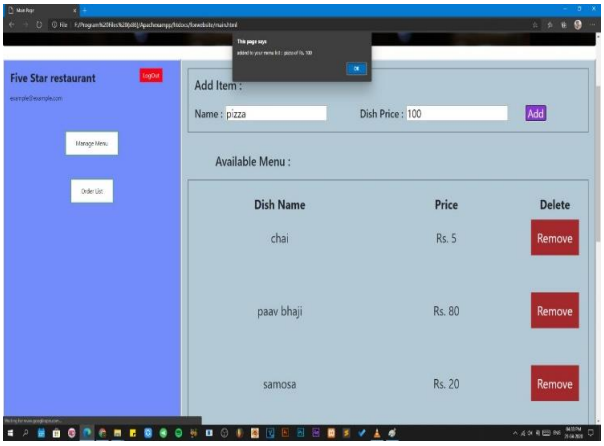


Fig : Chatting between customer and Restaurant

7.8 Manage Menu:- The Real time Menu list is created by the Restaurant admin by clicking on the manage menu in the website. The admin will add name as well as its prize.





7.9 Menu List:- The user will able to place orders from real time menu list updated by restaurant admin. User can access menu list anytime by switching tab to Menu list. The menu list will pop up all the available menus as-

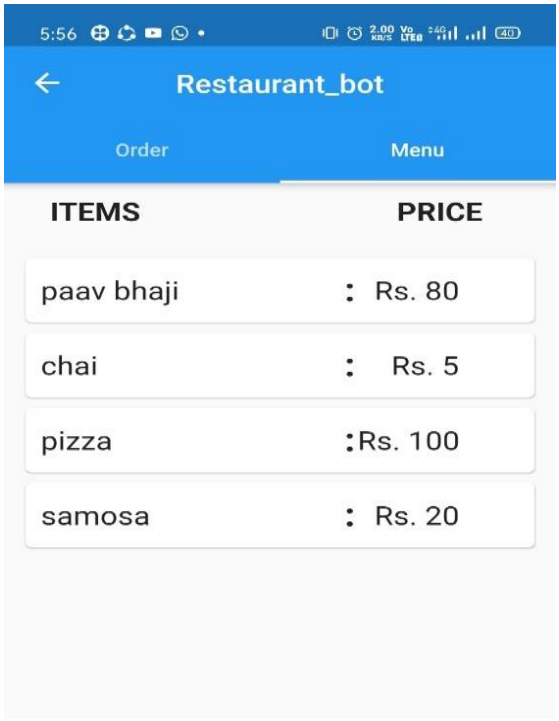


Fig : Menu list in Restaurant Bot App

7.10 Order Placement & Confirmation:- The user will place the order using bot and also confirm it right then and there.

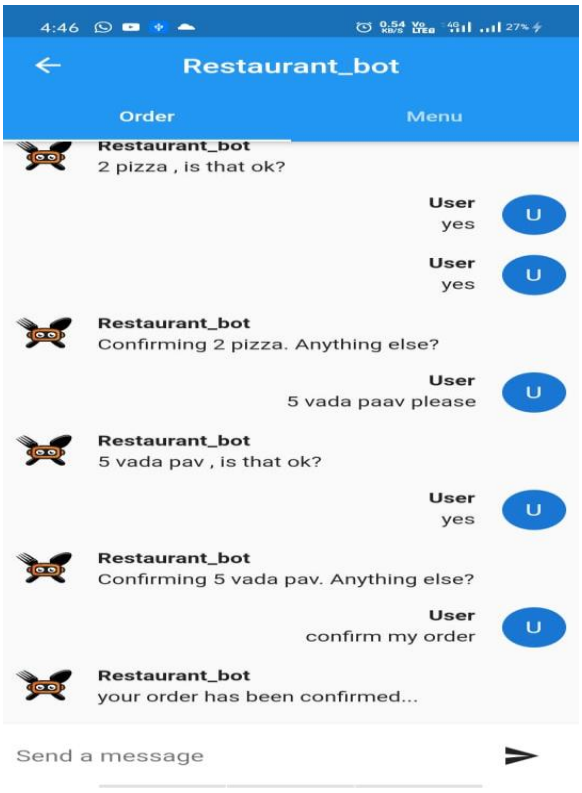


Fig : Order Confirmation

7.11 Real time order list:- Once the order is confirmed, the restaurant admin will able to see it in real time along with order details.. After that he can proceed for placing the orders.

Fig : Confirmed orders on website

CHAPTER 8

TESTING

6.1 Introduction of Testing

Testing is a process of executing a program with the intent of finding an error. A good test case is one that has a high probability of finding an as-yet-undiscovered error. A successful test is one that uncovers an as-yet-undiscovered error. System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently as expected before live operation commences. It verifies that the whole set of programs hang together. System testing requires a test consists of several key activities and step for run program, string, system and is important in adopting a successful new system. This is the last chance to detect and correct errors before the system is installed for user acceptance testing.

The software testing process commences once the program is created and the documentation and related date structures are designed. Software testing is essential for correcting errors. Software testing is the critical element of software quality assurance and represents the ultimate the review of specification design and coding. Testing is the process of executing the program with the intent of finding the error. A good test case design is one that as a probability of finding a yet undiscovered error. A successful test is one that uncovers a yet undiscovered error. Any engineering product can be tested in one of the two ways.

6.2 Levels of Testing

6.2.1 Testing

A process of executing a program with the explicit intention of finding error that is making the power fails.

6.2.2 Software Testing

It is the process of testing the functionality and correctness of software by running it. Process of executing a program with the intent of finding an error.

A good test case is one that has a high probability of finding an as yet undiscovered error. A successful test is one that uncovers an as yet undiscovered error. Software Testing is usually performed for one of two reasons:

- Defect detection
- Reliability estimation

6.2.3 Black Box Testing

Applies to software system or module, tests functionality in terms of inputs and outputs at interfaces. Test reveals if the software function is fully operational with reference to requirements specification.

6.2.4 White Box Testing

Knowing the internal working i.e., to test if all internal operations are performed according to perform structures and data structures. To test if all internal components have been adequately exercised.

6.3 Strategies Of Software Testing

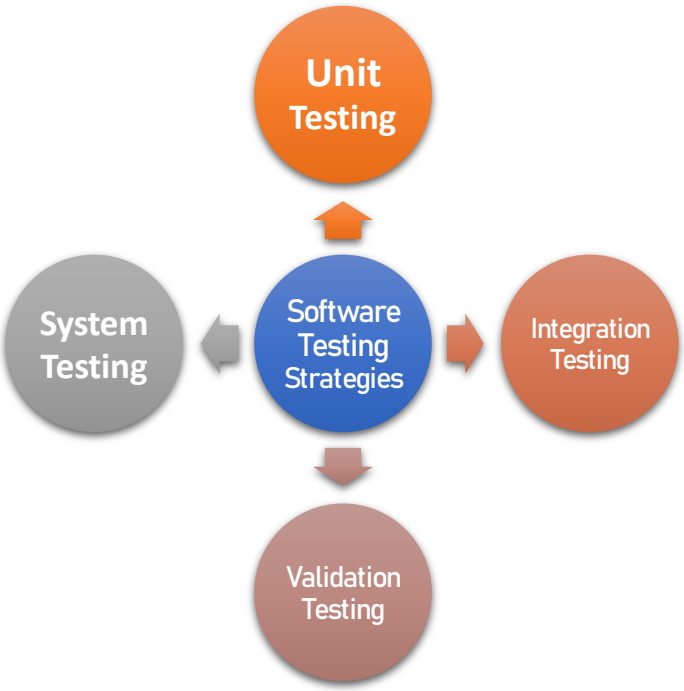


Fig 6.3 Strategies of Software Testing

6.3.1 Unit Testing

6.3.2 Integration Testing

6.3.3 Validation Testing

6.3.3 System Testing

6.3.1 Unit Testing

Unit testing is a type of software testing where individual units or components of a software are tested. The purpose is to validate that each unit of the software code performs as expected. Unit Testing is done during the development (coding phase) of an application by the developers. Unit Tests isolate a section of code and verify its correctness. A unit may be an individual function, method, procedure, module, or object. Sometimes software developers attempt to save time by doing minimal unit testing. This is a myth because skipping on unit testing leads to higher Defect fixing costs during System Testing, Integration Testing and even Beta Testing after the application is completed. Proper unit testing done during the development stage saves both time and money in the end. Unit Testing is of two types: Manual & Automated.

6.3.2 Integration Testing

Integration testing is defined as a type of testing where software modules are integrated logically and tested as a group. A typical software project consists of multiple software modules, coded by different programmers. The purpose of this level of testing is to expose defects in the interaction between these software modules when they are integrated. Integration Testing focuses on checking data communication amongst these modules. Hence it is also termed as 'I & T' (Integration and Testing), 'String Testing' and sometimes 'Thread Testing'. Integration Test Case differs from other test cases in the sense it focuses mainly on the interfaces & flow of data/information between the modules. Here priority is to be given for the integrating links rather than the unit functions which are already tested.

6.3.3 Validation Testing

Validation Testing is carried out to determine if the system complies with the requirements and performs functions for which it is intended and meets the organization’s goals and user needs. This kind of testing is very important, as well as verification testing. Validation is done at the end of the development process and takes place after verification is completed.

Thus, to ensure customer satisfaction, developers apply validation testing. Its goal is to validate and be confident about the product or system and that it fulfils the requirements given by the customer. The acceptance of the software from the end customer is also its part. When software is tested, the motive is to check the quality regarding the found defects and bugs. When defects and bugs are detected, developers fix them. After that, the software is checked again to make sure no bugs are left. In that way, the software product's quality scales up.

The aim of software testing is to measure the quality of software in terms of a number of defects found in it, the number of tests run and the system covered by the tests. When bugs or defects are found with the help of testing, the bugs are logged and the development team fixes them. Once the bugs are fixed, testing is carried out again to ensure that they are indeed fixed and no new defects have been introduced in the software. With the entire cycle, the quality of the software increases.

6.3.4 System Testing

System testing is a level of testing that validates the complete and fully integrated software product. The purpose of a system test is to evaluate the end-to-end system specifications. Usually, the software is only one element of a larger computer-based system. Ultimately, the software is interfaced with other software/hardware systems. System Testing is actually a series of different tests whose sole purpose is to exercise the full computer-based system. Two types of Software Testing are there: White Box Testing & Black Box Testing.

Black Box Testing is a software testing method in which the internal structure/ design/ implementation of the item being tested is not known to the tester.

White Box Testing is a software testing method in which the internal structure/ design/ implementation of the item being tested is known to the tester.

6.4 Test Cases

6.4.1 Application Testing:

- Are customers getting OTP within 30sec of registration?
- Are customers able to verify themselves?
- Are already registered customers able to log in themselves?
- Are customers getting scanning result properly?

- Are customers able to select the menus according to their needs?
- Is chatbot identifying customer needs?
- Are placed orders getting to the customers from the hotel?
- Is there proper synchronization between chatbot and the website?

6.4.2 Website Testing:

- Is admin able to register the restaurant and fill details properly?
- Are admins able to log in?
- Are admins getting real time entries from the customers?
- Are admins able to add new dishes in real time according to availability?
- Are admins able to delete dishes in real time according to availability?

CHAPTER 9

LIMITATIONS & ADVANTAGES

7.1 Limitations

- It requires 3G/4G internet connection.
- Chat Bot will be particularly used for bookings and interactions only.
- Waiter for serving food will be necessarily needed although it's a digitalized system.

7.2 Advantages

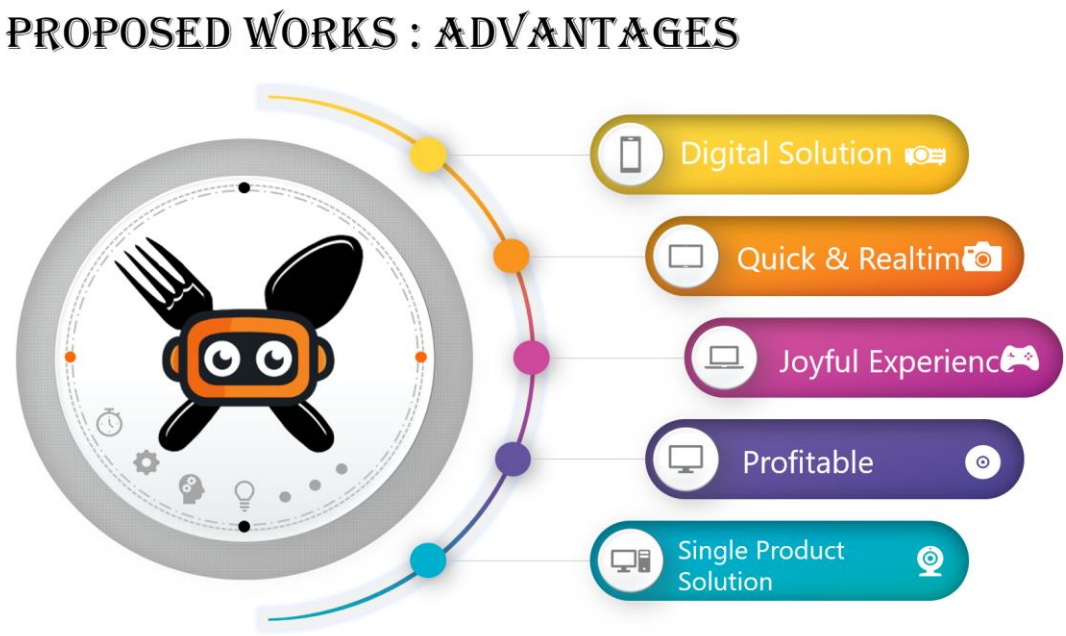


Fig 7.2 Advantages of our Project

- Our Application will work on both Android and iOS Operating System
- Manual Work required is comparatively less.
- The service becomes fast due to it's digital nature.
- Customers can get attended in a more quick and efficient manner.
- No possible human interference.
- Any restaurant can implement this application and website for providing service.

CHAPTER 10

CONCLUSION

Restaurant_Bot will reduce the human efforts, enhance paper less work and save time. There is chance of error while serving foods to the customers. By using proposed system there will no chance of errors and we can easily book table from home or from wherever you want. Customer will order the dishes using the bot according to customer requirement.. The developed system will be very useful in saving time of customer .The Customer can get attended in more quick and efficient manner. The service becomes fast due to its digital nature. Restaurant bot is a chatbot that will be used for fastly interacting with the customers. Also, it is much more user-friendly than other apps. Customer will experience in better way and efficiently. This system will also increases attraction of place for large range of customers. Implementing this system gives a cost-efficient opportunity to give your customers a personalized service experience where they are in control choosing what they want, when they want it – from dining to ordering to payment and feedback..

CHAPTER 11

FUTURE SCOPE

The system can be further enhanced and several other functionalities can be added.

- The future Chatbot will improve like location based services, pre-booking, preordering, & home delivery etc.
- The future Chatbot will not be just a Customer Support agent, it will be an advance assistant for both the business and consumer.
- Also, Human doesn't like storing up contents (mugging up) in their mind. And today with the Internet they can leverage that part. So tasks which require storing the information (data) can be transferred to AI Chatbot. The exciting potential for bots, especially for brands, is the opportunity to talk to their audiences at scale.
- The future, 'always on', global, and digitally native customer demands more from brand interactions and created the expectation of real time messaging contact. Personalized communication has evolved from SMS notifications and segmented social media advertisements, to one-on-one communication through conversational experiences like chat bots.
- Think of messaging and conversational commerce as an extension of a content marketing strategy—making conversations the content, and converting conservations into customers. Scaling personalization means greater engagement with a brand's target demographic and developing relationships that promote brand trust.
- Chat bots can be used simply for brand engagement, or have a functional service like facilitating bookings, ecommerce, delivering targeted information, or customer service.

There are so many opportunities and endless implementations for bots in the future.

CHAPTER 12

REFERENCES

1. Bai Li, Nanyi Jiang, Joey Sham, Henry Shi and Hussein Fazal “Real World Conversational AI for Hotel Bookings”

2. G. Mesnil, Y. Dauphin, K. Yao, Y. Bengio, L. Deng, D. Hakkani-Tur, X. He, L. Heck, G. Tur, D. Yu et al., “Using recurrent neural networks for slot filling in spoken language understanding,” IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 23, no. 3, pp. 530–539, 2015.

3. M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, “Deep contextualized word representations,” in Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), vol. 1, 2018, pp. 2227–2237.

4. E. F. T. K. Sang and F. De Meulder, “Introduction to the CoNLL- 2003 shared task: Language-independent named entity recognition,” in Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, 2003.

5. J. Pennington, R. Socher, and C. Manning, “GloVe: Global vectors for word representation,” in Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, pp.1532–1543.

6. A. Severyn and A. Moschitti, “Learning to rank short text pairs with convolutional deep neural networks,” in Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval. ACM, 2015, pp. 373–382.

7. Experiments with 1,000,000 generated contexts and 100,000 context groups Experiments with 1,000,000 generated contexts and 1,000,000 context groups paper had introduced a method called B-Point Tree to speed up the search process by adding an additional data structure that contains shortcut pointers to the traditional search BST. The experiments had been conducted on a FloristBot, a chatbot that behaves as human personnel in a flower shop. The FloristBot is used to entertain customers and take orders. The results of the experiments indicate that the B-Point Tree manages to outperform the traditional BST algorithm. This suggests that the use of shortcut

pointers may reduce the searching processing time, consequently may increase the efficiency of the chatbot.

8. Jusoh, S., “A study on nlp applications and ambiguity problems.” Journal of Theoretical & Applied Information Technology, vol. 96, no. 6, 2018.

9. “Conversational interfaces powerful speech technology from chinas leading internet company makes it much easier to use a smartphone,” https://www.technologyreview.com/s/600766/10-breakthrough- technologies-2016- conversational-interfaces/ accessed: 2018-01-05.

10. “Will be the year of conversational com-merce,” https://medium.com/chris-messina/2016-will- be-the- year-of-conversational-commerce-1586e85e3991 accessed: 2018-01-10.

11. “Tma associates specialized digital assistants and bots,” http://www.tmaa.com/specializeddigitalassistantsandbots.html accessed: 2017-12-25.

12. “Facebook hopes chatbots can solve app overload,” http://www.wsj.com/articles/facebook-hopes-chatbots-can-solve-app-overload-1460930220 accessed: 2018-03-20.

13. Gregori, E., “Evaluation of modern tools for an omscs advisor chatbot,” 2017.

14. Kandampully, J. and Suhartanto, D., “Customer loyalty in the hotel industry: the role of customer satisfaction and image,” International journal of contemporary hospitality management, vol. 12, no. 6, pp. 346–351, 2000.

15. Hallowell, R., “The relationships of customer satisfaction, customer loyalty, and profitability: an empirical study,” International journal of service industry management, vol. 7, no. 4, pp. 27–42, 1996.

16. Lee, J., Lee, J., and Feick, L., “The impact of switching costs on the customer satisfaction-loyalty link: mobile phone service in france,” Journal of services marketing, vol. 15, no. 1, pp. 35–48, 2001.

17. M¨agi, A. W., “Share of wallet in retailing: the effects of customer satisfaction, loyalty cards and shopper characteristics,” Journal of Retailing, vol. 79, no. 2, pp. 97–106, 2003.

18. Dimitriades, Z. S., “Customer satisfaction, loyalty and commitment in service organizations: Some evidence from greece,” Management Research News, vol. 29, no. 12, pp. 782–800, 2006.