

**WEB BASED E-COMMERCE SYSTEM INTEGRATED WITH CHATBOT**

*A project report submitted in partial fulfilment of the requirement for the award of degree of*

##### BACHELOR OF TECHNOLOGY

*In*

##### INFORMATION TECHNOLOGY

*Submitted by*

|  |  |
| --- | --- |
| **M. SHYAM MANIKANTA** | **18341A1227** |
| **J. RUSHI** | **18341A1218** |
| **A. LALITHA** | **18341A1202** |
| **B. SHRAVAN KUMAR GOUD** | **18341A1206** |
| **V. SURESH** | **18341A1247** |

*Under the esteemed guidance of*

###### Mrs. T. DANIYA

Assistant Professor, Dept. of Information Technology

GMR Institute of Technology

**An Autonomous Institute Affiliated to JNTUK, Kakinada**

(Accredited by NBA, NAAC with ‘A’ Grade & ISO 9001:2008 Certified Institution)

###### GMR Nagar, Rajam – 532 127, Andhra Pradesh, India

**MAY 2022**

#### Department of Information Technology

**CERTIFICATE**

This is to certify that the thesis entitled **WEB BASED E-COMMERCE SYSTEM INTEGRATED WITH CHATBOT** submitted by **M. SHYAM MANIKANTA (18341A1227)**, **J. RUSHI (18341A1218), A. LALITHA (18341A1202), B. SHRAVAN**

**KUMAR GOUD (18341A1206), V. SURESH (18341A1247)** has been carried out in partial fulfilment of the requirement for the award of degree of **Bachelor of Technology** in **Information Technology** of **GMRIT, Rajam** affiliated to **JNTUK, KAKINADA** is a record of bonafide work carried out by them under my guidance & supervision. The results embodied in this report have not been submitted to any other University or Institute for the award of any degree.

###### Signature of Supervisor Signature of HOD

**Mrs. T. Daniya Dr.Ajit Kumar Rout**

Assistant Professor Professor & Head

Department of IT Department of IT

GMRIT, Rajam. GMRIT, Rajam.

The report is submitted for the viva-voce examination held on ………………..

##### ACKNOWLEDGEMENT

It gives us an immense pleasure to express deep sense of gratitude to our guide **Mrs. T. Daniya,** Assistant Professor, Department of Information Technology of whole hearted and invaluable guidance throughout the project work. Without his sustained and sincere effort, this project work would not have taken this shape. He encouraged and helped us to overcome various difficulties that we have faced at various stages of our project work.

We would like to sincerely thank our Head of the department **Dr. Ajit Kumar Rout**, for providing all the necessary facilities that led to the successful completion of our project work.

We would like to take this opportunity to thank our beloved Principal **Dr.C.L.V.R.S.V.Prasad**, for providing all the necessary facilities and a great support to us in completing the project work.

We would like to thank the project coordinator **Mrs. T. Daniya**, all the faculty members and the non-teaching staff of the Department of Information Technology for their direct or indirect support for helping us in completion of this project work.

Finally, we would like to thank all of our friends and family members for their continuous help and encouragement.

###### M. SHYAM MANIKANTA (18341A1227) J. RUSHI (18341A1218)

**A. LALITHA (18341A1202)**

###### B. SHRAVAN KUMAR GOUD (18341A1206) V. SURESH (18341A1247)

iii

##### ABSTRACT

E-commerce has the potential to be a huge success in today's commercial world. E-Commerce platform is used for purchasing of the human required goods and selling of items through it as well as the payments of the purchased goods made through online mode over the different payment platforms over the internet. E-commerce may represent a paradigm change that affects both marketers and customers. Rather, e- commerce is merely a new way to spice up existing business methods. It's causing a complete shift in the traditional company model. A web application, which is a piece of software that runs on a website. It is a computer-based software programme that is saved locally on the system's operating system. It is used by users or administrators over the Internet. It can be used to create an application for a variety of platforms. However, we believe that using Python is the safest option and can be employed in a variety of situations. It also plays a significant role in our web apps. Hackers won't be able to use SQL Injection as rapidly if the web application is written in Python. This project is intended to create a chatbot to be used by customers to get their queries responded easily from the e-commerce website. This chatbot has the capacity to make friendly conversations and respond to the queries. Moreover, it provides information with product details, payment method and many more.

**Keywords:** E-commerce, Python, Chatbot etc.

iv

##### TABLE OF CONTENTS

|  |  |
| --- | --- |
| **ACKNOWLEDGEMENT** | iii |
| **ABSTRACT** | iv |
| **LIST OF FIGURES** | v |
| **LIST OF SYMBOLS & ABBREVIATIONS** | vi |
| **CHAPTER 1 INTRODUCTION** | **1** |
| 1.1 E-COMMERCE SYSTEM | 2 |
| 1.2 WEBSITE PROPRIETORS AND PROVIDERS | 2 |
| 1.3 ONLINE SHOPPING APPLICATION | 3 |
| 1.3.1 HOMEPAGE | 3 |
| 1.3.2 CLOTHING PAGE(PRODUCTS) | 3 |
| 1.3.3 CONTACT US PAGE | 3 |
| 1.3.4 REGISTER PAGE | 3 |
| 1.3.5 LOGIN PAGE | 3 |
| 1.4 WEB APPLICATION DEVELOPMENT | 4 |
| 1.4.1 BACKGROUND | 5 |
| 1.5 WEB DEVELOPMENT STRUCTURE | 5 |
| 1.5.1 FRONT END DEVELOPMENT | 5 |
| 1.5.2 BACKEND DEVELOPMENT | 6 |
| 1.5.3 DATABASE DESIGN | 7 |
| 1.6 INTEGRATING WEBSITE AND THE DATABASE | 7 |
| 1.7 DEVELOPMENT TOOLS | 8 |
| 1.7.1 HTML | 8 |
| 1.7.2 CSS3 | 9 |
| 1.7.3 JAVASCRIPT | 10 |
| 1.7.4 BOOTSTRAP | 11 |
| 1.7.5 PYTHON | 11 |

[1.7.6 FLASK 11](#_TOC_250047)

* 1. [CHATBOT 12](#_TOC_250046)
  2. [PAYMENT SYSTEM 13](#_TOC_250045)

[CHAPTER 2 LITERATURE SURVEY 16](#_TOC_250044)

[2.1 STUDY OF RELATED PAPERS 18](#_TOC_250043)

CHAPTER 3 REQUIREMENT SPECIFICATION 28

* 1. [FUNCTIONAL REQUIREMENTS 29](#_TOC_250042)
  2. [NON-FUNCTIONAL REQUIREMENTS 29](#_TOC_250041)
  3. [SOFTWARE REQUIREMENTS 30](#_TOC_250040)
     1. [FRONTEND FRAMEWORKS 30](#_TOC_250039)
     2. [BACKEND FRAMEWORKS 30](#_TOC_250038)
  4. [HARDWARE REQUIREMENTS 31](#_TOC_250037)
     1. [WINDOWS VERSION 31](#_TOC_250036)

[CHAPTER 4 METHODOLOGY 32](#_TOC_250035)

* 1. [IMPLEMENTATION OF BASIC PAGES 33](#_TOC_250034)
     1. [CREATION OF TEMPLATE BY USING FLASK 33](#_TOC_250033)
     2. [TEMPLATE INHERITANCE 33](#_TOC_250032)
     3. [FILE ORGANIZATION 33](#_TOC_250031)
     4. [PYTHON SUPPORT FOR WEB DEVELOPMENT 33](#_TOC_250030)
  2. [DEFINING USE CASE MODELS 34](#_TOC_250029)
     1. [DOMAIN MODELING 35](#_TOC_250028)
     2. [ARCHITECTURAL PATTERN 35](#_TOC_250027)
     3. [WEB APPLICATION WORKING 36](#_TOC_250026)
  3. [MODULAR DESIGN 36](#_TOC_250025)
  4. [PROPOSED ARCHITECTURE FOR COGNITIVE BOT 37](#_TOC_250024)
  5. [MODELS USED IN CHATBOT 38](#_TOC_250023)
  6. [PERSONALIZED AND INTERACTIVE DOMAIN 38](#_TOC_250022)
  7. [METHODS 39](#_TOC_250021)
     1. [RETRIEVAL BASED METHOD 39](#_TOC_250020)
     2. [GENERATION BASED METHOD 39](#_TOC_250019)
  8. [CHIT-CHAT CONVERSATION MODELING 39](#_TOC_250018)
  9. [META ENGINE 40](#_TOC_250017)

[CHAPTER 5 SYSTEM DESIGN 41](#_TOC_250016)

* 1. [UML DIAGRAMS 42](#_TOC_250015)
     1. [ACTIVITY DIAGRAM 42](#_TOC_250014)
     2. [CLASS DIAGRAM 43](#_TOC_250013)
     3. [DEPLOYMENT DIAGRAM 43](#_TOC_250012)
     4. [COMPONENT DIAGRAM 44](#_TOC_250011)
     5. [USECASE DIAGRAM 44](#_TOC_250010)
     6. [STATE CHART DIAGRAM 45](#_TOC_250009)

[CHAPTER 6 RESULTS AND DISCUSSIONS 46](#_TOC_250008)

* 1. [RISE IN USE OF E-COMMERCE WEBSITES 47](#_TOC_250007)
  2. [ADVANTAGES OF PYTHON FOR BACKEND 48](#_TOC_250006)
  3. [ONLINE SHOPPING 50](#_TOC_250005)
  4. [CHATBOT DEVELOPED BY XATKIT 50](#_TOC_250004)
  5. [SUPERAGENT CHATBOT 52](#_TOC_250003)
  6. [PAYMENT GATEWAY 53](#_TOC_250002)
  7. [PROPOSED SYSTEM 54](#_TOC_250001)

CHAPTER 7 CONCLUSIONS AND FUTURE SCOPE 57

[CHAPTER 8 REFERENCES 59](#_TOC_250000)

**APPENDIX A APPENDIX B**

##### LIST OF FIGURES

###### FIGURE NO TITLE PAGE NO

4.2 LOGIN MODULE ARCHITECTURE 10

* 1. ACTIVITY DIAGRAM 42
  2. CLASS DIAGRAM 43
  3. DEPLOYMENT DIAGRAM 43
  4. COMPONENT DIAGRAM 44
  5. USECASE DIAGRAM 44
  6. STATE CHART DIAGRAM 45
  7. SCREENSHOT OF HOME PAGE 54
  8. SCREENSHOT OF LOGIN PAGE 55
  9. SCREENSHOT OF PAYMENT PAGE 55

v

##### LIST OF ABBREVIATIONS

CV : Computer Vision

OOP : Object Oriented Programming

MVC : Model View Controller

PHP : Hypertext Preprocessor

HTML : Hypertext Markup Language

CSS : Cascading Style Sheets

DBMS : Database Management System

GNU : General Public License

SQL : Structured Query Language

KB : Knowledge Base

QA : Question Answering

FAQs : Frequently Asked Questions

MDE : Model Driven Engineering

DSLs : Domain Specific Languages

RST : Rhetorical Structure Theory

seq2seq : Sequence to Sequence

PRF : Polished Response Filter

GP : Generation Based Polisher

PS : Prototype Selector

vi

#### Chapter 1 INTRODUCTION

##### INTRODUCTION

###### E-commerce system

E-commerce is fast gaining ground as an accepted and used business paradigm. More and more business houses are implementing web sites providing functionality for performing commercial transactions over the web. It is reasonable to say that the process of shopping on the web is becoming commonplace. The objective of this project is to develop a general purpose e-commerce store where product like clothes can be bought from the comfort of home through the Internet. However, for implementation purposes, this paper will deal with an online shopping for clothes. An online store is a virtual store on the Internet where customers can browse the catalog and select products of interest. The selected items may be collected in a shopping cart. At checkout time, the items in the shopping cart will be presented as an order. At that time, more information will be needed to complete the transaction. Usually, the customer will be asked to fill or select a billing address, a shipping address, a shipping option, and payment information such as credit card number. An e-mail notification is sent to the customer as soon as the order is placed.

Now a days the new COVID-19 outbreak has radically altered the existing business situation. Many people are feeling paralyzed by fear. Unfortunately, the future of business appears to be bleak for some. For the time being, there are still feasible options of generating cash for stores that have closed their doors! Having a solid E-Commerce platform can help consumer and also the business owner to generate consistent revenue and save their store’s or company.

###### Website Proprietors and Providers

Web based moving can likewise widen your client base, as you can pitch to individuals who are interstate or abroad. It enables you to work without fundamentally requiring a conventional shop front and gives your clients greater adaptability since they can purchase your items whenever of the day. Internet shopping doesn't speak to

everybody, so ensure your objective market is OK with purchasing on the web before you begin. You may discover you can achieve more clients by offering web based shopping close by a customary shop. Web based shopping makes utilization of advanced innovation for dealing with the stream of data, items, and instalment between shopper, website proprietors and providers. In the event that you were move your item disconnected, at that point you will ready to move your items in constrained region where your shop is found, however when you are going to move your items online then you can move the items in various locales. For this situation, your benefit will be naturally expanded. Presently multi day everybody is occupied with their work they didn't get time for buying the items by going to specific.

###### Online Shopping Application

Anyone can view Online Shopping portal and available products, but every user must login by his/her Username and password in order to purchase or order products. Unregistered members can register by navigating to registration page. Only Admin will have access to modify roles, by default developer can only be an ‘Admin’. Once user register site, his default role will be ‘User’.

* + 1. **Homepage:** The Home Screen will consist of screen were one can browse through the products which we have on our website.
    2. **Clothing page (products):** This page consists of product details. This page appears same for both visitors and users.
    3. **Contact Us Page:** Visitors and Registered users can contact website owners or administrators from Given Bottom of The portal.
    4. **Register page**: New users can register or Signup for New Account After Successfully Register user can Access The portal by Login in to website.
    5. **Login page**: Login page for both users and administrators have different Authentication’s

###### Web Application Development

From the inception of modern programming languages in the early 50s of the 20th century, numerous programming languages have been invented. A few have been

discontinued along the road, and a few have survived with appreciation from computer scientists, programmers, and engineers. Of those survived, languages such as C, C++, Java, Python, JavaScript, and Ruby have been the go-to programming languages to build small to enterprises applications for desktop, mobile and other handheld devices.

While building a software application, one of the most important tasks of a programmer is to make the code readable, understandable and reusable. Often termed as DRY methodology, adopting this methodology not only reduces the boilerplate code in the code-base but also organizes the same logic in one place and makes it reusable in other parts of the codebase for the same purpose.

Another difficulty for the programmer is to understand the code which is written by other fellow programmers. If the same logic is repeated and found in many places, following the small piece of code could be a very demanding task as the source of truth for the same can be found in many places in the codebase resulting in multiple interpretations increasing the complexity. Using the software frameworks, boilerplate, duplicate and un-standardized code can be reduced.

A software framework is a set of standardized libraries and tools for a programming language. It helps to build a concise software application efficiently. Because of its code reusability and efficiency, it allows a programmer to bootstrap an application in no time and start implementing the business logic. Software frameworks come in different types for different programming languages. In this thesis, however, we will try to compare and uncover the various aspects of the two most popular web application frameworks of Python programming language, i.e. Flask and Django. According to the documentation, Flask is as a micro framework. Because of it being lightweight and having the features of flexibility and extensibility, it can be bootstrapped in no time. On the other hand, Django is known as a “battery included” framework, which claims to have most of the required extensions and libraries to boot- strap a generic application providing a developer more time on implementing the business logic.

###### Background

Web application development is a process of developing software applications that can run on websites. Even though web application development follows the software development process, the technology and the architecture used for it is quite different. The software application that runs on a personal computer (PC) might not depend on the internet in contrast to the web application that depends upon the remote servers.

The web applications and the client-server technology have come quite far in comparison to the simple standalone phone book app made by Tim Berners-Lee in 1989. Now a-days, web application comes in different shapes and sizes such as static, dynamic, con-tent management system, e-commerce, and gaming to live content sharing portals. Commonly shared technology of applications mentioned above types is the backend and frontend technology.

###### Web Development Structure

The entire development process has been subdivided into two: the front end development and the backend development. The front end comprises of the visually visible parts such as the home page, contact page, admin panel, shopping cart page. The back end contains the database and its interaction with the front-end.

###### Front End Development

The front end was initially raw coded using JavaScript. JavaScript is a client side scripting language which is a dedicated language for web development. JavaScript code was simply mixed with the Hypertext Mark-up Language (HTML5) code. A static page is an HTML5 document that is stored on the web server and does not change. Hypertext mark-up language is the language used to design the web pages of an application. Cascading Style Sheet (CSS) is a style sheet language used for describing the look and formatting a document written in a mark-up language. These CSS files are linked with the class files with php extensions to put the panels in order, the text with correct font, size and color. JavaScript is a client side scripting language most commonly used as

part of web browsers and its implementations allow client side scripts to interact with the user, control the browser and alter the document content which is displayed. For example, in website for the client’s registration, the system asks to provide their details which contains their name, email address, mobile number, etc. If they missed any of the details, then immediately the browser asks them to fill the particular field. This is implemented & handled by a JavaScript.

On the other hand, frontend development is mostly concerned with the aesthetic and the content displaying part, which is also known as client-side development. One of its difficult challenges is to be able to show the material in the different types of devices and browser. Many devices have its own Software Development Kit (SDK) which should be followed to serve the same content that is served in the browser. Website design, usability and user-friendliness are the essential factors that are addressed during the development. The figure below is an example of how the frontend side of an application works.

###### Backend Development

Backend development deals with the logical side of the web application. It mainly concerns the programming languages, core architecture and the logics. Those logics are mainly written in programming languages that can run on computer servers. It also influences how the data is stored, accessed and served from the servers. the backend part of the application, which consists of server-side scripts, frameworks, database and APIs.

The Database Management System (DBMS) provides support for the back end. The database management system is essentially software where admin can create the database, add, drop, alter and update tables. The tables can hold different types of data for example: integer, variable characters etc. in our application we have chosen the MySQL DBMS to hold the database. MySQL is a relational database management system. The main reason is MySQL development project has made its source code available under the terms of the General Public License (GNU) which is an open source web application.

###### Database Design

The information provided by the customer while registering in the website is stored in the database. The products with their identification, description and image is stored in the database. Moreover, if admin update any of the featured products then update takes place in the database. So the program has a lot to do with the database. Any query is run on the database by Structured Query Language (SQL). As stated earlier that PHP has some useful features one of them is the support to connect the database and run queries. The following diagram explains the details the database design. The system has six tables in the database namely: advertise, brand, category, product, small add and registration. The formation of three tables are shown in the diagram. The table entitled “product” has the attributes namely: product\_id, product\_rate, category\_id, brand\_id, and product\_rate. Where product\_id is the primary key for this table i.e. each product will have a unique identity.

The attribute category\_id plays as a foreign key for this table *i.e.* this foreign key creates a link with table category having two attributes: website his information will be stored in the customer table with the attributes- name, customer\_id, product\_id, quantity and email. Again customer\_id is the unique key for this table and product\_id is the foreign key to link with the product table.

###### Integrating Website and the Database

Admin should be able to keep track of the all products ordered, payment details, customer inquiries. So a well-organized database is very essential for maintenance of an ecommerce web-application. Customers ordering the products from an e-commerce website should be able to get the information of products, ask questions, review, and give feedback on the products. Also, they can select the products they would like to buy, submit payments. The user must be able to access the database and for this the remote database connectivity is established. The user information is stored in and retrieved from the remote database. The database used in this application is PostgreSQL Server.

The administrator includes new items, settles on the ideas on certain items. The items that are not sought after can be for all time erased from the database which implies that the closeout of those items on the site is ceased. The administrator is given the benefits to change and refresh the database. The seller subtleties, installments to the merchant are to be taken consideration by the administrator. The administrator can choose the money down office to make accessible in all territories or limit to certain regions as it were. The transportation subtleties, for example, the regions where the delivery offices are accessible, the subtleties of the delivery organizations and the exchange between them are to be checked. The reports of the regular deal, benefit, the items are very on interest, etc are produced according to the demand. The report age isn't feasible for the client. A client suspected to have broken the protection strategy can be hindered by the administrator. The customer accesses the website by typing the URL corresponding to the website. The user can request for product details by clicking the products menu. The user is provided with two types of Add to Cart – onsite add to cart and offsite add to cart. In the add to cart the user is allowed to add any product available on the site to the list. E-commerce website is made by Scripting languages like HTML,CSS3,JavaScript and Bootstrap .This markup language make the website more attractive and useful and user-friendly to use and shopping .Markup languages help in making the things more attractive and imaginary. We used Front-end Technologies like HTML, CSS, JAVASCRIPT, BOOTSTRAP. And for Back-End we used Python (Flask). For Database we used PostgreSQL.

###### Development Tools

* + 1. **HTML**

HTML is HyperText markup language. It is an emerging technology, cascading style sheets, could eliminate many of the HTML table could be used to control the layout of a webpage. A web designer might separate the header, body text, and sidebar of a webpage by putting each into a distinct cell. Additionally, the net designer could put each link button on the header and sidebar into a separate cell so he or she could define unique properties for every button. Then, within the body of the page, the net

designer could separate the textual and graphical elements into different cells to regulate spacing and other attributes individually.

HTML has wide ranges of support for the contents from text to spreadsheets, video clips, animations pictures. The web page often called as a document in a technical term, the document type defined at the beginning of the page determines what type of document it is and how it should be taken by the web browser while rendering it. Being the essential part of the world wide web, it has gone through the various phases of development from 1991 till date, while HTML being the initial version and HTML5 being the latest one respectively.

###### CSS3

CSS may be a formatting language want to add styling to your page. This can be done by having the CSS document linked into your html page. This page then has selectors and properties which affect the tags inside your html document.CSS was introduced in 1996. It had been created to prevent people from having to repeat plenty of code. For instance, if someone wanted to alter the paragraph text, they'd should have intercourse every single time they wanted to alter the properties. CSS has since become more adapted to having more features, for instance we will now use the tools and alter the background to an enormous array of colors.

W3C (Worldwide Web Consortium), CSS (Cascading Style Sheets) defines CSS as the language describing the presentation of Web pages, colours, fonts and the layouts.CSS allows one to stylize the HTML document according to their need. It contains a set of style rules with which the web pages could be rendered. Not only to that, but it also helps render the page responsively on the devices of different shapes and sizes.

There are two style formatting rules for the CSS. One is called inline styling, and another one is external styling. In the external styling, however, the styling rule sits in a different file. This could then be linked by a unique tag in the HTML document. The cascading part of the CSS refers to how the rules are applied to the HTML elements. The HTML document is styled hierarchically. Therefore, it is the

responsibility of CSS to find the precedence of the style rules, accordingly, ultimately establishing a cascading effect.

###### JavaScript

JavaScript was initially developed to replace the Java Applets which were used in the web browser. In 1996, one of the dominant browsers creating company, Netscape, submitted JavaScript to the Ecma International to influence its presence in all of the browser. Even though each browser has its implementation of the JavaScript, the underlying standard for it is the same. Chrome uses v8 JavaScript Engine; Internet Explorer uses Jscript in contrast to the Mozilla, which uses the standard JavaScript.

JavaScript is a powerful client-side scripting language. JavaScript is employed mainly for enhancing the interaction of a user with the net page. In other words, you can make your web content more lively and interactive, with the assistance of JavaScript. JavaScript is additionally being employed widely in game development and Mobile application development.

Fast forward almost twenty years; JavaScript has become one of the most popular languages for frontend technology. A simple script can be added inline within the HTML document or in a separate file with a link tag in the header file. This way the HTML document knows which script to lead from which location. The main impacting feature of JavaScript is the ability to load or refresh the page content without reloading the whole page. This can be achieved by targeting only the concerned tag. Based on the requirement, it also helps on adding the CSS style dynamically. There are many JavaScript libraries available on the internet, which makes a developer’s life easier by leveraging the inner work of JavaScript and help focus on the aesthetic part of it. Some examples are React, Vue.js, jQuery Backbone.js, and Angular.

###### BootStrap

Bootstrap could be a web framework that focuses on simplifying the event of informative sites. The primary purpose of adding it to an internet project is to use

Bootstrap's choices of color, background effect, mobility size, font and layout to it project. As such, the primary factor is whether or not the developers answerable find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements. The result an identical appearance for prose, tables and form elements across web browsers. In addition, developers can cash in of CSS classes defined in Bootstrap to customize the look and component of their tools of their contents. Bootstrap is used for light-and dark-colored tables, more prominent pull quotes, page headings, and text with a highlight.

###### Python

Python is a general purpose, high level programming language that focuses on the code readability. for web development lines of code will be fewer lines in python compared to other languages. Python is able to do this because of its extensive standard libraries, which make Web development code straightforward and simple. Python may be used on a variety of systems (Windows,Mac, Linux, Raspberry Pi, etc). Python has a simple syntax that is similar to that of English. Python may be approached in three ways: procedural, object-oriented, and functional. Python is an interpreter language, which means that code may be run as soon as it is written. As a result, prototyping may be done quickly.

###### Flask

Flask is a micro-framework designed to create a web application in a short time. It only implements the core functionality giving developers the flexibility to add the feature as required during the implementation. It is a lightweight, WSGI application framework. This framework can either be used for pure backend as well as frontend if need be. The former provides the functionality of the interactive debugger, full request object, routing system for endpoints, HTTP utilities for handling entity tags, cache controls, dates, cookies etc. It also provides a threaded WSGI server for local development including the test client for simulating the HTTP requests. Werkzeug and Jinja are the two core libraries The Jinja, however, is another dependency of the Flask. It is a full-featured template engine. Sandboxed execution, powerful XSS prevention,

template inheritance, easy to do debug, configurable syntax is it's few of many features. In addition, the code written in the HTML template is compiled as python code.

Since Flask is often termed as a prototyping framework, it does not include the abstraction layer for the database or any sorts of validation and security whatsoever. Therefore, Flask has given full flexibility to the implementor to add the requirements. There are extensions available for the Flask frameworks. Libraries but not limited to are, gunicorn for server, SQLAlchemy for database, Alembic for database migration management, celery & Redis for an asynchronous task runner, Flask-WTF form for form validation and Flask-limiter for rate-limiting the web requests. Flask is available for Python 3 and the newer version; it is also available in PyPy and easily installable with Python’s official package manager pip.

Flask is a Python API that allows us to create web-based applications. It was developed by Armin Ronacher. Flask's framework is easier to understand than Django's, and it takes less fundamental code to develop a simple web application. A Web-Application Framework, often known as a Web Framework, is a set of modules and libraries that allow programmers to create apps without having to write low-level code such as protocols and thread management. Flask is an open-source web framework.

###### Chatbot

Chatbot communication, an essential task in both natural language processing and artificial intelligence fields, involves a robot communicating with human beings using natural language in open domains. Chatbots play a critical role in many real-world applications, such as smart speakers, customer service systems, and social robots. Research on chatbots began in the 1960s initially, researchers used sets of handwritten rules and templates. However, such rule-based models require significant human effort and lack flexibility. In recent years, as large-scale dialog corpora and high-speed computational resources have become available, the early rule-based models have been rapidly replaced by data-driven models. The existing data-driven models can be

categorized as retrieval-based or generation-based models. Retrieval-based models, return responses from a corpora by computing contextual similarity. The retrieved responses are usually fluent and informative because they are written by humans; however, they may contain irrelevant content. In contrast, generation-based models return responses generated based on language rules learned during the training process.

Thus, the generated responses are relevant but may suffer from problems such as being safe responses, lacking fluency or including grammatical errors. An excellent conversational robot should return replies that are contextually relevant, informative, and fluent. Therefore, researchers have proposed using retrieved results as a basis for response generation and this approach has made progress. However, the combined methods do not actually take advantage of both methods, and their results tend to be similar to retrieved responses. To address this issue, we introduce a polishing process into the generation model that was inspired by writing articles. Human authors typically write an early draft and then polish it in detail. When polishing a specific sentence, authors tend to adopt writing styles and techniques from existing literature. In terms of response generation, the context, that consists of the sentences that immediately precede the response, provides background knowledge to generate a draft response, while retrieved responses provide information about the language style and techniques that can be used to polish the draft response.

###### Payment System

Payment is that the integral process within the mercantile process, electronic payment system is that the integral a part of electronic commerce. Thanks to the emergence of electronic commerce has created new financial needs through which new payment systems are created while traditional payment systems cannot be ready to fulfill its needs. As an example new payment systems are of the forms like auctions between individuals 1 ⁄ 2s online leads to looking for new payment systems meaning peer to look payment methods that enables individuals to create payments through their emails. By recognizing these needs for all interested parties.

Payment gateway is the technology that captures and transfers payment data from the customer to the acquirer and then transfers the payment acceptance or decline back to the customer. A payment gateway validates the customer’s card details securely, ensures the funds are available and eventually enables merchants to get paid. It acts as an interface between a merchant’s website and its acquirer. It encrypts sensitive credit card details, ensuring that information is passed securely from the customer to the acquiring bank, via the merchant. In other words, the payment gateway works as the middleman between your customer and the merchant, ensuring the transaction is carried out securely and promptly. An online payment gateway can simplify how merchants integrate the necessary software. As the middleman during the payment processing, the gateway manages the customer’s sensitive card details between the acquirer and the merchant. The payment gateway is a key component of the electronic payment processing system, as it is the front-end technology responsible for sending customer information to the merchant acquiring bank, where the transaction is then processed.

Payment gateway technologies are always evolving to reflect new consumer tastes and technical capacities. In the past, terminals would accept credit cards using magnetic strips and required paper signatures from the customer. With the development of chip technologies, the signature phase could be removed in favor of a personal identification number(PIN) entered directly into the payment gateway hardware. Today, contactless purchases are also available, with many customers now using their phones as a payment device instead of plastic credit cards. The architecture of a payment gateway will differ depending on whether it is an in-store gateway or an online payment portal.

This paper is structured as follows: Chapter 2 introduces the Literature Survey, Chapter 3 describes about the requirements for developing the project, Chapter 4 gives an overview of Methodologies that are implemented, Chapter 5 shows the system design, Chapter 6 discusses about the Results and Discussions. Finally Chapter 7 shows the Conclusion and Future Scope.

#### Chapter 2 LITERATURE SURVEY

##### LITERATURE SURVEY

###### Study of Related Papers

**Paper [1] Vangala Rama Vyshnavi, Amit Malik, “Efficient Way of Web Development Using Python and Flask”, International Journal of Recent Research Aspects Vol. 6, Issue 2, June 2019.**

In this paper, the author described Python is a general purpose, high level programming language that focuses on the code readability, for web development lines of code will be fewer than other languages. It is possible for Python because of large standard libraries which make the Web development code simple and short. These libraries have pre-coded functions provided by Python community which can be easily downloaded and can be used as per the development needs. Initially Python was designed for web designed for web servers to deal with the incoming traffic on the server. Flask uses Jinja Template Engine and the Werkzeug WSGI ToolKit.

###### Flask

Flask structures categories into two parts “Static files & Template files”, template file have all the Jinja templates including Html pages, whereas static file have all static codes needed for website such as CSS code, JavaScript code and Image files.

###### Technical Advantages

Robust Robustness is the ability of system to cope with errors during execution. Robustness is also used as the ability of an algorithm to continue operating despite abnormalities in input, calculations, etc. robustness can encompass many areas of web development. Open Source: Python & Flask are open source languages in which the source code is available to the general public for use and/or modification from its original design. Open source code is typically a collaborative effort where other developers can improve upon the source code and share the changes within the community so that order members can help improve it further.

Flask is a web framework. This means flask provides you with tools, libraries and technologies that allow you to build a web application. This web application can be web pages, a blog, a wiki or go as a web-based calendar application or a commercial website. Flask is a lightweight web application framework written in Python and baseband on the WSGI toolkit and Jinja2 template engine. Flask is part of the Categories of the micro framework. Micro-framework is normally frame-work with little to no dependencies to external libraries.

###### Features of Flask

* + 1. Integrated supports for Unit Testing
    2. Uses Jinja2 Templating
    3. Supports for secure cookies
    4. Extensive documentation
    5. Google app engine compatibility
    6. Restful request dispatching.

This paper can conclude as python can be more efficient than any other web development languages, because python is fast, broad and better than java, perl, tcl etc. Python Jinja flask are more useful for maintaining complexity technologies. These are also helps for data fetch from [WWW.](http://WWW/) It is more powerful, fast with the help of flask Template Engine. Therefore, python is great for backend web development.

###### Paper [2] S. E. Ullah, T. Alauddin and H. U. Zaman, "Developing an E-commerce website," 2016 International Conference on Microelectronics, Computing and Communications (MicroCom), 2016.

**E-Commerce**

Electronic commerce refers to a wide range of online business activities for products and services. It is usually associated with online buying and selling over the internet or conducting any transaction involving the transfer of ownership or rights to use goods or services through a computer mediated network. In our eyes we see it as a new dimension to the varied use of the internet and our purpose is to make it trendy in our country where its use is particularly very low. Because of the high context culture,

it is very important to develop trust among the people interested in a transaction. E- commerce in Bangladesh actually started in the year of 1999 based in USA with some non-resident Bangladeshis.

###### Features of Website

An end user can perform free online registration. Can search a specific product of his/her interest. Can order online the payment method is currently the “Cash on delivery” method. The administrator possesses the only right to add any product, update its price or delete any product. Can promote small or big advertisements and delete any specific advertise as well.

Customers can update their personal information at any time. After logging in to the system the customers can order whatever they want without giving their billing information again and again. In this website products are organized based on categories and brands. Customer can enjoy the detail view of any product by just panning cursor over the product image in the product details section. The website also inherits automated inventory system. So, whenever a customer buys a product it is automatically being deducted from the inventory system & if any product becomes less than five in quantity then automatically e-mail should be delivered to the admin and supplier. Moreover, if any product becomes out of stock then no customer would be able to buy that product.

E-Commerce has changed our life styles entirely because we do not have to spend time and money travelling to the market. One can pick up the pace of his online business with the help of e-commerce application development and web development solutions. It is one of the cheapest means of doing business as it is e-commerce development that has made it possible to reduce cost of promotion of products and services. There are no time barriers in selling the products. One can log on the internet even at midnight and can sell products at a single click of mouse. An interactive user friendly and focused website in the form of online shop can generate good business. So we are of the opinion that big companies should invest more on research and development for e-commerce.

###### Paper [3] D. Carlander Reuterfelt, A. Carrera, C. A. Iglesias, O. Araque, J. F. Sánchez Rada and S. Muñoz, "JAICOB: A Data Science Chatbot," in IEEE Access, vol. 8, pp., 2020.

In this paper, the author described as to identify the way students learn and the types of questions. Different types of requirements for different types of learning (inductive and deductive) were identified due to the nature of students’ curiosity, and the specifics of the topic. The following pedagogical solutions were identified:

1. A definition of a concept is a consequence of the usual teaching style, which is deductive, starting from the main concepts and developing towards the applica tions. It is part of the process of learning, but cannot be the whole process. In the Oliver model, definitions provide learning content.
2. As stated in the learning of programming techniques can be enhanced by using examples of code using analogy and induction. Also, learning is significantly facilitated by examples in initial coding attempts. Furthermore, surveys suggest that engineering students usually view themselves as inductive learners. In the Oliver model examples can provide learner support.
3. Lastly, the human need for small-talk, such as joking and asking for the weather, must be satisfied to provide a more significant communication source. With that in mind, the architecture was designed, having identified the pedagogical needs of the student. There are several steps involved in the process and are explained below. A Knowledge Base (KB) was populated with pertinent information regarding the topic at hand, to satisfy the requests for definitions and examples. The Question Answering (QA) module is designed to extract meaning from all the data with the pedagogical requirements in mind to make sense of that information. To analyze the students’ question, we use the Speech Act Classifier. It selects the module where the question must be delegated. If small talk is detected, it is passed onto the Small Talk module or into the QA Module if a question regarding Data Science is detected.

###### Chatbot

The use of chatbots has become prevalent in the last years in shopping, customer support, general assistance, and, though less developed, education. The use of chatbots as a form of e-learning brings lots of opportunities. This article identified the advantages of cognitive assistants in education and the corresponding challenges in implementation. A result is a tool for students with a comfortable and usable interface and a human experience. It can provide insights and solve doubts about Data Science. The main contribution is the adaptation of students’ real pedagogic needs to the design of the architecture and being flexible in maintaining a conversation. Teachers can also use it as a tool to identify gaps in the knowledge of their students. They can also outsource to Jaicob the answering of all the questions. The pedagogue is also an excellent asset to select the most valuable sources of information from which Jaicob feeds from, thus providing a curated source of information instead of a regular Google Search.

###### Paper [4] G. Daniel, J. Cabot, L. Deruelle and M. Derras, "Xatkit: A Multimodal Low-Code Chatbot Development Framework," in IEEE Access, vol. 8, pp., 2020.

In this paper, the author(s) described about the Instant messaging platforms have been widely adopted as one of the main technologies to communicate and exchange information. Nowadays, most of them provide built-in support for integrating chatbot applications, which are auto mated conversational agents capable of interacting with users of the platform Chatbots have proven useful in various contexts to automate tasks and improve the user experience, such as automated customer services education, and e-commerce. Moreover, existing reports highlight the large-scale usage of chatbots in social media, and empha size that chatbot design will become a key ability in IT hires in the near future. Additional predictions say that by 2022, 80% of the companies will use chatbots and banks will be able to automate up to 90% of their customer interaction with them. The global chatbot market is projected to reach.

The widespread interest and demand for chatbot applications has emphasized the need to be able to quickly build complex chatbot applications supporting AI-based

natural language processing in order to be able to fluently chat with the user. Moreover, any non-trivial chatbot requires accessing an orchestration of internal and external services in order to perform the requested user actions (e.g. to check and query the data to be served back to the user or to actually execute some processes/actions in response). As such, chatbots are becoming complex software artifacts that require a more methodical development approach to be developed with the proper quality standards. As such, the definition of chatbots becomes a challenging task that requires expertise in a variety of technical domains, ranging from natural language processing to a deep understanding of the APIs of the targeted instant messaging plat forms and third-party services to be integrated.

###### Xatkit

In this paper we introduced Xatkit, a multi-channel and multi-platform chatbot modeling framework. Xatkit proposes a set of domain-specific languages to decouple the chatbot definition from the technical details of the platform-specific aspects where the bot is going to be deployed. This increases the reusability of the chatbot and facilitates its redeployment when the needs of the company change, including the pos sibility of evolving the NLU engine used during the text analysis phase. Moreover, the runtime component can be easily extended to support additional platform-specific actions and events beyond those already shipped with the current version of Xatkit. For instance, some platforms like Alexa or Trello have been recently added by external contributors to the core Xatkit team. Xatkit is ready to be used in real-case scenarios. But it has still plenty of room for improvements. At the language level we plan to improve the variability of the bot specification, moving towards a product-line approach that enables companies to create and quickly update several versions of the same bot (e.g. to create localized versions of the bot for each branch of the company). At the framework level, we plan to work on the integration of chatbot generators, able to create partial bot specifications from existing data sources within the company (e.g. FAQs or user guides).

###### Paper [5] E. H. Wu, C. Lin, Y. Ou, C. Liu, W. Wang and C. Chao, "Advantages and Constraints of a Hybrid Model K-12 E-Learning Assistant Chatbot," in IEEE Access, vol. 8, pp., 2020.

In this paper, the author(s) described about the advancement of modern technologies, people’s lives are more convenient than ever before. Education and ways of learning have also begun to change with the development of new technologies that improve learning efficiency and effectiveness. One such example of these new technologies is the Summit Learning Platform, an online learning tool that helps students set and track learning goals at their own pace. E-Learning is gaining popularity and is widely used in today’s modern education. In many higher learning institutions, such as universities, are using E-Learning plat forms because younger students are familiar with these new The associate editor coordinating the review of this manuscript and approving it for publication was Chongsheng Zhang. technologies and are potentially attracted by new types of learning methods.

###### E-Learning

E-Learning is a wonderful way to learn. It utilizes many convenient platforms, such as smartphones and tablets, so students can learn and study anytime, anywhere. However, learning on E-Learning platforms lacks interaction between students or teachers which often causes students to experience feelings of isolation and detachment. Evaluation results show that our proposed hybrid chatbot is able to reduce these negative feelings. Responses from our chatbot are relatively close to what a real human would say, and chatting with our chatbot is a fun and entertaining experience. Compared with the teacher counselling services provided by the E-Learning platform on which our chatbot is based, our chatbot does have advantages in chatting with students. These advantages include interesting conversations and instant responses at any time of day or night. However, our chatbot still falls short of real human teachers in solving learning problems due to the fact that its current datasets and knowledge base are still in infancy, a huge constraint for any kind of chatbot. It is important that a chatbot designed for educational use has rich, accurate responses for course-related questions.

We must invest more effort into adding more course-related and educational materials to our chatbot. Both the response con texts and the hybrid model still have room for improvement. Our experiments and questionnaires need tweaks as well, since the questions are not fully optimized for the comprehension ability of K-12 students. Advantages and Constraints of a Hybrid Model K-12 E-Learning Assistant Chatbot E- Learning platform on which our chatbot is based. However, we do have a general picture of how our chatbot performs as an E-Learning assistant. Even though most users feel that our chatbot is not like a real human companion, we think our chatbot does perform well as a dedicated E-Learning companion and assistant. In our future work, we want to improve our bot’s ability to understand and solve more learning-related problems. Our designed topics in the dialogue module, the hybrid model’s performance, the response contexts, and the evaluation design still have room for improvement. The concept of combining models for different purposes in one chatbot can be used in other domains.

###### Paper [6] Boris Galitsky, Dmitry Ilvovsky "Chatbot with a Discourse Structure- Driven Dialogue Management", Association for Computational Linguistics, 2017.

In this paper author described about the Modern search engines have become very good at understanding typical, most popular user in tents, recognizing topic of a question and providing a relevant links. However, search engines are not necessarily capable of providing an answer that would match a style, personal circumstances, knowledge state, an attitude of a user who formulated a query. This is particularly true for long, complex queries, and for a dialogue-based type of interactions. In a chat bot, the flow query – clarification request - clarification response - candidate answer should be cohesive, not just main tain a topic of a conversation. Moreover, modern search engines and modern chat bots are unable to leverage an immediate, explicit user feedback on what is most interesting and relevant to them. The chat bot we introduce in this demo pa per is inspired by the idea that knowledge exploration should be driven by navigating a single dis course tree (DT) built for the whole corpus of relevant content. We refer to such tree as extended discourse tree.

###### Discourse tree:

Moreover, to select rhetorically cohesive answers, chat bot should be capable of classifying question-answer pairs as cohesive or not. This can be achieved by learning the pairs of extended discourse trees for the question-answer pairs. A question can have an arbitrary rhetoric structure as long as the subject of this question is clear to its recipient. An answer on its own can have an arbitrary rhetoric structure. However, these structures, the DT of a question and the DT of an answer should be correlated when this answer is appropriate to this question. We apply a computational measure for how logical, rhetoric structure of a request or a question is in agreement with that of a response, or an answer. Over last decade, Siri for iPhone and Cortana for Windows Phone have been designed to serve as digital assistants. They analyze input question sentences and return suitable answers for users queries (Crutzen et al., 2011). However, they assume patterned word sequences as input commands. Moreover, there are previous studies that combine natural language processing techniques with ontology technology to implement computer system for intellectual conversation. There are chat bot systems including ALICE3, which utilizes an ontology, like Cyc, API.ai and Amazon Lex, the platforms for developers to build chat bots. Most of these systems expected the correct formulation of a question, certain domain knowledge and a rigid grammatical structure in user query sentences; however, less rigid structured sentences can appear in a user’s utterance to a chat bot. Developers of chat bot platforms.

###### Paper [7] L. Zhang, Y. Yang, J. Zhou, C. Chen and L. He, "Retrieval-Polished Response Generation for Chatbot," in IEEE Access, vol. 8, pp., 2020.

In this paper author(s) described about the Chatbot communication, an essential task in both natural language processing and artificial intelligence fields, involves a robot communicating with human beings using natural language in open domains. Chatbots play a critical role in many real-world applications, such as smart speakers, customer service systems , and social robots. Research on chatbots began in the 1960s

; initially, researchers used sets of handwritten rules and templates. However, such rule- based models require significant human effort and lack flexibility. In recent years, as

large-scale dialog corpora and high-speed computational resources have become available, the early rule-based models have been rapidly replaced by data-driven models. The existing data-driven models can be categorized as retrieval-based or generation-based models. Retrieval-based models return responses from a corpora by computing contextual similarity.

###### Chatbot Responses:

The retrieved responses are usually fluent and informative because they are written by humans; however, they may contain irrelevant content. In contrast, generation-based models return responses generated based on language rules learned The associate editor coordinating the review of this manuscript and approving it for publication was Imran Sarwar Bajwa . during the training process. Thus, the generated responses are relevant but may suffer from problems such as being ‘‘safe responses’’, lacking fluency or including grammatical errors. An excellent conversational robot should return replies that are contextually relevant, informative, and fluent. Therefore, researchers have proposed using retrieved results as a basis for response generation and this approach has made progress. However, the combined methods do not actually take advantage of both methods, and their results tend to be similar to retrieved responses. To address this issue, we introduce a polishing process into the generation model that was inspired by writing articles. Human authors typically write an early draft and then polish it in detail. When polishing a specific sentence, authors tend to adopt writing styles and techniques from existing literature. In terms of response generation, the context that consists of the sentences that immediately precede the response pro videos background knowledge to generate a draft response, while retrieved responses provide information about the language style and techniques that can be used to polish the draft response.

###### Response Generation:

In this work, we propose a retrieval-polished response generation method for human-robot dialog. In contrast to the existing approaches, our method polishes a draft response by considering a contextually similar prototype and finally chooses the better of the retrieved and polished responses as the final reply. Our method uses both

the background provided by the context and the sentence style provided by the retrieved response.

###### Paper [8] L. Zhang, Y. Yang, J. Zhou, C. Chen and L. He, "Retrieval-Polished Response Generation for Chatbot," in IEEE Access, vol. 8, pp., 2020.

In this paper author(s) described about the Semantic Web shows great potentials in the e-commerce. A key word searching in current online shopping is not only tedious and time-consuming but also often results in large amounts of irrelevant information. Current Web services offer only an inflexible interface with some human oriented metadata that describes what the service does, and which organization developed it. Web Services are typically intended for applications consumption, in contrast with contemporary Web applications which are meant for human users. However, the lack of machine readable semantics is hampering their usage in complex business environment. Semantic Web Services (SWS) come along and provide a solution with rich formal descriptions of their capabilities, thus facilitating automated discovery, dynamic binding, and invocation of services within an open environment, which can be utilized by applications or other services without human assistance and immune to highly constrained agreements on interfaces or protocols. According to a semantic specification in ontology, a commercial infrastructure can be featured for a better communication between buyer and seller.

###### Paper [9] Lei Cui, Shaohan Huang, Furu Wei, Chuanqi Tan, Chaoqun Duan, and Ming Zhou "SuperAgent: A Customer Service Chatbot for E-commerce Websites", Association for Computational Linguistics, 2017.

In this paper, the author describing about the Customer service plays an important role in an organization’s ability to generate income and revenue. It is often the most resource-intensive department within a company, consuming billions of dollars a year to change the entire perception customers hold. Support staff spend a lot of time answering questions via telephone or messaging applications to make sure are customers satisfied with their business. This traditional customer service has two problems: First, staff usually receive repetitive questions asked by a variety of customers, which can be cost-effectively answered by machines. Second, it is difficult

to support 7×24 services, especially for most non global businesses. Therefore, chatbots can be a great way to supplement customer service offerings since they are more economical. The first two authors contribute equally to this work. gable, and free up support staff to answer much higher value queries. Recently, virtual assistants for customer service have become more and more popular with customer oriented businesses. Most of them are built from human conversations in the past, which is straightforward but faced with problems of data scale and privacy. Most of the time, customers need to wait online to get a support staff person’s answer, which is less effective and difficult to scale up. Meanwhile, customers may have privacy concerns about the conversations, hence conversations with customers cannot be easily leveraged to train a chatbot. It is essential to find large-scale and publicly available customer service data sources on which to build such assistants.

#### Chapter 3 REQUIREMENT SPECIFICATIONS

##### REQUIREMENT SPECIFICATIONS

###### Functional Requirements

* + - The website should work on all the devices without any interruption in the services.
    - The website should allow all users to get authenticated and verified with their Emails
    - Website uses the Internet to connect with server and display ll the products to the user.
    - Website will store the basic data of the User information like Name, Email ID, Phone, *isUserSignedIn* Status.
    - Details which are entered by the user should be correct else the information which is displayed may not be accurate.
    - Users can communicate to customer service through Chatbot directly from the Website itself.
    - Customers can add the selected products to the cart and order the items with payment options.
    - Several Payment options are provided to the user like Credit card, Debit cards, UPI payments etc.
    - Users can log out from the account anytime from the website with the help of Sign out Button.
    - Users can sign in to their account from any device and their basic data gets loaded.
    - Only one time sign in, users are not required to sign in every they use the website.

###### Non-Functional Requirements

* + - Availability
    - Performance
    - Reliability
    - Uptime of servers
    - The website start time should be within less than 10 seconds
    - Website should work efficiently
    - The list of products displayed should be correct and no incorrect details should be shown.
    - Website should be given basic permission like INTERNET, and STORAGE for uninterrupted use of the app.
    - The data should fetch from the server within less than 3 seconds and there should be no any delay while retrieving the data.
    - Usability: Regardless of the size of your business, the website of your business should be easy to use for even a non-technical user.
    - App size is minimized to make the app run efficiently and accurately on the user’s device.

###### Software Requirements

###### Frontend Frameworks:

* + - * HTML
      * CSS
      * BootStrap
      * JavaScript
      * Chatbot
      * RazorPay Button

###### Backend Frameworks:

* + - * Python
      * Flask
      * PHP
      * NodeJS
      * Django

###### Hardware Requirements

###### Windows Version:

* + - * 64-bit Microsoft® Windows® 8/10
      * x86\_64 CPU architecture; 2nd generation Intel Core or newer, or AMD CPU with support for a Windows Hypervisor
      * 8 GB RAM or more
      * 8 GB of available disk space minimum (IDE + Anaconda or Jupyter Notebook)
      * 1280 x 800 minimum screen resolution

#### Chapter 4 METHODOLOGY

##### METHODOLOGY

###### Implementation of Basic Pages

###### Creation of Template by Using Flask

Hello World Program: The basic example of Flask is a Hello World program in Flask where we are importing Flask class using import function, then we are defining hello world function and then returning ‘Hello World!’.

###### Template Inheritance

Template Inheritance in Flask allows developers to build a base template which can be overridden by child template. It uses {%block%} and tells base template that child template may override the functionality defined in {%block%}.{%extends %} section defines that this template can “extend” another template. When the template system evaluates this template, first it locates the parent. The extends tag must be first in the template. To provide the contents of a block defined in the parent template, use

{{super()}}.

###### File Organization

File Organization is used to define the way in which the files are arranged in the directory and it shows the way in which they can be called. Example “/” is used to define the index page.

###### Python support for web development

Example: Hand gestures is an application which is efficiently designed in python by using some external libraries like open cv. The open cv is the library which can be used in the real time applications which advantages in multicore processing. Open cv, in this project is used to capture video from the camera as a input from the user side. Gesture recognition has been a very interesting problem in computer vision for a long time.

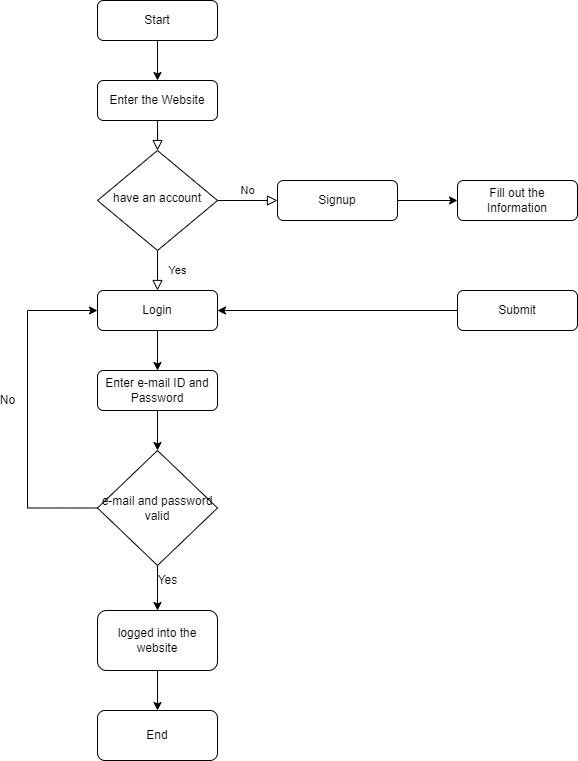
This is particularly due to the fact that segmentation of foreground object from a clustered background is a challenging problem in real time. In this implementation we are implementing few contents using open cv and python for gestures.

* + - * Segment the hand region
      * Background subtraction
      * Motion detection and threshold
      * Contour Extraction
      * Count my fingers
      * Four intermediate steps
      * Contours
      * Bitwise AND
      * Euclidian distance
      * Convex Hull

###### Defining Use Case Models

Our goal was to develop a web application that would be attractive enough, have a professional look and user friendly. So that people of all age groups would be its end users. Our job started with subdividing the entire task and setting milestones. The milestones would be a marker of percentage of the work actually accomplished and success story. The entire planning process took the following steps.Writing use cases or stories of using a system an excellent technique to understand and describe requirements. An end user with internet browsing facility enabled registers into our site and logs into our site. Finds products of his interest using the search option. Adds them into the shopping cart and finally orders the products online when the electronic copy of the bill is automatically generated. So, from the stated use case model we found out the following to be the primary requirements:

* A registration page
* Search option
* Shopping cart
* Billing system



###### Figure 4.2 Login Module Architecture

###### Domain Modeling

As with most of the web applications developed using the Object Oriented Programming (OOP) we followed the same. So we moved forward for Object Oriented (OO) analysis which emphasizes on finding and describing the objects – or concepts – in the problem domain. For example, a product in our system is an object.

a

###### Architectural Pattern

Our application has been developed using the standard “Model-View- Controller” pattern. Model view controller (MVC) is an application architectural pattern for implementing user interfaces. It divides the application into three interconnected parts; so as to separate internal representations of information from the

ways that information is presented to accept from the user. ‘Code igniter’ is an open source web development framework that provided us withthe support to build our application using PHP following MVC pattern. Hence view works as the user interface. Controller has the essential class files to manipulate the data stored in the backend

i.e. the database. It actually works as a traffic between the model and view. But it doesn’t have the access to directly interact with the database.

###### Web Application Working

Our application comprises of dynamic web pages which has been created both client and server side scripts. A dynamic web page is a web page that is generated by a server-side program or script. For the testing purpose we have hosted it on our local hosts i.e. our personal computers later we hope to host it on hosting sites to use it professionally. So when we run the program on our local computer the web server is the local web server. The browser like Google Chrome, Mozilla Firefox or Safari executes this program. The browser makes a Hypertext Transfer Protocol (HTTP) request to the web server for a specific dynamic web page, the web server then looks up the extension of the requested file to find out which application server should process the request. When the application server receives a request, it runs the specified script. Often, this script uses the data that it gets from the web browser to get the appropriate data from a database server. This script can also store the data that it receives in the database. When the application server finishes processing the data, it generates the HTML for a web page and returns to the web server. Then, the web server returns the HTML to the web browser as part of an HTTP response.

###### Modular Design

The proposed application is implemented using HTML5, CSS, Java script and PHP.The currently working e- commerce shopping websites provide the features of a sophisticated shopping cart, on-site wish list. This paper proposed an e-commerce shopping website to sell and promote only the Indian products. The functionality of the proposed application is divided into number of sub modules. The modules to be taken into account are customer shopping cart module, orders, payment and product

module. These modules are integrated together give the functionality desired out of the application.

###### Proposed architecture for cognitive bot

The first step to design the proposed architecture was to identify the way students learn and the types of questions. Different types of requirements for different types of learning (inductive and deductive) were identified due to the nature of students' curiosity, and the species of the topic.

###### The following pedagogical solutions were identified:

1. A definition of a concept is a consequence of the usual teaching style, which is deductive, starting from the main concepts and developing towards the applications. It is part of the process of learning, but cannot be the whole process. In the Oliver model , definitions provide learning content.
2. As stated, the learning of programming techniques can be enhanced by using examples of code using analogy and induction. Also, learning is significantly facilitated by examples in initial coding attempts. Furthermore, surveys suggest that engineering students usually view themselves as inductive learners. In the Oliver model , examples can provide learner support.
3. Lastly, the human need for small-talk, such as joking and asking for the weather, must be satisfied to provide a moresignificant communication source. With that in mind, the architecture was designed, having identified the pedagogicalneeds of the student. There are several steps involved in the process. A Knowledge Base (KB) was populated with pertinentinformation regarding the topic at hand, to satisfy the requests for definitions and examples. The Question Answering (QA) module is designed to extract meaning from all the data with the pedagogical requirements in mind to make sense of that information. To analyze the students' question, we use the Speech Act Classifier. It selects the module where the question must be delegated. The way it works will be explained in greater detail in Section III-B. If small talk is detected, it is passed onto the Small Talk module or into the QA Module if a question regarding Data Science is detected. Afterward, the modules generate an answer to satisfy the student request. The answer is sent back to the student, and

feedback is collected to evaluate and improve the model.

###### Models used in Chatbot

Current commercial chat bots typically use two types of models to create responses: a retrieval-based model and a generative-based model. A retrieval-based model is a common commercial chat bot choice since it is easier to develop and maintain, e.g., Pizza Hut's pizza ordering bot and the CNN News bot. This model is essentially a matching process between the input question and an output answer. We can use cosine similarity or a trained CNN, LSTM model to calculate the matching probability between a question and an answer. All bot responses are maintained and saved in databases. When an input question comes in, this model will calculate the matching probability of the input question and all responses in the database. The higher the matching probability is, the more probable that the response is right. Google's automated email reply suggestion system is powered by the same type of retrieval- based model. This model is great for a QA system that needs precise and accurate responses since responses are maintained in databases. The quality of responses can also be ensured. However, this model suffers from low coverage in responses like because natural language is complex. It cannot answer a question or input sentence that is not in the coverage of pre-defined responses. Building a database that contains all kinds of topics, answers and responses is incredibly time consuming, hard to maintain and difficult to cover all possible responses in natural language. Unlike a retrieval- based model, a generative-based model does not store pre-defined responses in a database. It creates a response by the model itself so it can use it in applications of open or wide coverage responses like chitchatting.

###### Personalized and Interactive Domain

Most chat bots are designed to imitate human intellectual activity maintaining a dialogue. The purpose of the chat bot with iterative exploration is to provide most efficient and effective information access for users. A vast majority of chat bots nowadays, especially based on deep learning, try to build a plausible sequence of words to serve as an automated response to user query. Such most plausible responses, sequences of syntactically and semantically compatible words, are not necessarily most

informative. Instead, in this demo we focus on a chat bot that helps a user to navigate to the exact, insightful answer as fast as possible. For example, if a user is formulated her query Can I use one credit card to pay for another, the chat bot attempts to recognize a user intent and a background knowledge about this user to establish a proper context.

###### Methods

###### Retrieval Based Method

One simple implementation of a retrieval-based model involves computing the cosine similarity between the input utterances and the candidate replies. To improve the relevance of the responses, researchers have proposed a series of matching algorithms that consider contextual information. SMN matches a response with each utterance in the context of multiple levels of granularity. DAM not only constructs text segment representations at different granularities solely using stacked self-attention but also extracts well-matched segment pairs with attention across the context and response.

###### Generation Based Method

Most generation-based methods are based on a variant of the sequence-to-sequence (seq2seq) framework. Studies on improving informativeness can be divided into two categories. The first category directly optimizes the seq2seq model: Li *et al. proposed the use of maximum* mutual information (MMI) as the objective function, and Zhang *et al. optimized a variational lower bound* on the pairwise mutual information between context and response. The second category introduces external elements such as topics keywords, and external knowledge to seq2seq models to increase the informativeness.

###### Chit-chat Conversation Modeling

The chit-chat engine is mainly designed to reply to greeting queries such as “hello“ and “thank you”, as well as queries that cannot be answered by the previous three engines, such as “you are so cute”. However, general chit-chat engines tend to be topic- deviated so that the replies may be irrelevant. To avoid such deviations, we follow the smart reply approach for email reply suggestions to predefine a permitted response set. Formally, the chit-chat model is an attention-based LSTM seq2seq mode trained on twitter conversation data.We select the 5 million most frequent non-duplicate short

replies as the permitted response set, most of which are greetings and common replies. The end-to-end perplexity of the seq2seq model, which is similar to result. The engine’s output is very topic-coherent, which is shown as follows:

Q: hello

R: hey how are you?

Q: thank you

R: you’re very welcome sir Q: you are so cute

R: u r more

###### Meta Engine

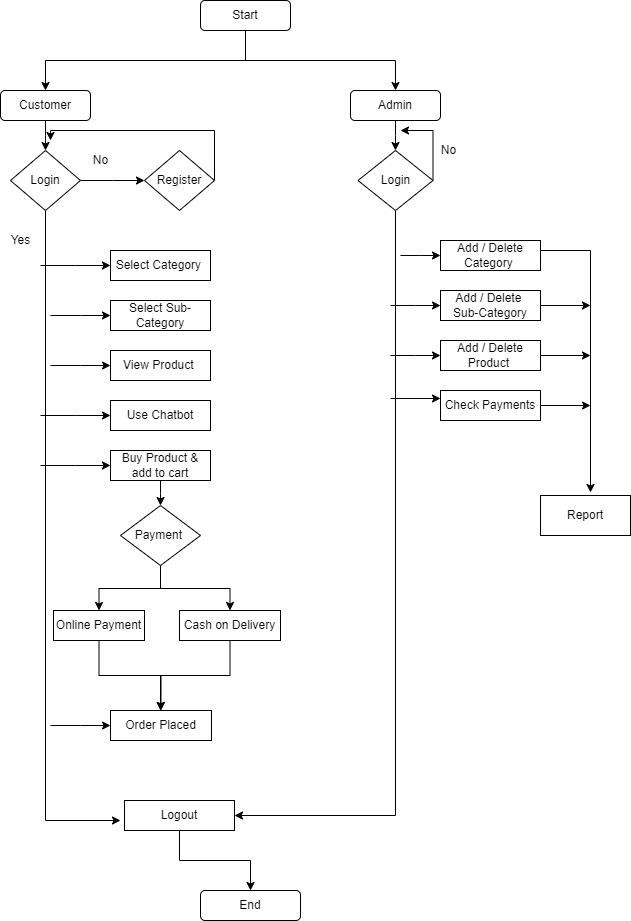
For each query, Super-Agent will call the abovementioned sub-engines in parallel. The meta engine is then used to merge and prioritize the results from the different engines. We use a simple strategy to implement the meta engine, which prefers results from the engines in order of fact QA, FAQ search, text QA and chit-chat engine according to tunable threshold.

#### Chapter 5 SYSTEM DESIGN

##### SYSTEM DESIGN

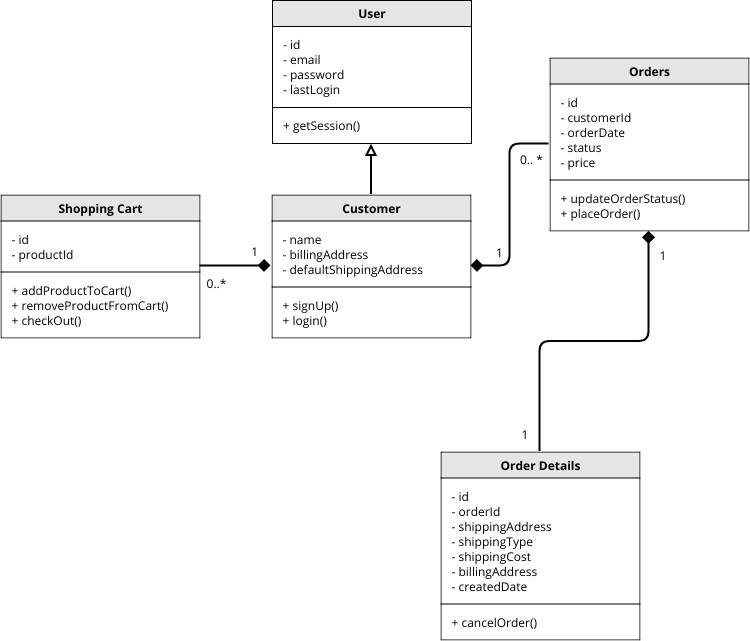
###### UML Diagrams

###### Activity Diagram



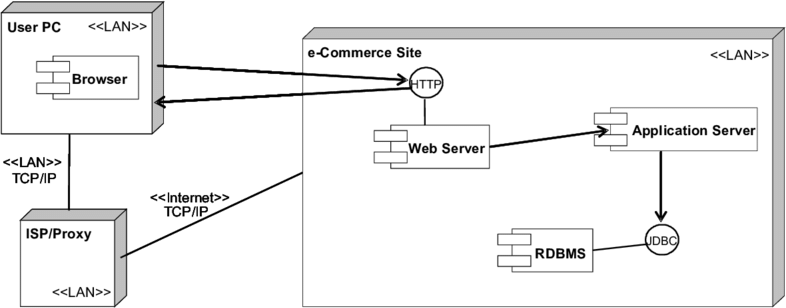
###### Figure 5.1 Activity Diagram

###### Class Diagram



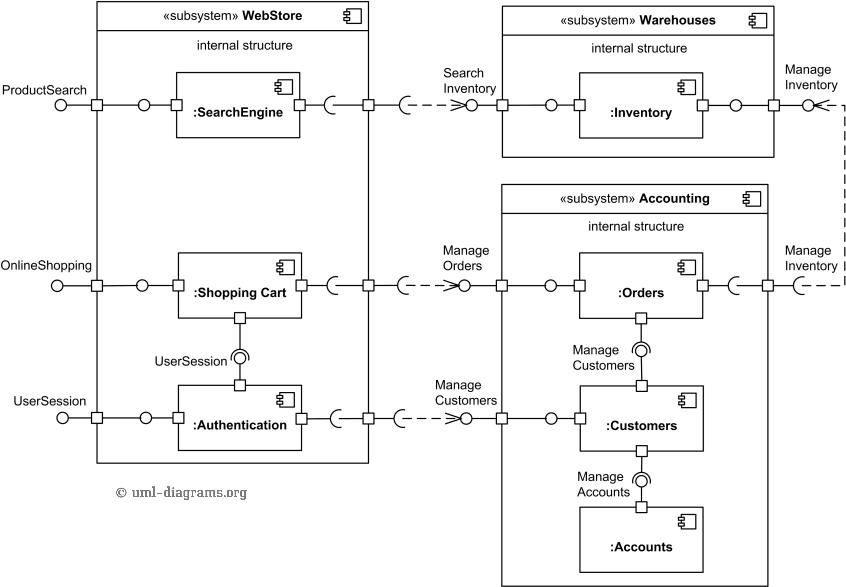
###### Figure 5.2 Class Diagram

###### Deployment Diagram



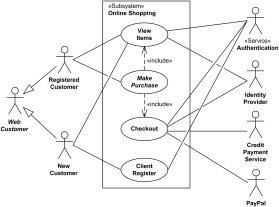
###### Figure 5.3 Deployment Diagram

###### Component Diagram



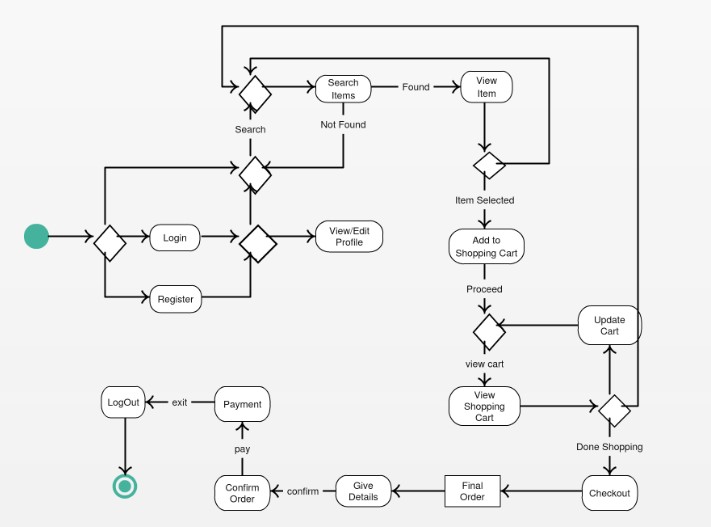
###### Figure 5.4 Component Diagram

###### Use case diagram



###### Figure 5.5 Use case diagram

###### State Chart Diagram



**Figure 5.6 State Chart Diagram**

#### Chapter 6 RESULTS AND DISCUSSIONS

##### RESULTS AND DISCUSSIONS

###### Rise in use of E-Commerce Websites

While understanding the nitty gritties of its different aspects with special emphasis on B2C e-commerce. Which has shown tremendous growth in the recent years because of increased consumer awareness, investor trust and technological proliferation. The study has also produced certain trends and factors which shall propel further growth within the e-commerce market in India. A sustainable business practice would be to push e-commerce and m-commerce as complements instead of substitutes to traditional business. The synergies between offline and online businesses will enhance efficiency and end in a more stable existence. The longer term also entails a scenario which shall witness an increase in niche businesses, as well as mergers and acquisitions to enable firms to grow inorganically. Firms must commit themselves to the provision of services and application of internet marketing, to get and retain larger audience. Through mobile penetration, opportunities are emerging within the rural markets which is mobile app infrastructure, content development in local languages and walk connectivity. Also, within the aftermath of demonetization. The share of digital payments is anticipated to rise, to sustain which, firms must develop safer payments infrastructure.

In order to implement the dynamic functionality, the server contains a PHP file that serves as a medium of interaction between the Chat Client and the MySQL database. When a trigger that contains an object macro is called, the response is parsed and then executed by the Interpreter's Javascript Object handler. The response makes an AJAX request to the PHP page, and on receiving a response, displays it within the chat window. This response contains a hyperlink to the respective product pages of the suggested products. Since the responses are not hardcoded, but depend on the user and the backend database, the chatbot is capable of providing updated information. In a scenario where the backend database is updated with newer products, the chatbot will make suggestions taking into account the updated database.

Thus we have implemented a website based chatbot that attempts to improve User Interaction with the E-Commerce website. The chatbot has a stored set of responses,

but also takes dynamic user input into account and thus tends to provide relevant responses and product suggestions. Since the product database is independent of the stored responses, newer products under the respective category can be easily added and removed and require no modification of the stored chatbot responses.

Online websites can reach to broadest audience and are not restricted to a platform that constrain native applications. The viewers or customers can have easy access i.e. from a desktop/laptop to a mobile device, as websites have the capability to display the content through web browser. In comparison to native application, web browsing is much simpler because it does not require users to access the app store on their devices or download their app (which may include one or more processes in accessing to your contents). Distribution of your data is much more flexible and agile with web-based application than native ones as there are no stringent app store requirements and content restrictions to follow.

Instead of establishing a physical outlet for the company, if the business owner develops a website as an online forum or similar, there will be good chance of gaining larger audience online to link up. This is because, most of the people are wired up with the Internet all day. Generally, people prefer to go for the smartest way to check online first and then take a decision. So if the business owner fills all the basic details of the product and make a secure way to get the product to the customer in timely manner, then people will prefer buying online instead of visiting the outlet physically. This also allows people to access it even in the oddest hour of the day.

###### Advantages of Python for Backend

Python and Flask are portable and interactive languages for web development including dynamic semantics potential. In python it is also possible to bind new modules to python to extend its core functionality. Many of the great websites are moving to python due to its robustness. It can be of great databases and very easy way to use standard libraries. Powerful text processing is also enabling. Python is a fast growing technology having many features. This paper can conclude as python can be more efficient than any other web development languages, because python is fast,

broad and better than java, perl, tcl etc. Python Jinja flask are more useful for maintaining complexity technologies. These are also helps for data fetch from [WWW.](http://WWW/) It is more powerful, fast with the help of flask Template Engine. Therefore python is great for backend web development.

A sustainable business practice would be to push e-commerce and m-commerce as complements instead of substitutes to traditional business. The synergies between offline and online businesses will enhance efficiency and end in a more stable existence. The longer term also entails a scenario which shall witness an increase in niche businesses, as well as mergers and acquisitions to enable firms to grow inorganically. Firms must commit themselves to the provision of services and application of internet marketing, to get and retain larger audience. Through mobile penetration, opportunities are emerging within the rural markets which is mobile app infrastructure, content development in local languages and walk connectivity. Also, within the aftermath of demonetization. The share of digital payments is anticipated to rise, to sustain which, firms must develop safer payments infrastructure.

Python is a high-level programming language that has English-like syntax. This makes it easier to read and understand the code. Python is really easy to pick up and learn, that is why a lot of people recommend Python to beginners. You need less lines of code to perform the same task as compared to other major languages like C/C++ and Java. It is a very productive language. Due to the simplicity of Python, developers can focus on solving the problem. They don’t need to spend too much time in understanding the syntax or behavior of the programming language. You write less code and get more things done. It is an interpreted language which means that Python directly executes the code line by line. In case of any error, it stops further execution and reports back the error which has occurred. Python shows only one error even if the program has multiple errors. This makes debugging easier. Python doesn’t know the type of variable until we run the code. It automatically assigns the data type during execution. The programmer doesn’t need to worry about declaring variables and their data types.

###### Online Shopping

Online shopping is the process whereby consumers directly buy product services from seller in real time, without any intermediary service. Over an internet. World Wide Web has become a major resource in modern business. It gives new opportunities to business.

An online website can be compared to a traditional shop’s interior. The website is attractive, provides easy way for navigation, multiple options in terms of brands, color and design the customer would stay on the site. This paper supports a flexible, attractive and an easy to use environment for an online website application along with the addition feature like an offsite add to cart. Hence as future enhancements to the proposed system.

###### Chatbot developed by Xatkit

Xatkit is a multi-channel and multiplatform chatbot modelling framework. Xatkit proposes a set of domain-specific languages to decouple the chatbot definition from the technical details of the platform-specific aspects where the bot is going to be deployed. This increases the reusability of the chatbot and facilitates its redeployment when the needs of the company change, including the possibility of evolving the NLU engine used during the text analysis phase.

Moreover, the runtime component can be easily extended to support additional platform specific actions and events beyond those already shipped with the current version of Xatkit. For instance, some platforms like Alexa or Trello have been recently added by external contributors to the core Xatkit team.

Xatkit is ready to be used in real-case scenarios. But it has still plenty of room for improvements. At the language level we plan to improve the variability of the bot specification, moving towards a product-line approach that enables companies to create and quickly update several versions of the same bot.

At the framework level, we plan to work on the integration of chatbot generators, able to create partial bot specifications from existing data sources within the company

(e.g. FAQs or user guides). We also plan to study the combination of sentiment analysis and behavioral design patterns to create more likeable and effective chatbots.

This proves that our chatbot’s responses are on par with how humans respond**.** Result of a chatbot response context design test. Red represents the ratio that a user chose the chatbot-generated answer. Blue represents the ratio that a user chose human- generated answer. Average ratio of a chatbot response context design test. Red represents the average ratio that a user chose the chatbot generated answer. Blue represents the average ratio that a user chose the human-generated answer. to a question. Most users will not be able to tell whether one is chatting with a bot or a human. This result also proves that our chatbot can produce human-like responses and conversation experiences that could help reduce feelings of detachment in the user. Ratio of how many participants have feelings of isolation and detachment in three different situations. Finally, we collected questionnaires from all 53 participants. The first category we will discuss is feelings of isolation and detachment. shows how many of our participants had feelings of isolation and detach mention three different situations. The left bar in shows results from participants using only the E-Learning platform without other services. About 74% of participants experienced feelings of isolation and detachment while using the E-Learning platform. The middle bar in Fig 9 shows results from participants using the same E-Learning platform with teacher counselling services provided by the platform. The ratio of participants who experienced feelings of isolation and detachment dropped to about 55.5%, proving that interaction with a human teacher could reduce feelings of isolation and detachment (even though over 50% of participants still experience the issue). The right bar shows results from participants using the same E-Learning platform with our designed chatbot. Replacing teacher counselling services with our chatbot could further reduce the ratio of participants who experience this issue to about 41.5%, proving that our chatbot could reduce feelings of isolation and detachment with greater effect than teacher counselling services.

###### SuperAgent Chatbot

SuperAgent takes advantage of data from in-page product descriptions as well as user generated content from e commerce websites, which is more practical and cost- effective when answering repetitive questions, freeing up human support staff to answer much higher value questions. We demonstrate SuperAgent as an add-on extension to mainstream web browsers and show its usefulness to user’s online shopping experience. SuperAgent is integrated into e-commerce websites as an add-on extension, which can directly improve customer’s online shopping experience.

We have developed SuperAgent, a customer service chatbot for e-commerce websites. Compared to conventional customer service chatbots, SuperAgent takes advantage of large-scale, publicly available, and crowd-sourced customer data. In addition, SuperAgent leverages state-of-the-art NLP and machine learning techniques, including fact QA, FAQ search, opinion-oriented text QA, as well as chit-chat conversation modeling. Usability analysis shows that SuperAgent has improved the end-to-end user experience in terms of online shopping. It is more convenient for customer’s information acquisition especially when a product page contains too much user-generated content. In the future, we will focus on two main problems. First, we need to integrate a customer’s query intent detection module, so that we can better leverage individual engines. Second, we will have a deeper delve on multi-turn queries, where context modeling will be further investigated.

We think this is due to two factors: 1) Our chatbot responds almost instantly at any time compared to teacher services, which means users have more time to be with our chatbot. 2) Unlike teacher counselling services, our chatbot can chat casually with users. This is not possible for teacher counselling services due to its dedication to question answering and limited resources.

The second category we will discuss is the comparison between our chatbot’s E- Learning course-related QA performance to that of the teacher counselling service. the average user preference score of the chatbot and the teacher service. The top bar inshows how thorough and easy-to-understand the response of our chatbot is compared to the teacher counselling service. The average score (mean) is 2.61, meaning that our

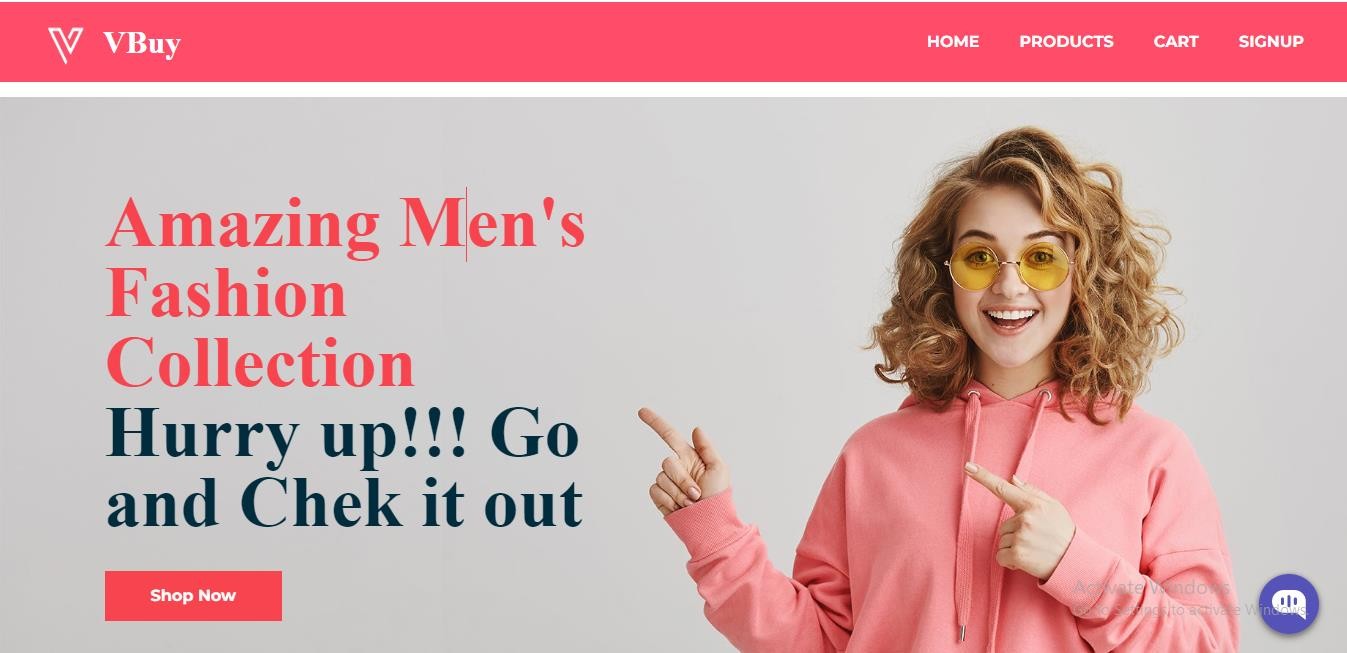
chatbot is comparable to the teacher service in this area (and actually scored . Average user preference scores in E-Learning course related question-answering performance between our chatbot and the teacher counselling service. A score of 0 means the chatbot performed much worse than the teacher counselling service. A score of 4 means the chatbot performed much better than the teacher counselling service. User preference score distribution between our chatbot and teacher counselling services in E-Learning course-related question-answering performance. A score of 0 means the chatbot performed worse than the teacher counselling service. A score of 4 means the chatbot performed better than the teacher counselling service. slightly higher). rated a score of 2 or 3 and the standard deviation is 0.927. We think the reason for this score is that our chatbot not only provides quality answers but also provides additional tutorial information, such as related course videos, at the user’s fingertips. The middle bar in compares the accuracy and reasonableness between our chatbot’s responses and the teacher counselling service. The average score (mean) is 1.67. shows that most participants rated a score of 1 or 2 and the standard deviation is 0.807. In this area, our chatbot falls short compared to teacher counselling services because our chatbot’s knowledge base is still not comparable to that of a teacher or counsellor. There are still many questions that our chatbot cannot answer correctly or properly, and we are continuously improving and extending our chatbot’s knowledge base. Lastly, the bottom bar shows how fast our chatbot responds to users compared to the teacher counselling service. In this area, there is really no comparison between our chatbot and the teacher counselling service. All participants rated a score of 4 and the standard deviation is 0. This shows that a major advantage of our chatbot is that it can respond immediately at any time.

###### Payment Gateway

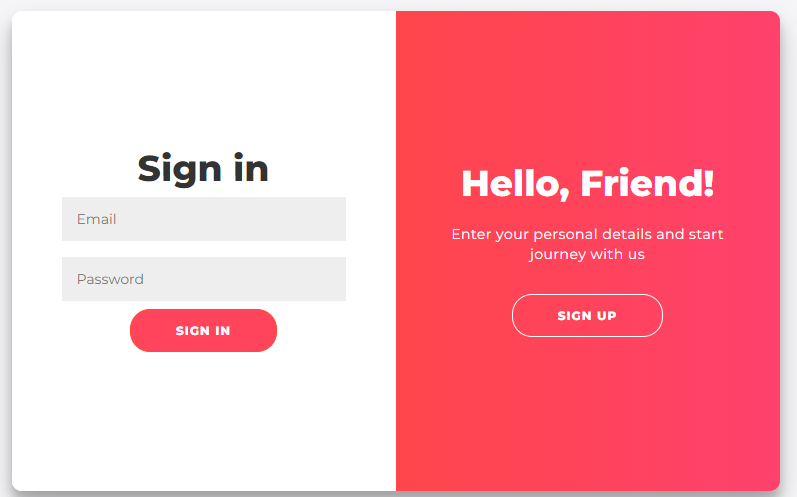
Payment integration refers to the service provided by the business provider to authorize the payment process and is being used widely in the world. Processing can be done through credit cards, cash, or direct payments. People are using payment integration for their businesses, banks, stores, and online applications. It is a very easy and reliable process for the current era. In the current context, it has become an essential

part of every business. It makes processing easy and fast. This project reflects the reliable and secure payment processing website using the help of third-party vendors. For this project, we have utilized a variety of technologies. It consists, mainly of a database (PostgreSQL server management) which used to record the data of the users. JSP is popular for creating form and standard language for documents designed to be displayed in a web browser. So, a form has created using JSP which includes the credit card number, expiration date, security code, payer name, payer address, zip code fields. There are different forms for real customers and the admin. In this project, Eclipse is used as the main front-end platform. The reason for using Eclipse is that this is the compatible platform specially to create the website. Java programing language is used to develop the website along with that CSS, HTML, bootstraps, JSP5 tag is used to create the form with the help of Eclipse. The information related to a credit card is sent to a third-party website using one of the many data transmission methods such as XML, POST, AJAX, etc. Third-party payment website examples are PayPal, etc.

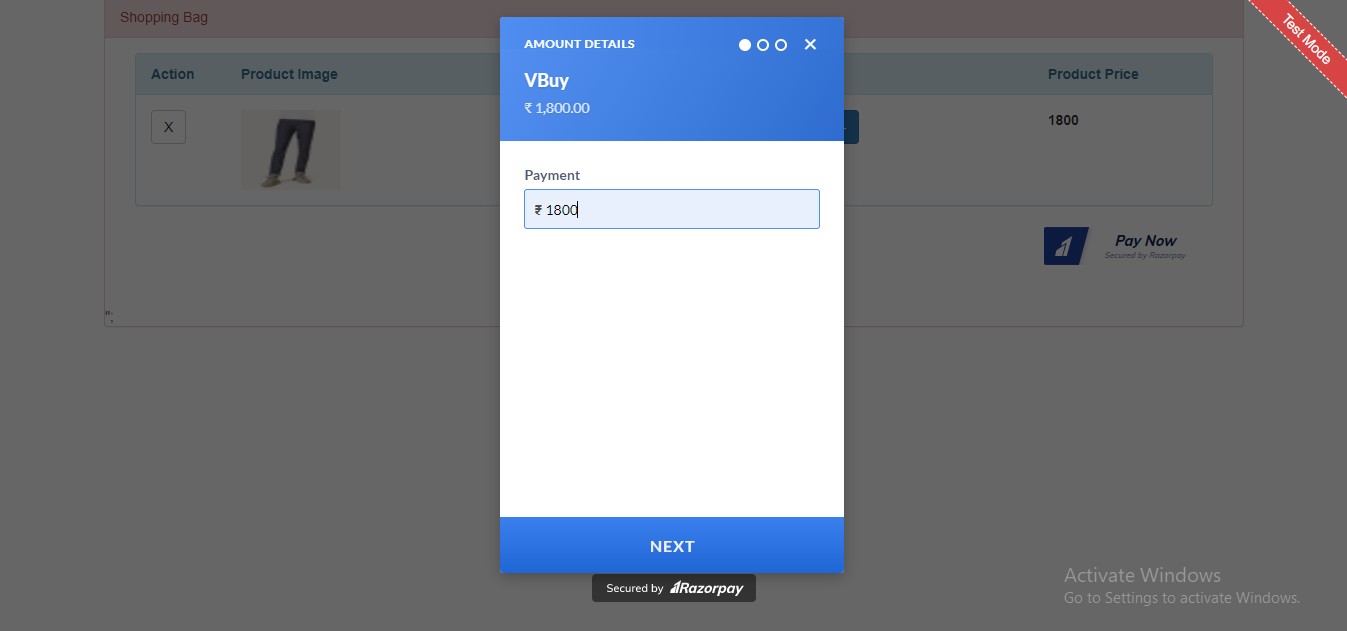
###### Proposed System



**Figure 6.1 Screenshot of the Home Page**



###### Figure 6.2 Screenshot of the Login Page



**Figure 6.3 Screenshot of the Payment Page**

#### Chapter 7 CONCLUSION AND FUTURE SCOPE

##### CONCLUSION AND FUTURE SCOPE

The main aim of our project is to create an e-commerce website with the python language using the flask web framework. The flask is the new language used for web development which provides more secured cookies and with the lesser code compared to the traditional languages. This website consists of chatbot, where users can use it as service. The chatbots gives quicker responses than the customer care service and will be available 24/7. The customers can buy the products by selecting them and making the payment in the online mode or cash on delivery method. From our perspective, chatbots or smart assistants with artificial intelligence are dramatically changing businesses. There is a wide range of chatbot building platforms that are available for various enterprises, such as e- commerce, retail, banking, leisure, travel, healthcare, and so on. In this project the implemented BOT helps to the users to know the maximum information about college related queries and provides a joyful environment to interact with the chatbot through text as a input and provide text as a output. It tries to answer the maximum queries up to its knowledge. Chatbots reach put to a large audience on massaging apps and be more effective than humans. They may be developing into a capable information- gathering tool soon.

The use of chatbots has become widespread in the last few years in healthcare, shopping, customer support, general assistance and many more promising fields. The use of chatbots as a form of e-learning brings lots of opportunities. Future work of this project is primarily based on text to speech conversation, speech to text conversation and speech to speech conversation. This idea could also help the physically handicapped people to a great extent.

#### Chapter 8 REFERENCES

##### REFERENCES

* + 1. Vangala Rama Vyshnavi, Amit Malik , “Efficient Way of Web Development Using Python and Flask” , International Journal of Recent Research Aspects Vol. 6, Issue 2, June 2019.
    2. S. E. Ullah, T. Alauddin and H. U. Zaman, "Developing an E-commerce website," 2016 International Conference on Microelectronics, Computing and Communications (MicroCom), 2016.
    3. Mr. Pankaj Gaikwad, Ms. Aditi Ghavate, “E-commerce Platform Using Website Development,” Volume 8, Issue 4, 2020.
    4. D. Carlander Reuterfelt, A. Carrera, C. A. Iglesias, O. Araque, J. F. Sánchez Rada and S. Muñoz, "JAICOB: A Data Science Chatbot," in IEEE Access, vol. 8, pp., 2020.
    5. G. Daniel, J. Cabot, L. Deruelle and M. Derras, "Xatkit: A Multimodal Low-Code Chatbot Development Framework," in IEEE Access, vol. 8, pp., 2020.
    6. E. H. Wu, C. Lin, Y. Ou, C. Liu, W. Wang and C. Chao, "Advantages and Constraints of a Hybrid Model K-12 E-Learning Assistant Chatbot," in IEEE Access, vol. 8, pp., 2020.
    7. Boris Galitsky, Dmitry Ilvovsky "Chatbot with a Discourse Structure-Driven Dialogue Management", Association for Computational Linguistics, 2017.
    8. L. Zhang, Y. Yang, J. Zhou, C. Chen and L. He, "Retrieval-Polished Response Generation for Chatbot," in IEEE Access, vol. 8, pp., 2020.
    9. J. Ni, X. Zhao and L. Zhu, "A Semantic Web Service-Oriented Model for E- Commerce," 2007 International Conference on Service Systems and Service Management, 2007.
    10. Lei Cui, Shaohan Huang, Furu Wei, Chuanqi Tan, Chaoqun Duan, and Ming Zhou "SuperAgent: A Customer Service Chatbot for E-commerce Websites", Association for Computational Linguistics, 2017.

**APPENDIX A**

#### Ch\_5.ipynb:

import nltk

from nltk.stem import WordNetLemmatizer lemmatizer = WordNetLemmatizer() import json

import pickle

import numpy as np

from keras.models import Sequential

from keras.layers import Dense, Activation, Dropout from keras.optimizers import SGD

import random

words=[] classes = [] documents = []

ignore\_words = ['?', '!']

data\_file = open('intents.json').read() intents = json.loads(data\_file)

for intent in intents['intents']:

for pattern in intent['patterns']:

#tokenize each word

w = nltk.word\_tokenize(pattern) words.extend(w)

documents.append((w, intent['tag'])) # add to our classes list

if intent['tag'] not in classes: classes.append(intent['tag'])

# lemmaztize and lower each word and remove duplicates

words = [lemmatizer.lemmatize(w.lower()) for w in words if w not in ignore\_words]

words = sorted(list(set(words))) # sort classes

classes = sorted(list(set(classes)))

# documents = combination between patterns and intents print (len(documents), "documents")

# classes = intents

print (len(classes), "classes", classes) # words = all words, vocabulary

print (len(words), "unique lemmatized words", words)

pickle.dump(words,open('words.pkl','wb')) pickle.dump(classes,open('classes.pkl','wb'))

# training data training = []

# create an empty array for our output output\_empty = [0] \* len(classes)

for doc in documents: bag = []

pattern\_words = doc[0]

pattern\_words = [lemmatizer.lemmatize(word.lower()) for word in pattern\_words]

for w in words:

bag.append(1) if w in pattern\_words else bag.append(0)

output\_row = list(output\_empty) output\_row[classes.index(doc[1])] = 1

training.append([bag, output\_row]) random.shuffle(training)

training = np.array(training) # X - patterns, Y - intents train\_x = list(training[:,0]) train\_y = list(training[:,1]) print("Training data created")

# model

model = Sequential()

model.add(Dense(128, input\_shape=(len(train\_x[0]),), activation='relu'))

model.add(Dropout(0.5))

model.add(Dense(64, activation='relu'))# rectified linear unit activation function

model.add(Dropout(0.5)) model.add(Dense(len(train\_y[0]), activation='softmax'))

sgd = SGD(lr=0.01, decay=1e-6, momentum=0.9, nesterov=True)

model.compile(loss='categorical\_crossentropy', optimizer=sgd, metrics=['accuracy'])

#fitting and saving the model

hist = model.fit(np.array(train\_x), np.array(train\_y), epochs=200, batch\_size=5, verbose=1)

model.save('chatbot\_model.h5', hist)

print("model created")

#### Ch\_6.ipynb:

import nltk

from nltk.stem import WordNetLemmatizer lemmatizer = WordNetLemmatizer() import pickle

import numpy as np

from keras.models import load\_model model = load\_model('chatbot\_model.h5') import json

import random

intents = json.loads(open('intents.json').read()) words = pickle.load(open('words.pkl','rb')) classes = pickle.load(open('classes.pkl','rb'))

def clean\_up\_sentence(sentence):

sentence\_words = nltk.word\_tokenize(sentence) sentence\_words = [lemmatizer.lemmatize(word.lower())

for word in sentence\_words] return sentence\_words

def bow(sentence, words, show\_details=True): # tokenize the pattern

sentence\_words = clean\_up\_sentence(sentence) bag = [0]\*len(words)

for s in sentence\_words:

for i,w in enumerate(words):

if w == s:

bag[i] = 1

if show\_details:

print ("found in bag: %s" % w) return(np.array(bag))

def predict\_class(sentence, model):

p = bow(sentence, words,show\_details=False) res = model.predict(np.array([p]))[0]

ERROR\_THRESHOLD = 0.25

results = [[i,r] for i,r in enumerate(res) if r>ERROR\_THRESHOLD]

results.sort(key=lambda x: x[1], reverse=True) return\_list = []

for r in results:

return\_list.append({"intent": classes[r[0]], "probability": str(r[1])})

return return\_list

def getResponse(ints, intents\_json): tag = ints[0]['intent']

list\_of\_intents = intents\_json['intents'] for i in list\_of\_intents:

if(i['tag']== tag):

result = random.choice(i['responses']) break

return result

def chatbot\_response(msg):

ints = predict\_class(msg, model) res = getResponse(ints, intents) return res

#Creating GUI with tkinter import tkinter

from tkinter import \*

def send():

msg = EntryBox.get("1.0",'end-1c').strip() EntryBox.delete("0.0",END)

if msg != '':

ChatLog.config(state=NORMAL)

ChatLog.insert(END, "You: " + msg + '\n\n') ChatLog.config(foreground="#442265",

font=("Verdana", 12 ))

res = chatbot\_response(msg) ChatLog.insert(END, "Bot: " + res + '\n\n')

ChatLog.config(state=DISABLED) ChatLog.yview(END)

base = Tk() base.title("Hello") base.geometry("400x500")

base.resizable(width=FALSE, height=FALSE)

#Create Chat window

ChatLog = Text(base, bd=0, bg="white", height="8", width="50", font="Arial",)

ChatLog.config(state=DISABLED)

#Bind scrollbar to Chat window

scrollbar = Scrollbar(base, command=ChatLog.yview, cursor="heart")

ChatLog['yscrollcommand'] = scrollbar.set

#Create Button to send message

SendButton = Button(base, font=("Verdana",12,'bold'), text="Send", width="12", height=5,

bd=0, bg="#32de97", activebackground="#3c9d9b",fg='#ffffff',

command= send )

#Create the box to enter message

EntryBox = Text(base, bd=0, bg="white",width="29", height="5", font="Arial")

#Place all components on the screen scrollbar.place(x=376,y=6, height=386) ChatLog.place(x=6,y=6, height=386, width=370) EntryBox.place(x=128, y=401, height=90, width=265) SendButton.place(x=6, y=401, height=90)

base.mainloop()

International Journal of Research Publication and Reviews, Vol 3, no 4, pp 1655-1659, April 2022

#### International Journal of Research Publication and Reviews

Journal homepage: [www.ijrpr.com](http://www.ijrpr.com/) ISSN 2582-7421

1661

#### Web based E-commerce system integrated with Chatbot

***M. Shyam Manikanta a, J. Rushi b, A. Lalitha c, B. Shravan Kumar Goud d, V. Suresh e, T. Daniya f***

*a Student, b Student, c Student, d Student, e Student, f Asst. Professor Department of Information Technology, GMR Institute of Technology, Rajam – 532127, Andhra Pradesh, India.*

*DOI:*[*https://doi.org/10.55248/gengpi.2022.3.4.12*](https://doi.org/10.55248/gengpi.2022.3.4.12)

A B S T R A C T

E-commerce has the potential to be a huge success in today's commercial world. E-Commerce platform is used for purchasing of the human required goods and selling of items through it as well as the payments of the purchased goods made through online mode over the different payment platforms over the internet. E-commerce may represent a paradigm change that affects both marketers and customers. Rather, e-commerce is merely a new way to spice up existing business methods. It's causing a complete shift in the traditional company model. A web application, which is a piece of software that runs on a website. It is a computer-based software programme that is saved locally on the system's operating system. It is used by users or administrators over the Internet. It can be used to create an application for a variety of platforms. However, we believe that using Python is the safest option and can be employed in a variety of situations. It also plays a significant role in our web apps. Hackers won't be able to use SQL Injection as rapidly if the web application is written in Python. This project is intended to create a chatbot to be used by customers to get their queries responded easily from the e-commerce website. This chatbot has the capacity to make friendly conversations and respond to the queries. Moreover, it provides information with product details, payment method and many more.

Keywords: E-commerce, Python, Chatbot etc.

1. **Introduction**

E-commerce is the practice of conducting business over the internet. All transactions and deals are conducted over the internet. An ecommerce site is an online store similar to a physical store where customers come in, look at the goods, make a purchase, and then leave. However, an ecommerce website design gives owners access to a back-end area where they can go in and check on other orders as well as manage products. The primary goal of our project is to develop a general-purpose e-commerce web application in which the products (men's wear, women's wear, and children's wear) are displayed, can be purchased through our website via the Internet from the comfort of one’s different places. However, for the sake of execution, this project will concentrate on an online clothing business. An online store is a virtual store on the Internet where clients may explore and select things of interest. To collect the desired things, a shopping cart can be utilized. When you go to check out, the items in your shopping basket will be shown as an order.

Chatbots, commonly referred to as chatterbots, are artificial intelligence (AI) systems used in messaging apps. Clients will be benefited from this product because it is an automated program that continuously interacts with customers like a human would and costs very less to use when compared to customer care services. Chatbots will be having a friendly discussion and responding to quickly to customer questions.

\* *Corresponding author.*

E-mail address: [18341A1227@gmrit.edu.in](mailto:18341A1227@gmrit.edu.in)

International Journal of Research Publication and Reviews Vol ( ) Issue ( ) (2021) Page 000

1662

Payment gateways provide e-commerce web application service providers with a way for online merchants, brick-and-click retailers, and conventional brick-and-mortar retailers to accept various payment methods including credit cards and direct payments. A bank may offer a payment gateway to its users, but it can also be offered separately by a different financial service provider, such as a payment service provider. The transfer of data between a payment portal like website, cell phone, or interactive voice response service and the particular bank is made possible via a payment gateway.

Merchants can accept a wide variety of payment methods by using payment gateways on their websites, applications, and storefronts. The payment gateway is a web server to which a merchant's web-application or mobile application is linked, but it is not directly engaged in the money transfer. A payment gateway is a system that connects a variety of institutions, clients, and payment methods. Various institutions, clients, and payment methods can be connected by a payment gateway.

1. **Literature Review**
   1. **Web Development using Flask and Python:**

Web is the most widely and quickly utilized networking tool, meeting the needs of all types of users and providing solutions to any problem. We must select the appropriate technology in order to create and develop such well-defined and well-structured systems. As a result, flask and python can be used to create a dynamic web application or portal. A well-designed web page or application can readily attract users, resulting in the project's success. Using "python" and "flask," you can meet the technological requirements of a web development project. Flask has the technical advantages like Integrated support for unit testing, secure cookies, extensive documentation, Restful request dispatching etc. Python and Flask are web development languages that are both portable and interactive, with the ability to support dynamic semantics. It is also feasible to connect new modules to python in order to extend the language's fundamental features. Because of Python's robustness, many great websites are migrating to it. This paper concludes that, in contrast to other languages used for web development, Python is faster, more versatile, and more powerful than java, perl, and tcl, and so on..

* 1. **E-commerce websites development:**

In the present developing world, the growing number of people use e-commerce websites for making their everyday purchases. The article outlines numerous ways to build e-commerce websites as well as best practices and solutions to development problems. The entire development process of e-commerce is divided into two parts. Development forms include front-end development and back-end development. There are several tabs visible on the front-end, including the home page, administration page, contact page, and cart page etc… which comes under the UI designing. The Back-end part comprises of the coding parts of front-end pages and the database connection and its interaction with the front-end. This paper shows the developing of the e-commerce with standard languages such as: HTML, CSS, JavaScript, BootStrap for the front-end part and PHP for the back-end part and MySql for its database connection.

* 1. **The E-commerce platform:**

People now a days are completely dependent on internet. Web based will always give a upper hand where client base is increased, as we can see individuals who are interstate or abroad. Having a web-based store allows you to operate without the need for a physical location, and allows your customers the option to buy your product whenever they want to. Regardless of the e-commerce site you use, you will be able to move your products online. Online shopping can be used to send and receive large quantities of data, items, and installments between customers, website owners, and providers. Online platforms will offer item depiction, pictures, cost, and product reviews. A typical application will be split into two parts: front-end development and back-end development. During front-end development, the site's login page, home page, contact page, admin page, shopping cart page, and payment page are developed visually.

By default, the back end will manage the database and all interactions with the front end. For front end developing we use JavaScript. It is a client-side programming language designed specifically for web development. It was simply a matter of combining JavaScript code with HTML5 code. The markup language used for web pages in applications is called hypertext markup language. A client-side scripting language typically used in web browsers, JavaScript allows the user to interact with client-side scripts.

* 1. **JAICOB Chatbot with Data Science:**

Chat bots execute automated tasks using software. The chat bot is designed to converse with humans through written or audio means. Using a chat bot, you can interact as if you were talking with a human, answering questions and interacting further. CHATBOT is the word where we are hearing everywhere over the internet. People are getting habituated to this feature. Nowadays, every online application has the capacity to interact with clients and answer questions and requests. Chat bot feature usage has been increased rapidly because whenever a user interacts the level of satisfaction is on higher side and it facilitates finding the necessary information in a more comfortable way. These

are fantastic analytical tools for everyone, because end users are more likely to submit messages to chat bots than to real humans. Chat bots produce favourable user responses such as engagement, active learning, and sociability when compared to e-learning training.

* 1. **Xatkit Chatbot with less code:**

Chatbots, such as those used for customer service, education, and e-commerce, enable companies and end users to communicate directly, by executing certain activities and improving the user experience. Most of them today have built-in capabilities for chatbots. In order to meet quality criteria, chatbots are becoming more complicated software artifacts that require a more systematic development strategy. We have introduced XatKit chatbot developed using Model Driven Engineering technique. XatKit performs rich conversation flow between user and a bot. XatKit is going to used by end users in real life scenarios. It uses the reusability of the code. It could also use NLP engine for text analysing phase. XatKit improves by moving towards a product line approach and can be quickly update several versions of the same bot. Xatkit is a ready-to-use, complete tool that can be utilized in real-world settings. The bot specification will be increasingly diverse at the language level, moving towards a product-line approach to allow businesses to design and rapidly update several variants of the same bot at the same time.

* 1. **Merits and Demerits of Chatbot:**

Over the last few years, e-learning has gained in popularity with the introduction of new technology. people can expand their knowledge anytime and anywhere by using their mobiles. Learning and studying can take place wherever and whenever the student chooses. In e- learning platforms, students often experience feelings of loneliness and isolation due to the lack of interaction between professors and students. Throughout today's modern education, e-learning is becoming increasingly popular. Since younger students may be more familiar with these new forms of technology and are more likely to be interested in new ways of learning, many higher education institutions, including universities, are increasingly using E-Learning platforms. Chatbots can be used to engage in chats or respond to questions any time of the day or night, while real-life teachers help solve issues with learning. By increasing the bot's intelligence, additional learning- related issues can be understood and solved. It creates an interactive learning environment that is similar to a typical classroom with all of the benefits of E-learning, as well as a more human-like engagement because it is based on voice dialogue.

* 1. **Chatbots with Dailogue Management:**

Chat bot is like a question answer pair between user and the computer system. The disclosure tree acts as an interface. For example if user gives a dailogue to a system, the responses should aligned on the left and other responses on the right which are system responses. We have developed a chatbot for complex queries are best suitable for user interactive. The conversation between user and system should not be just like a conversation , it should be like clarification request and perfect, suitable clarification response. Chatbots used earlier are responsive to user. So we use DT for building the chatbot effectively. The chatbot examples are siri- iphone and Cortana-Windows.

* 1. **Chatbots with Response Generation:**

Chatbot is based on question and answer interaction between human being and a robot. In this paper, we first find a generated response from generation based polisher and then GR is passed into polished response filter. According to the evaluate relevance score, the robot id going to gives the output. The output generated is based on score. If the score is high then the output is achieved to that node. Retrieval polished response generation for chatbot is an advanced technology which gives efficient, fluent answers. The first model used is retrieval based which is written by humans. The main disadvantage is irrelevant context. Retrieval based is similar to computational contextual. The second model is generation based which is language rules learned learning during the training process. The main disadvantage is safe responses and grammatical errors. The combined two models gives efficient question and answer pair fluent and no grammatical errors.

1. **Related Work**
   1. **Python:**

A high-level programming language such as Python puts a strong emphasis on code readability. In the web development, the source lines of code (SLOC) is less in python when compared to the traditional programming languages. Python has extensive standard libraries, which make the web development code straight forward, simple and easy to understand. Python is used on a various systems like Windows, Mac, Linux, etc. Python has a simple syntax which is analogous like English. Python is categorized in three ways: object-oriented, functional, and procedural. Python is an interpreter language, which means that the lines of code runs fast, as it is written. As a result of it, the prototyping will be done very fastly as compared to other languages.

* 1. **Flask:**

A Python Application Programming Interface, Flask, allows users to create web-based applications, it is developed by Armin Ronacher. Flask's framework is easier to understand and develop a simple web application than Django's. Web Frameworks, often referred to as Web- Application Frameworks, are a collection of modules and libraries that enable programmers to create applications without writing protocol or thread code. A free and open source web application framework, Flask is popular among web developers. This tells us that flask is supplied with all of the tools, frameworks, and the technologies which helps the users to build a web application. It can be used in conjunction with web pages, a blog, a wiki, a web-based calendar application, or a commercial website. PyFlask is a Python-based framework for building web applications. Flask uses Werkzeug WSGI toolKit and Jinja2 template engine to develop an application. Flask is one of the micro framework's Categories. Micro-frameworks are often frameworks that depend on few or no additional libraries. Flask is divided into categories like "Static files" and "Template files. "All Jinja templates, including the HTML pages, are stored in the template files. Static files, on the other hand, include all the static scripts required for a website, such as CSS, JavaScript, and image files etc.

* 1. **E-commerce:**

E-commerce, it is frequently used to simply refer to the buying and selling of the products or services over the internet, and the movement of funds and data necessary to perform the required transactions. online commerce is the another name for Electronic commerce. As a business approach, e-commerce is rapidly growing in popularity and acceptance. A majority of businesses have websites that enable them to conduct commercial transactions via various payment gateways via the internet today. The number of people purchasing items online is definitely increasing in recent years. "Online stores" refer to virtual stores that exist on the Internet and allow users to explore chosen goods of interest. Using a shopping cart, a customer can collect goods that interest him/her. As soon as the user makes a payment, the items in the shopping cart will be presented to him or her as an order. Additional statistics should be provided by the user in the website like Address, Mobile number etc. to finish the purchase. The customer is asked to provide the details like name, house number, street name, city, state and also the verified e-mail id. The very next moment of the product is placed, the customer gets an order placement notification to their e-mail. There are four market segments in which e-commerce operates: business-to-business, business-to-customer, customer-to-customer, and customer-to-business. The way customers shop, purchase the products and consume the services has evolved as a result of e-commerce. In the present scenario, the people are increasingly using their systems and smart devices to place an order for the required goods which can be delivered immediately to their required places. As a result, the retail sellers has been effected. Amazon and Alibaba has risen in popularity. Traditional retailers have indeed been forced to change their ways of doing business as a result of their growing popularity.

* 1. **Chatbot:**

According to the Natural language processing (NLP), the Chatbot is classified into the four parts. The front-end is the primary part, then it followed by the knowledge base, the back-end part, and finally the corpus, which will be containing of the training data. On the front end, the customer is communicated. Natural language understanding refers to the process of interpreting context and purpose of user input. The user will be responded with a suitable response. The knowledge base is used to determine the Chatbot’s knowledge, that is done with using the NLU and supported on the back-end. The Corpus domain is used by the back-end to create the knowledge base. The user provides the input to the Chatbot in the mode of Text or Speech. By entering the data into a conversation management system for processing, a suitable response is generated, and the Chatbots are instructed to take appropriate actions based on the input. Both the text and speech used to create and provide the responses.

The Chatbot is constructed with the help of Artificial Intelligence and Machine Learning (AIML), that has the raw analytics data. In order to get the necessary data, the analytics tool uses raw data. Any website or application keeps track of all the statistics that clients submit. AIML is a set of queries and responses that can be used by the chatbots. Three sections make up the report: a template, a collection of categories, and a pattern. Patterns and templates are included in every area. Chatbot-Customers may enter a variety of patterns when requesting products and services. The Template is the answer to each pattern.

1. **Proposed Methodology**
   1. **Overall Architecture:**

We have developed the web application in such a way that the Customer can visit the website and view the products. The Customer needs to be authenticated andneed to verify his/her email then he will be able to Buy the required products. The user can use the Chatbot provided in the website and can clarify his/her doubts regarding the website. After the selection of products, the customer should make the payment of the products he/her wanted to purchase. Admin have the access to make changes in the website and take necessary actions. Generally many websites have customer care services available for users to clarify their doubts. But the Major drawback of the traditional customer care services is, they are not available 24/7. To overcome this, we have integrated a chatbot in this website which is developed using the Natural Language Processing (NLP). These Chatbots are available 24/7 and can be responded quickly to user queries. The main advantage

of using the chatbots are, the usage of man power is reduced and the working expenditure will be saved. The problems arised in the human communications in the customer care services can also be controlled using these chatbots. The Chatbots are very highly programmed and can be responded quickly. After the selection of the products required for users, the users should add the products to the cart. Finally the orders are placed by making the payment. Our website was developed by the python programming language which was a trending technology. This technology provides good accuracy and security to the developed website. And we have integrated a chatbot in the website, where the traditional e-commerce websites does not contain it. Traditional e-commerce website contains only customer care services, that does not available every time to the customer. This problem can be reduced by using the chatbots. This Chatbots are trained by Admin with the required datasets which provides the answers to the customer queries. We should train the chatbots using the Natural Language Processing (NLP) with the training data set and tested using the test data set.

CUSTOMER – The customer can login to the website and view the products, select the products and Make the payment. ADMIN – The Admin has all the permissions to make changes in the website and perform the necessary tasks.

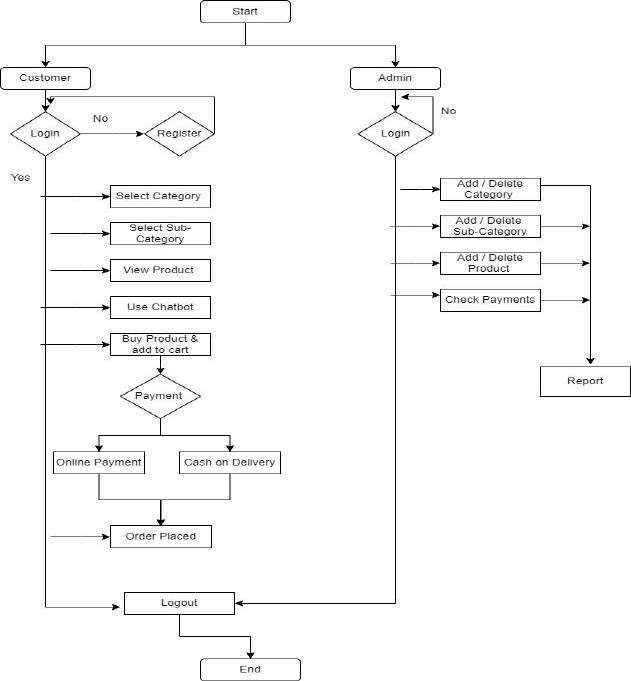


Fig 4.1: Flowchart of the proposed system

* 1. **Proposed System:**

Our system consists of two parts frontend and backend. The whole frontend is designed in HTML, CSS, BootStrap and JavaScript using the help of the available Widgets and Material UI. The backend is designed in the Python language using the flask. The Postgresql is used for the database connection in the back end. As compared to traditional backend languages like PHP, Java Script, etc. Python is very advanced and powerful. We have used the python language for the backend development and the flask frameworks in the python. The Python is very advanced language and can be easily programmed . The security of the python is high compared to the traditional languages. The Postgresql is used for the database connection in this website development. Traditional languages used the MySQL database connection which are less secured. The MySQL database can be cracked by SQL injections. This problem is overcommed in the postgresql. The Flask is a light weighted web application framework in the Python. The Python is a advanced language using for the programming. The Python

language has less source line of code compared to the traditional languages, this makes the backend programming of the website easier and can be developed in the less time. Using the python programming language provides more security than the other traditional languages. The following are the screenshots of the working model of the application in frontend.

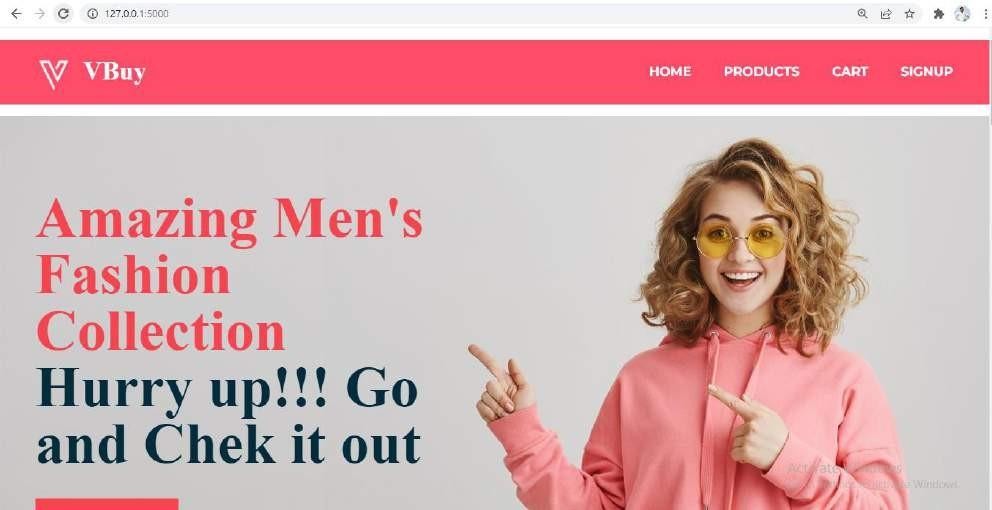


Fig 4.2.1: Home Page of proposed website

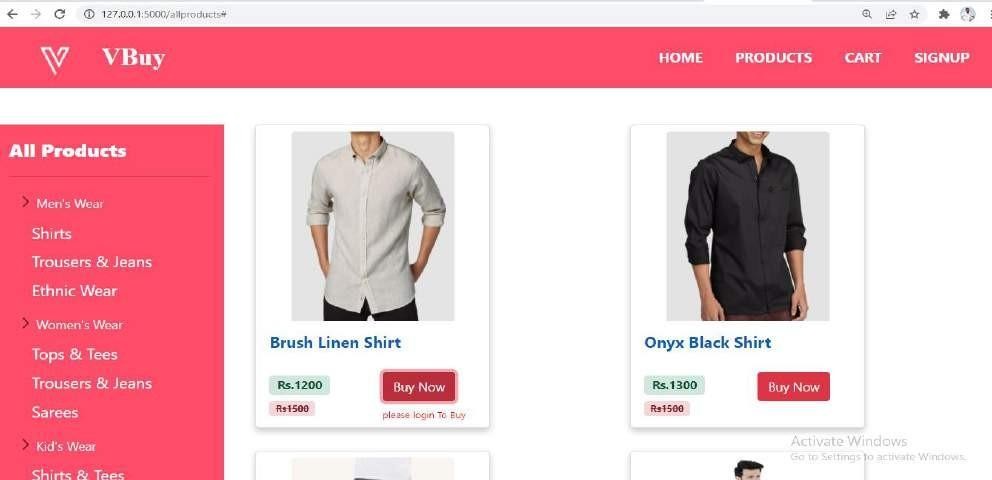


Fig 4.2.2: Products page of proposed website

1. **Conclusion**

The main aim of our project is to create an e-commerce website with the python language using the flask web framework. The flask is the new language used for web development which provides more secured cookies and with the lesser code compared to the traditional

languages. This website consists of chatbot, where users can use it as service. The chatbots gives quicker responses than the customer care service and will be available 24/7. The customers can buy the products by selecting them and making the payment in the online mode or cash on delivery method.

**REFERENCES**

1. Vangala Rama Vyshnavi, Amit Malik , “Efficient Way of Web Development Using Python and Flask” , International Journal of Recent Research Aspects Vol. 6, Issue 2, June 2019.
2. S. E. Ullah, T. Alauddin and H. U. Zaman, "Developing an E-commerce website," 2016 International Conference on Microelectronics, Computing and Communications (MicroCom), 2016.
3. Mr. Pankaj Gaikwad, Ms. Aditi Ghavate, “E-commerce Platform Using Website Development,” Volume 8, Issue 4, 2020.
4. D. Carlander Reuterfelt, A. Carrera, C. A. Iglesias, O. Araque, J. F. Sánchez Rada and S. Muñoz, "JAICOB: A Data Science Chatbot," in IEEE Access, vol. 8, pp., 2020.
5. G. Daniel, J. Cabot, L. Deruelle and M. Derras, "Xatkit: A Multimodal Low-Code Chatbot Development Framework," in IEEE Access, vol. 8, pp., 2020.
6. E. H. Wu, C. Lin, Y. Ou, C. Liu, W. Wang and C. Chao, "Advantages and Constraints of a Hybrid Model K-12 E-Learning Assistant Chatbot," in IEEE Access, vol. 8, pp., 2020.
7. Boris Galitsky, Dmitry Ilvovsky "Chatbot with a Discourse Structure-Driven Dialogue Management", Association for Computational Linguistics, 2017.
8. L. Zhang, Y. Yang, J. Zhou, C. Chen and L. He, "Retrieval-Polished Response Generation for Chatbot," in IEEE Access, vol. 8, pp., 2020.
9. J. Ni, X. Zhao and L. Zhu, "A Semantic Web Service-Oriented Model for E-Commerce," 2007 International Conference on Service Systems and Service Management, 2007.
10. Lei Cui, Shaohan Huang, Furu Wei, Chuanqi Tan, Chaoqun Duan, and Ming Zhou "SuperAgent: A Customer Service Chatbot for E-commerce Websites", Association for Computational Linguistics, 2017.



**ISSN 2582-7421**

## International Journal of Research Publication and Reviews

#### (Open Access, Peer Reviewed, International Journal)

***Sr. No: IJRPR 8026***

# Certiﬁcate of Acceptance & Publication

### This certificate is awarded to M. Shyam Manikanta, and certifies the acceptance for publication of research paper entitled “Web based E-commerce system integrated with Chatbot” in “International Journal of Research Publication and Reviews”, Volume 3, Issue 4, 2022.

***Signed***  ***Date***

##### Editor-in-Chief

**International Journal of Research Publication and Reviews**

***21/04/2022***



**ISSN 2582-7421**

International Journal of Research Publication and Reviews

#### (Open Access, Peer Reviewed, International Journal)

***Sr. No: IJRPR 8027***

# Certiﬁcate of Acceptance & Publication

### This certificate is awarded to J. Rushi, and certifies the acceptance for publication of research paper entitled “Web based E-commerce system integrated with Chatbot” in “International Journal of Research Publication and Reviews”, Volume 3, Issue 4, 2022.

***Signed***  ***Date***

##### Editor-in-Chief

**International Journal of Research Publication and Reviews**

***21/04/2022***



**ISSN 2582-7421**

International Journal of Research Publication and Reviews

#### (Open Access, Peer Reviewed, International Journal)

***Sr. No: IJRPR 8028***

# Certiﬁcate of Acceptance & Publication

### This certificate is awarded to A. Lalitha, and certifies the acceptance for publication of research paper entitled “Web based E-commerce system integrated with Chatbot” in “International Journal of Research Publication and Reviews”, Volume 3, Issue 4, 2022.

***Signed***  ***Date***

##### Editor-in-Chief

**International Journal of Research Publication and Reviews**

***21/04/2022***



**ISSN 2582-7421**

International Journal of Research Publication and Reviews

#### (Open Access, Peer Reviewed, International Journal)

***Sr. No: IJRPR 8029***

# Certiﬁcate of Acceptance & Publication

### This certificate is awarded to B. Shravan Kumar Goud, and certifies the acceptance for publication of research paper entitled “Web based E-commerce system integrated with Chatbot” in “International Journal of Research Publication and Reviews”, Volume 3, Issue 4, 2022.

***Signed***  ***Date***

##### Editor-in-Chief

**International Journal of Research Publication and Reviews**

***21/04/2022***



**ISSN 2582-7421**

International Journal of Research Publication and Reviews

#### (Open Access, Peer Reviewed, International Journal)

***Sr. No: IJRPR 8030***

# Certiﬁcate of Acceptance & Publication

### This certificate is awarded to V. Suresh , and certifies the acceptance for publication of research paper entitled “Web based E-commerce system integrated with Chatbot” in “International Journal of Research Publication and Reviews”, Volume 3, Issue 4, 2022.

***Signed***  ***Date***

##### Editor-in-Chief

**International Journal of Research Publication and Reviews**

***21/04/2022***