

**A
PROJECT
ON
“IVA”
(INTELLECTUAL VIRTUAL ASSISTANT)**

Submitted to

**G. S. COLLEGE OF COMMERCE & ECONOMICS, NAGPUR
(AUTONOMOUS)
In the Partial Fulfillment of**

B.Com. (Computer Application) Final Year

**Submitted by
Mahima Dwivedi
Manalee Gedam**

**Under the Guidance of
Pravin J. Yadao**



**G. S. COLLEGE OF COMMERCE & ECONOMICS, NAGPUR
(AUTONOMOUS)
2021-2022**

**G. S. COLLEGE OF COMMERCE & ECONOMICS,
NAGPUR
(AUTONOMOUS)**

CERTIFICATE

(2021 - 2022)

This is to certify that Mr. Miss Mahima Dwivedi and Manalee Gedam have completed their project on the topic of IVA (INTELLECTUAL VIRTUAL ASSISTANT) prescribed by G. S. College of Commerce & Economics, Nagpur (Autonomous) for B.Com. (Computer Application)

Date:

Place: Nagpur

Pravin J. Yadao

Project Guide

External Examiner

Internal Examiner

ACKNOWLEDGEMENT

We take this opportunity to express our deep gratitude and whole hearted thanks to project guide Prof. Pravin Yadao, Coordinator for his guidance throughout this work. We are very much thankful to him for his constant encouragement, support and kindness.

We are also grateful to our teachers Prof. Rahul Tiwari, Prof. Sushma Gawande, Prof. Preeti Rangari, Prof. Prajkta Deshpande and Prof. Haresh Naringe for their encouragement, help and support from time to time.

We also wish to express our sincere thanks to Principal Dr. N. Y. Khandait for providing us wide range of opportunities, facilities and inspiration to gather professional knowledge and material without which this project could not have been completed.

Mahima Dwivedi Manalee Gedam

Date:

Place: Nagpur

DECLARATION

We (**Mahima Dwivedi and Manalee Gedam**) hereby honestly declare that the work entitled “**IVA (INTELLECTUAL VIRTUAL ASSISTANT)**” submitted by us at G.S. College of Commerce & Economics, Nagpur (Autonomous) in partial fulfillment of requirement for the award of B.Com. (Computer Application) degree by Rashtrasant Tukadoji Maharaj, Nagpur University, and Nagpur has not been submitted elsewhere for the award of any degree, during the academic session 2021-2022.

The project has been developed and completed by us independently under the Supervision of the subject teacher and project guide.

Mahima Dwivedi Manalee Gedam

Date:

Place: Nagpur

INDEX

Sr. No.	Project Report format	Page No.
1.	Introduction	6-9
2.	Objectives	10-14
3.	Preliminary system Analysis: 3.1: Preliminary Investigation 3.2: Present system in use 3.3: Flaws in present system 3.4: Need of new system 3.5: Feasibility study 3.6: Project category	15-36
4.	Software and hardware requirements specifications	37-39
5.	Detailed system analysis: 5.1: Data flow diagram 5.2: Number of modules and process logic 5.3: Data structure and Tables 5.4: Entity Relationship Diagram	40-44
6.	System Design 6.1: Form Design 6.2: Source code 6.3: Input and output screen	45-91
7.	Testing and validation check	92-97
8.	System security Measures	98-100
9.	Implementation, evolutions and maintenance	101-107
10.	Future scope of the project	108-110
11.	Conclusion	111-112
12.	Bibliography and Reference	113-114

INTRODUCTION

INTRODUCTION:

My project Title is “IVA : Intellectual Virtual Assistant” this project is based on helping people who need quick and hands free assistance or seek some form of help from available online resources or need to access basic IS embedded applications or need to send emails, create To-do lists, etc.

Today the development of artificial intelligence (AI) systems that can organize a natural human-machine interaction (through voice, communication, gestures, facial expressions, etc.) are gaining in popularity. One of the most studied and popular was the direction of interaction, based on the understanding of the machine by the machine of the natural human language. It is no longer a human who learns to communicate with a machine, but a machine learns to communicate with a human, exploring his actions, habits, behavior and trying to become his personalized assistant.

Virtual assistants are software programs that help you ease your day to day tasks, such as showing weather reports, creating reminders, making shopping lists etc. They can take commands via voice and using microphones. Voice-based intelligent assistants need an invoking word or wake word to activate the listener, followed by the command. We have so many virtual assistants, such as Apple’s Siri, Amazon’s Alexa and Microsoft’s Cortana.

This system is designed to be used efficiently on desktops. Personal assistants software improves user productivity by managing routine tasks of the user and by providing information from an online source to the user.

This project was started on the premise that there is a sufficient amount of openly available data and information on the web that can be utilized to build a virtual assistant that has access to making intelligent decisions for routine user activities.

IVA is a software agent that can perform tasks or services for an individual based on commands or questions. It is able to interpret human speech and respond via synthesized voices. Users can ask their assistants questions, control home automation devices and media playback via voice, and manage other basic tasks such as email, to-do lists, and calendars with verbal commands.

As of 2017, the capabilities and usage of virtual assistants are expanding rapidly, with new products entering the market and a strong emphasis on both email and voice user interfaces.

Virtual assistants use natural language processing (NLP) to match user text or voice input to executable commands. Many continually learn using artificial intelligence techniques including machine learning.

We also need feedback of users that may like our website or not it is very important to know how many users like our website that will help us to work on such flaws in future. Generally maximum website having a feedback form with details and information but here he provides different feedback technique with unique idea.

As this is all about our desktop application. It is very simple and easy but with latest updates and method what in regards to what Users want now a days. We definitely work on such flaws recommend by our users.

The easiest way to operate this project is to open the application and then to sign – in from the system and enter valid information. After that, user must log-in to the system and then the virtual assistant will automatically appear on the screen.

Once the assistant is opened, then the user just needs to talk to the assistant and ask their questions or request from the assistant to get their answers in a convenient way.

OBJECTIVES

OBJECTIVES:

- **Natural Language Processing:** Natural Language Processing or NLP refers to the branch of AI that provides the machines the power to read, understand and derive meaning from human languages. It can extract information from the documents and categorize them. Most of the NLP techniques rely on machine learning techniques to derive the corresponding meaning from human language. This app provides natural language processing.

- **Using API keys to interact with other Web apps:** API keys are security tokens that you use to access Conversational Cloud services. The API key gets installed automatically and assumes the security settings granted to the associated user in Conversational Cloud. API keys provide project authorization to decide which scheme is most appropriate, it's important to understand what API keys and authentication can provide.

API keys provide:

- **Project identification** — identify the application or the project that's making a call to this API.
- **Project authorization** — Check whether the calling application has been granted access to call the API and has enabled the API in their project

- **Use of Speech Recognition Module:** Speech recognition, also known as automatic speech recognition (ASR), computer speech recognition, or speech-to-text, is a capability which enables a program to process human speech into a written format. While it's commonly confused with voice recognition, speech recognition focuses on the translation of speech from a verbal format to a text one. Many speech recognition applications and devices are available, but the more advanced solutions use AI and machine learning. They integrate grammar, syntax, structure, and composition of audio and voice signals to understand and process human speech. Ideally, they learn as they go — evolving responses with each interaction.

This application makes use of this module for processing.

- **User Friendly:** Application is user friendly as any user can operate easily through laptop and language is used which is understandable for any common people. It also has a very easy UI that can be easily used and understood by users.
- **Use of Semantic Search and Sentence Transformers:** Semantic search is the task that allows us to find similarities between texts by their meaning instead of using just keywords. This application uses semantic search and recognizes keywords from sentence spoken by user to provide relevant answers to user query.

- **Text-to-speech conversion by VIA To Provide Answers:** Text-to-Speech (TTS) refers to the ability of computers to read text aloud. A TTS Engine converts written text to a phonemic representation, then converts the phonemic representation to waveforms that can be output as sound. TTS engines with different languages, dialects and specialized vocabularies are available through third-party publishers.

VIA uses TTS module to provide output and read aloud certain lines or interact with the user as well.

- **Using Semantic Data Resources Available:** Main objective of building personal assistant software (a virtual assistant) is using semantic data sources available on the web, user generated content and providing knowledge from knowledge databases.

This project does that through using API keys to interact with open source resources available online and use their databases to provide required information.

- **Providing Answers To Users Query:** The main purpose of an intelligent virtual assistant is to answer questions that users may have. This may be done in a business environment as well as a personal environment. It's done in this application through the use of TTS module, Speech Recognition Module, APL Keys, etc.

- **Facilitate User Time Optimization:** One of the main objectives of the application is to enable the users to achieve optimum time optimization through the ability to multitask, get answers quicker without manual searching and to get quicker and updated answers.
- **To Help Users with Organization of Daily Tasks:** This application can help users organize their time by helping those who use it, to set alarms and reminders, create notes and lists, etc.

PRELIMINARY **SYSTEM ANALYSIS**

PRELIMINARY SYSTEM ANALYSIS:

Preliminary Investigation basically refers to the collection of information that guides the management of an organization to evaluate the merits and demerits of the project request and make an informed judgment about the feasibility of the proposed system.

The preliminary investigation occurs within a short period ranging from a few hours to a few days and should not exceed two to three days. The purpose of the preliminary investigation is to determine whether the problem or deficiency in the current system really exists. The project team may re-examine some of the feasibility aspects of the project.

1) Preliminary Investigation:

Preliminary Investigation basically refers to the collection of information that guides the management of an organization to evaluate the merits and demerits of the project request and make an informed judgment about the feasibility of the proposed system. This sort of investigation provides us with a through picture of the kind of software and hardware requirements which are most feasible for the system, plus the environment in which the entire project has to be installed and made operational.

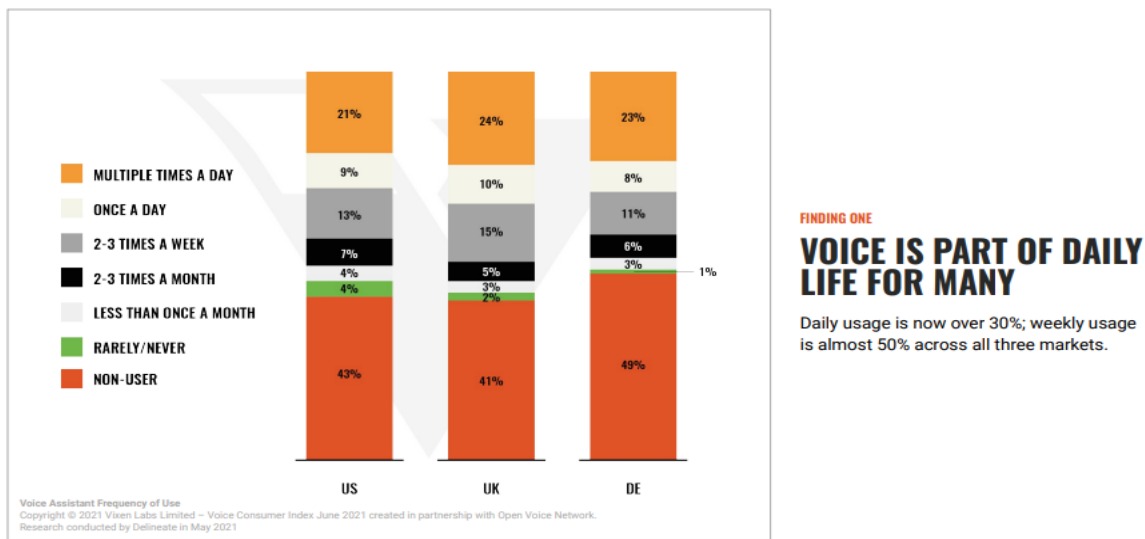
The preliminary-investigation phase sets the stage for gathering information about the current problem and the existing information system. This information is then used in studying the feasibility of possible information systems solutions. Intelligent virtual assistants, or IVA, can be used for a wide range of activities across different departments. For example, sales development teams can use them to help collect information about leads,

qualify leads, and automate booking meetings with new prospects.

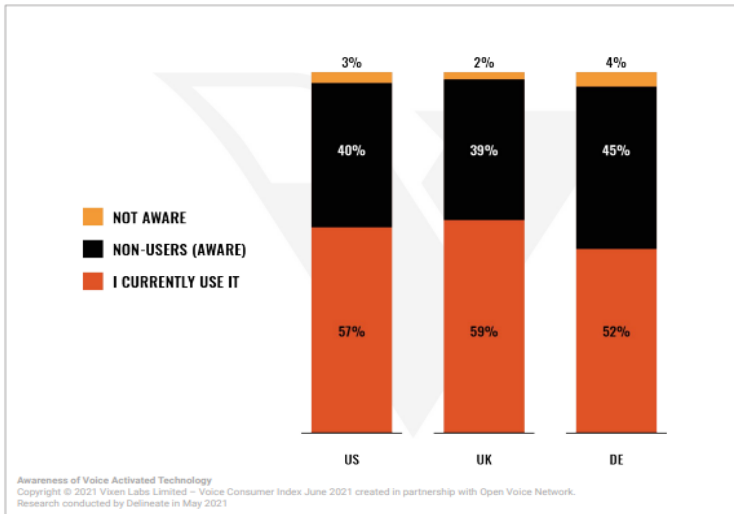
These tools can also be used by marketing and customer support teams to help answer questions from customers and web visitors, share content or links to other resources, and provide teams with a unified inbox for responding to requests. Within an organization, a virtual assistant can improve efficiency and offer support to both employees and customers of the organization. It allows them to offer more services as digital assistants can take over the more routine tasks. Because digital assistants do the more routine tasks, employees can spend more time on other tasks. Not only allowing them to offer more and better services, but it also allows the organization to save money. Which is why development of a virtual assistant can be classified as important and feasible.

This conclusion is based on information derived from the voice consumer index report by Vixen Labs in 2021.

The survey results are displayed in form of charts derived from data across multiple countries:



1.1: Chart stating the normal daily use observed by users of Virtual Assistants



FOR MORE DETAIL, DOWNLOAD THE WHITE PAPER AT VIXENLABS.CO

FINDING TWO

THERE IS SIGNIFICANT AWARENESS AND USE ACROSS ALL THREE REGIONS

Most people use at least one Voice assistant.

1.2: Global user awareness

	18-24	25-34	35-44	45-54	55-64	65+
ALEXA (AMAZON)	19%	29%	39%	34%	40%	41%
SIRI (APPLE)	60%	36%	19%	31%	28%	28%
GOOGLE ASSISTANT	18%	30%	35%	31%	29%	23%
CORTANA (MICROSOFT)	2%	1%	3%	2%	2%	6%
BIXBY (SAMSUNG)	2%	4%	3%	2%	1%	2%

Main Voice Assistant Used by Age Group and Country
Copyright © 2021 Vixen Labs Limited – Voice Consumer Index June 2021 created in partnership with Open Voice Network.
Research conducted by Delineate in May 2021

FOR MORE DETAIL, DOWNLOAD THE WHITE PAPER AT VIXENLABS.CO

US

FINDING THREE

VOICE ASSISTANT USAGE IS SPREAD ACROSS ALL AGE GROUPS

The opportunity is not just for brands with younger audiences.

UK

	18-24	25-34	35-44	45-54	55-64	65+
ALEXA (AMAZON)	32%	50%	59%	61%	71%	63%
SIRI (APPLE)	42%	24%	14%	12%	13%	15%
GOOGLE ASSISTANT	19%	20%	23%	25%	14%	21%
CORTANA (MICROSOFT)	3%	2%	2%	1%	2%	2%
BIXBY (SAMSUNG)	3%	4%	2%	1%	0%	0%

Main Voice Assistant Used by Age Group and Country
Copyright © 2021 Vixen Labs Limited – Voice Consumer Index June 2021 created in partnership with Open Voice Network.
Research conducted by Delineate in May 2021

FOR MORE DETAIL, DOWNLOAD THE WHITE PAPER AT VIXENLABS.CO

GERMANY

	18-24	25-34	35-44	45-54	55-64	65+
ALEXA (AMAZON)	25%	35%	54%	48%	55%	46%
SIRI (APPLE)	49%	28%	14%	13%	15%	17%
GOOGLE ASSISTANT	17%	23%	25%	34%	23%	34%
CORTANA (MICROSOFT)	4%	5%	4%	2%	7%	2%
BIXBY (SAMSUNG)	5%	8%	4%	2%	1%	1%

Main Voice Assistant Used by Age Group and Country
Copyright © 2021 Vixen Labs Limited – Voice Consumer Index June 2021 created in partnership with Open Voice Network.
Research conducted by Delineate in May 2021

FOR MORE DETAIL, DOWNLOAD THE WHITE PAPER AT VIXENLABS.CO

1.3: Usage divided as per age groups

	US	UK	GERMANY
I AM CONCERNED ABOUT THE PRIVACY OF MY DATA WHEN USING VOICE	52%	56%	41%
I'M INTERESTED IN HOW VOICE TECHNOLOGY WILL DEVELOP IN THE FUTURE	51%	52%	50%
USING A VOICE ASSISTANT TO SEARCH IS QUICKER THAN TEXT	46%	46%	46%
VOICE ASSISTANTS HELP ME TO SEARCH EFFECTIVELY FOR INFORMATION	41%	44%	42%
EVERYONE WILL BE USING VOICE ASSISTANTS IN THE FUTURE	41%	45%	35%
I WANT TO KNOW WHAT ELSE I CAN DO USING VOICE ASSISTANTS	45%	40%	45%
I FEEL COMFORTABLE USING MY VOICE ASSISTANT WHEN I'M WITH MY FRIENDS AND FAMILY	45%	42%	28%
I WANT TO BE ABLE TO DO MORE WITH A VOICE ASSISTANT	46%	40%	35%
VOICE ASSISTANTS HELP ME MULTITASK	36%	36%	36%
VOICE ASSISTANTS AREN'T FOR ME	34%	31%	30%
VOICE ASSISTANTS ARE USEFUL, FOR HELPING TO KEEP ME ORGANISED	34%	35%	39%
I TURN OFF MY VOICE ASSISTANT WHEN I'M NOT USING IT	35%	32%	36%
I DON'T TRUST VOICE ASSISTANTS	36%	31%	38%
I FIND VOICE ASSISTANTS FRUSTRATING	36%	31%	22%
I FEEL COMFORTABLE USING MY VOICE ASSISTANT IN PUBLIC	27%	22%	30%
I CAN NEVER GET THE ANSWERS I WANT WHEN USING VOICE ASSISTANTS	26%	29%	20%
VOICE ASSISTANTS CONFUSE ME	26%	26%	22%

User Attitudes towards Voice Assistants
 Copyright © 2021 Vixen Labs Limited – Voice Consumer Index June 2021 created in partnership with Open Voice Network.
 Research conducted by Delineate in May 2021

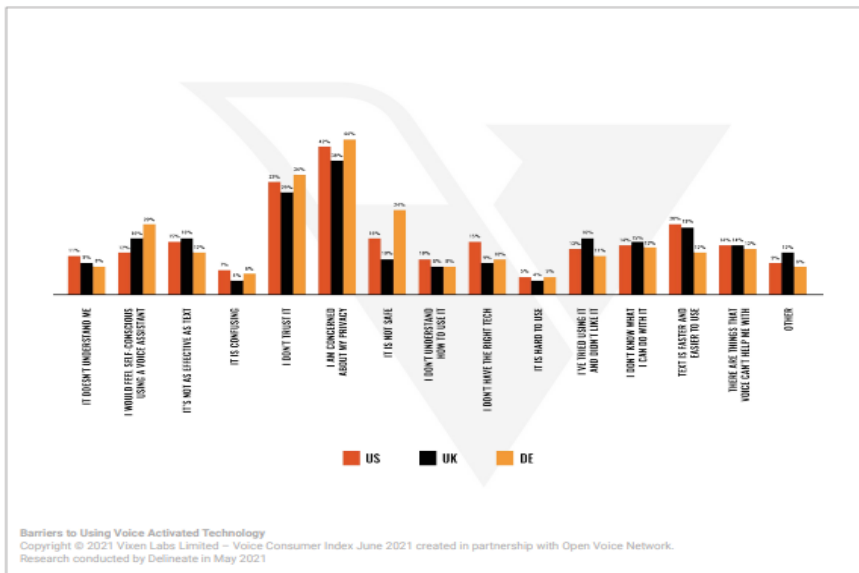
FOR MORE DETAIL, DOWNLOAD THE WHITE PAPER AT VIXENLABS.CO

FINDING FOUR

PRIVACY IS A CONCERN BUT NOT A BARRIER AMONG USERS

Users see the value and benefit of Voice in their lives and they are interested in doing more. Privacy is not just about data, it's also about who is around.

1.4: User opinions regarding privacy issues in virtual assistants.



Barriers to Using Voice Activated Technology
 Copyright © 2021 Vixen Labs Limited – Voice Consumer Index June 2021 created in partnership with Open Voice Network.
 Research conducted by Delineate in May 2021

FOR MORE DETAIL, DOWNLOAD THE WHITE PAPER AT VIXENLABS.CO

FINDING FIVE

PRIVACY, TRUST, AND EVEN SAFETY ARE BARRIERS FOR NON-USERS

They understand Voice but they do not see enough value to push their comfort levels.

1.5: Barriers faced by users who hesitate to use voice technology in their daily activities.

	US	UK	GERMANY
TRIAL AND ERROR / I HAD A GO	70%	76%	85%
PRODUCT INSTRUCTIONS OR PACKAGING	52%	57%	52%
A FRIEND OR FAMILY MEMBER SHOWED ME	52%	46%	52%
RECOMMENDATION FROM A FRIEND OR FAMILY MEMBER	47%	47%	51%
I LISTENED IN TO HOW A FRIEND OR FAMILY MEMBER SPOKE TO THE VOICE ASSISTANT AND I COPIED THEM	47%	46%	52%
THE PRODUCT'S WEBSITE OR WELCOME EMAIL	45%	47%	38%
SEARCHING ONLINE FOR ALEXA SKILLS OR ACTIONS ON GOOGLE	42%	47%	56%
EXAMPLES SHOWN IN ADVERTISING SUCH AS TV, RADIO ADVERTS OR SOCIAL	38%	38%	43%
READING CONSUMER OR TECH NEWS, BLOGS OR REVIEWS TO FIND OUT WHAT I CAN DO WITH MY VOICE ASSISTANT	30%	34%	43%

Ways Users Learn to Use Voice Assistants
 Copyright © 2021 Vixen Labs Limited – Voice Consumer Index June 2021 created in partnership with Open Voice Network.
 Research conducted by Delineate in May 2021

FOR MORE DETAIL, DOWNLOAD THE WHITE PAPER AT VIXENLABS.CO

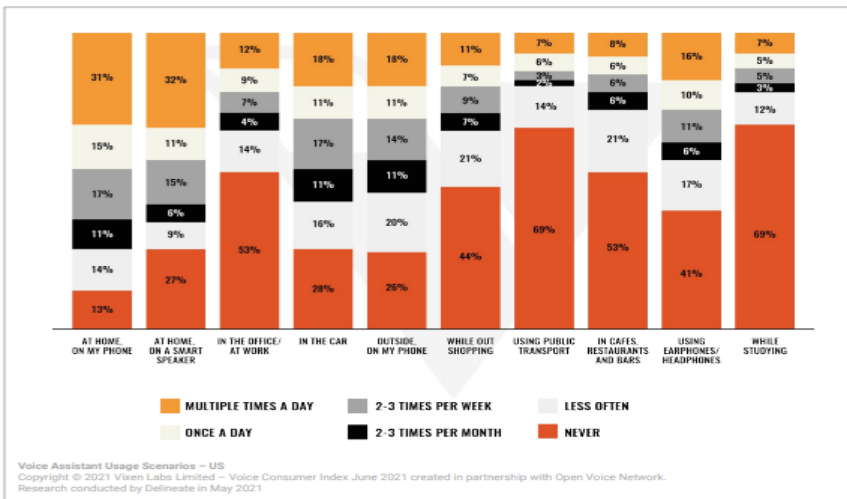
FINDING SIX

USERS SEE THE VALUE OF VOICE BUT NEED MORE SUPPORT FROM BRANDS TO LEARN ALL THAT IT CAN DO FOR THEM

Most users are relying on trial and error to find the experiences they want.



1.6: Initial introduction avenues to this technology for users



Voice Assistant Usage Scenarios – US
 Copyright © 2021 Vixen Labs Limited – Voice Consumer Index June 2021 created in partnership with Open Voice Network.
 Research conducted by Delineate in May 2021

FOR MORE DETAIL, DOWNLOAD THE WHITE PAPER AT VIXENLABS.CO

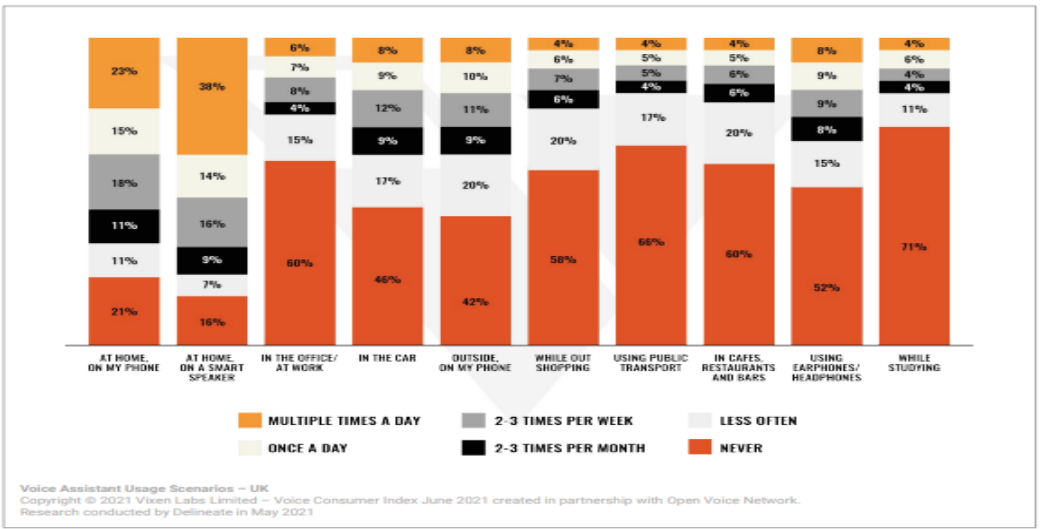
US

FINDING SEVEN

HOME IS WHERE THE VOICE ASSISTANT IS USED MOST

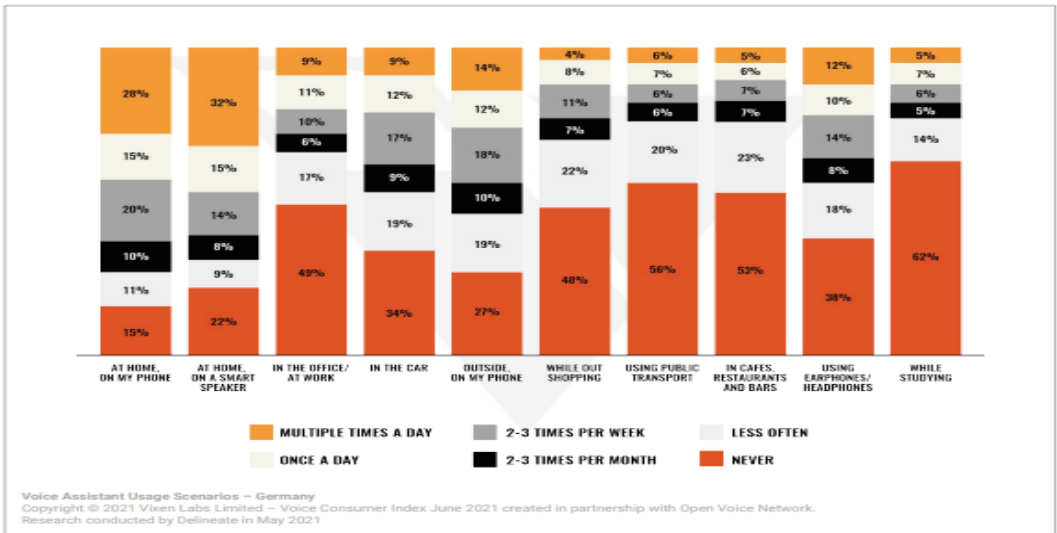
Voice assistants are primarily used at home on smart speakers and mobile devices. The US and Germany have higher rates of Voice assistant usage on smartphones outside than audiences in the UK.

UK



FOR MORE DETAIL, DOWNLOAD THE WHITE PAPER AT VIXENLABS.CO

GERMANY



FOR MORE DETAIL, DOWNLOAD THE WHITE PAPER AT VIXENLABS.CO

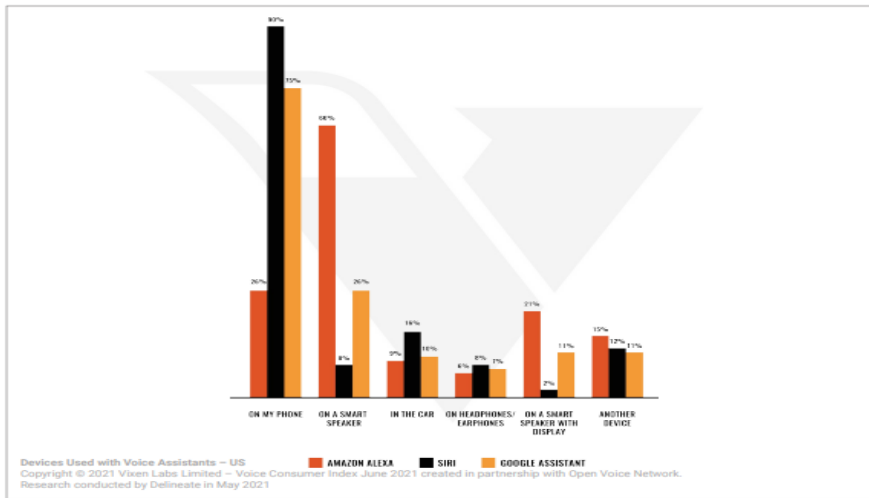
1.7: Place and frequency of use of virtual assistants

US

FINDING EIGHT

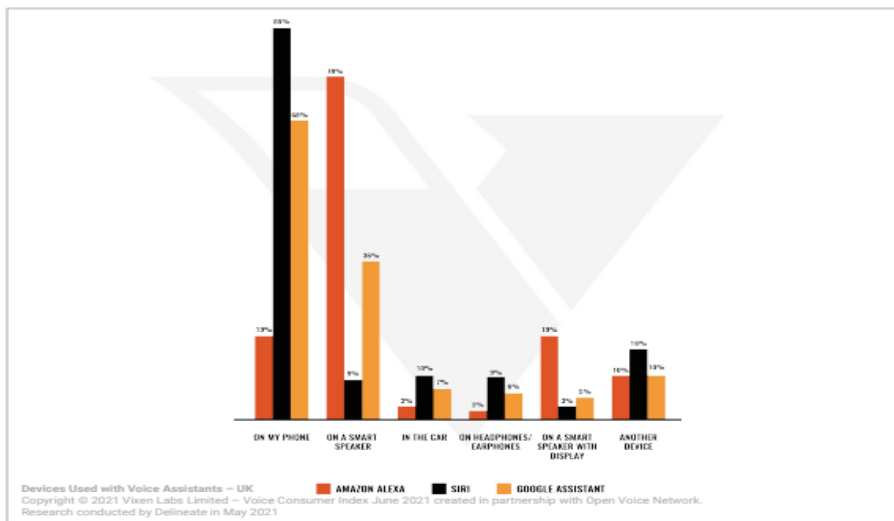
VOICE GOES BEYOND THE SMART SPEAKER

Alexa is primarily used on smart speakers while Siri is used mostly on smartphones. Google Assistant has high usage on smartphones but also has usage on smart speakers.



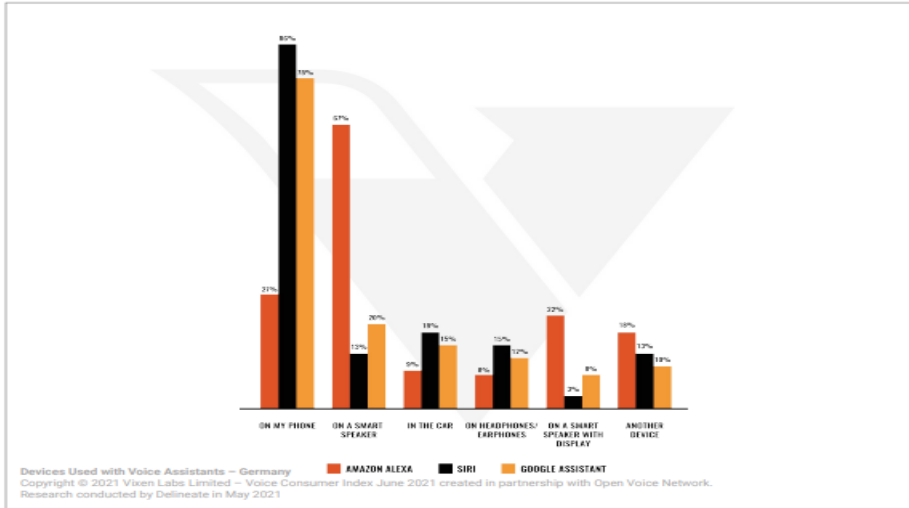
FOR MORE DETAIL, DOWNLOAD THE WHITE PAPER AT VIXENLABS.CO

UK



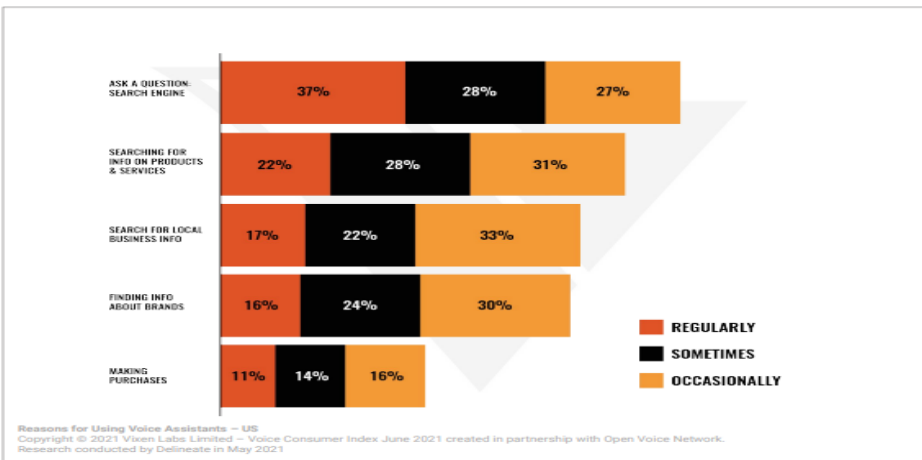
FOR MORE DETAIL, DOWNLOAD THE WHITE PAPER AT VIXENLABS.CO

GERMANY



1.8: Market share of popular virtual assistant software's available

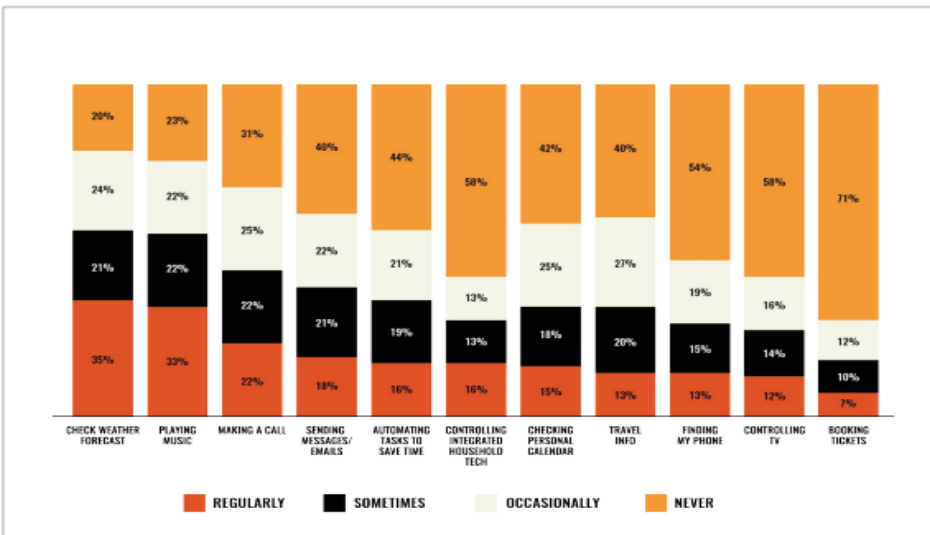
US



FINDING NINE
VOICE ACTIVITIES ALIGN WITH THE MARKETING FUNNEL
 Voice can be part of the marketing ecosystem and enhance the customer journey.

1.9: Utilization of virtual assistant technology in marketing

US



1.10: Utilization of technology for daily activities by normal users.

	US	UK	GERMANY
WEATHER	71%	75%	74%
MUSIC	66%	72%	69%
NEWS	54%	67%	67%
ENTERTAINMENT	53%	49%	53%
RETAIL	44%	34%	42%
HEALTHCARE & WELLNESS	42%	39%	43%
FOOD DELIVERY & RESTAURANTS	39%	36%	40%
LOCAL SERVICES	36%	37%	39%
CONSUMER PACKAGED GOODS	36%	31%	43%
TRAVEL	34%	33%	35%
FITNESS	34%	34%	41%
MAKING A RESERVATION	33%	31%	38%
FASHION	31%	29%	40%
FINANCE	30%	29%	32%
OTHER	22%	21%	29%

Likelihood of Using Voice Activated Searches by Sector
 Copyright © 2021 Vixen Labs Limited – Voice Consumer Index June 2021 created in partnership with Open Voice Network.
 Research conducted by Delineate in May 2021

FOR MORE DETAIL, DOWNLOAD THE WHITE PAPER AT VIXENLABS.CO

FINDING TEN

VOICE SEARCH IS AN OPPORTUNITY FOR ALL INDUSTRIES

However, the urgency to stake brand authority is greater in some industries than others.

1.11: Industry-wise impact of voice technology

2) Present system in use:

The present system in use is provided with lots of Services and facilities .The users can access the information they need though search engines after typing their needs physically.

In current system content is provided after searching the information via typing in search engine.

Users also have to find the certain in-built applications or downloaded applications manually before finding using them and completing their original objective.

3) Flaws in present system:

As looking to the present system due to outdated technology, there are many flaws and issues faced by the users. The perceived flaws are:

- **Time Consuming:** The present system consumes a lot more time as it requires manual searching and typing to find the actual application before trying to use it. This is not convenient as many times the user is in a hurry and must find a way to find things quickly and efficiently.
- **Poor or Outdated technology:** As present system is outdated users will not be satisfied as the efficiency of the system as well as the look and feel of the system is outdated and feels makes the user feel technologically outdated.

- **Lack of Personalization:** The current system lacks personalization that can make the user feel valued and special. This leads to eventual boredom and encourages the user to find other and more suitable systems.
- **Inefficient Data Parsing:** Manual information parsing is inefficient and time consuming. Many times, user searches for data in the wrong place. This leads to wastage of time for user and frustration as well.
- **Slow loading Time:** Due to outdated version slow loading time can absolutely kill the experience of users when they try to use the current system. This speeds up the process of obsolescence for the current system.
- **Remote Cyber Attacks:** The possibilities of cyber-attacks by opening the wrong websites are immensely high. Unaware users fall prey to these tactics most often as search engine may include such malicious links in the search results as well.
- **Data Security:** Websites accessed through the search engines may require sign-up. This may put your data in a vulnerable and easily accessible position.
- **Higher Level Specs Required:** Higher level Virtual Assistants require higher level specifications to run as well as compulsory internet connection.

4) **Needs of new system:**

This website is designed in such ways which help in overcome all the flaws which is in current system. The present system is outdated and also security in not provided by current system so there will be need of new system.

- **Outdated Techniques and coding:** If an application is developed several- years ago it probably has a lot of deprecated methods and modules that are not supported by modern browsers and new versions of OS.
- **Lack of Automation:** There is no scope of automation in present system as it requires manual input and needs user to physically search the applications before processing them.
- **Not As User Friendly:** Present system is not as user friendly as the new replacement system.
- **Responsive:** The speed of response of the present system is not as fast as the speed of the system being developed.
- **Hands-On:** The present system is very hands on and requires complete attention of the user and therefore hinders his ability to multitask. The new system that is being developed will get rid of any such obstacles and help to optimize the time of the user.

- **Time Optimization:** As the new system does not require a hands-on approach, it's easier for the user to optimize how they spend their time and also to be able to multitask. This makes good on the saying 'Time is money'.
- **No Need for Special Specifications:** The high level systems require special and high level specs. However, the new system does not require such special specs and can be run on the available specs as needed.

5) **Feasibility Study:**

A feasibility study is an analysis that takes all of a project's relevant factors into account—including economic, technical, legal, and scheduling considerations—to ascertain the likelihood of completing the project successfully.

A feasibility study is part of the initial design stage of any project/plan. It is conducted in order to objectively uncover the strength and weaknesses of a proposed project or an existing business. It can help to identify and assess the opportunities and threats present in the natural environment, the resources required for the project, and the prospects for success.

A feasibility study is an evaluation and analysis of a project or system that somebody has proposed. We also call it a feasibility analysis.

Feasibility Study in Software Engineering is a study to evaluate feasibility of proposed project or system. Feasibility study is one of stage among important four stages of Software Project Management Process. As name suggests

feasibility study is the feasibility analysis or it is a measure of the software product in terms of how much beneficial product development will be for the organization in a practical point of view. Feasibility study is carried out based on many purposes to analyze whether software product will be right in terms of development, implantation, contribution of project to the organization etc.

- **Types of Feasibility Study :**

The feasibility study mainly concentrates on bellow five mentioned areas. Among these Economic Feasibility Study is most important part of the feasibility analysis and Legal Feasibility Study is less considered feasibility analysis.

- a. Technical Feasibility –**

In Technical Feasibility current resources both hardware software along with required technology are analyzed/assessed to develop project. This technical feasibility study gives report whether there exists correct required resources and technologies which will be used for project development. Along with this, feasibility study also analyses technical skills and capabilities of technical team, existing technology can be used or not, maintenance and up- gradation is easy or not for chosen technology etc.

Technical feasibility includes finding out technologies for the project, both hardware and software. For virtual assistant, user must have microphone to convey their message and a speaker to listen when system speaks. These are very cheap now a days and everyone generally possess them. Besides, system needs internet connection. While using IVA, make sure you have a steady internet connection. It is also not an issue in this era where almost every home or office has Wi-Fi.

b. Operational Feasibility –

In Operational Feasibility degree of providing service to requirements is analyzed along with how much easy product will be to operate and maintenance after deployment. Along with this other operational scopes are determining usability of product, Determining suggested solution by software development team is acceptable or not etc.

In operational feasibility analysis the ease and simplicity of operation of proposed system is highlighted. System does not require any special skill set for users to operate it. In fact, it is designed to be used by almost everyone. Kids who still don't know to write can read out problems for system and get answers

c. Economic Feasibility –

In Economic Feasibility study cost and benefit of the project is analyzed. Means under this feasibility study a detail analysis is carried out what will be cost of the project for development which includes all required cost for final development like hardware and software resource required, design and development cost and operational cost and so on. After that it is analyzed whether project will be beneficial in terms of finance for organization or not.

Here, we find the total cost and benefit of the proposed system over current system. For this project, the main cost is documentation cost. User also would have to pay for microphone and speakers. Again, they are cheap and available. As far as maintenance is concerned, VIA won't cost too much.

d. Legal Feasibility –

In Legal Feasibility study project is analyzed in legality point of view. This includes analyzing barriers of legal implementation of project, data protection acts or social media laws, project certificate, license, copyright etc. Overall it can be said that Legal Feasibility Study is study to know if proposed project conform legal and ethical requirements.

The project IVA meets all legal and ethical requirements of the as needed as it has used solely open source references and has not violated any legal or ethical boundaries that the developers are aware of.

e. Social feasibility -

Social feasibility is a detailed study on how one interacts with others within a system or an organization. Social impact analysis is an exercise aimed at identifying and analyzing such impacts in order to understand the scale and reach of the project's social impacts.

Virtual assistant is built in accordance with the general culture. The project is named IVA so as to not offend any culture or undermine any local beliefs.

PROJECT **CATEGORY**

Project category:

In this project “IVA: Intellectual Virtual Assistant” we use Python language, HTML and PHP as frontend and MYSQL as database and for styling of application, CSS has also been used.

Python:

Python is an interpreted, interactive, object-oriented programming language. It incorporates modules, exceptions, dynamic typing, very high level dynamic data types, and classes. It supports multiple programming paradigms beyond object-oriented programming, such as procedural and functional programming. Python combines remarkable power with very clear syntax. It has interfaces too many system calls and libraries, as well as to various window systems, and is extensible in C or C++. It is also usable as an extension language for applications that need a programmable interface. Finally, Python is portable: it runs on many UNIX variants including Linux and macOS, and on Windows.

- Python’s elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms.
- It can be integrated with many databases such as Oracle, Microsoft SQL Server, MySQL, PostgreSQL, Sybase, and Informix.

- The Python interpreter is easily extended with new functions and data types implemented in C or C++ (or other languages callable from C). Python is also suitable as an extension language for customizable applications.
- Python is simple to use, but it is a real programming language, offering much more structure and support for large programs than shell scripts or batch files can offer. On the other hand, Python also offers much more error checking than C, and, being a *very-high-level language*, it has high-level data types built in, such as flexible arrays and dictionaries. Because of its more general data types Python is applicable to a much larger problem domain than Awk or even Perl, yet many things are at least as easy in Python as in those languages.
- Python allows you to split your program into modules that can be reused in other Python programs. It comes with a large collection of standard modules that you can use as the basis of your programs — or as examples to start learning to program in Python. Some of these modules provide things like file I/O, system calls, sockets, and even interfaces to graphical user interface toolkits like Tk.
- Many popular applications are built on python such as:
 - Netflix
 - Google
 - YouTube
 - Instagram

- Uber
 - Pintrest
 - Dropbox
 - Quora, etc.
-
- Python is a high-level general-purpose programming language that can be applied to many different classes of problems.
 - The language comes with a large standard library that covers areas such as string processing (regular expressions, Unicode, calculating differences between files), internet protocols (HTTP, FTP, SMTP, XML-RPC, POP, IMAP, CGI programming), software engineering (unit testing, logging, profiling, parsing Python code), and operating system interfaces (system calls, filesystems, TCP/IP sockets).
 - The version used to develop this project is the Python 3.10.2

MySQL:

MySQL is an open-source relational database management system (RDBMS). It is the most popular database system used with PHP. MySQL is developed, distributed, and supported by Oracle Corporation.

- The data in a MySQL database are stored in tables which consists of columns and rows.
- MySQL is a database system that runs on a server.

- MySQL is ideal for both small and large applications.
- MySQL is very fast, reliable, and easy to use database system. It uses standard SQL.
- MySQL compiles on a number of platforms.
- MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.
- MySQL works very quickly and works well even with large data sets.
- MySQL is very friendly to PHP, the most appreciated language for web development.

SOFTWARE AND
HARDWARE
REQUIREMENTS
SPECIFICATIONS

SOFTWARE AND HARDWARE REQUIREMENT SPECIFICATIONS-

Every application needs the software in which it has to be executed and a hardware the application is going to perform its function. Some application cannot run on every platforms and some applications needs some specific requirement in the software or in hardware to get operated. Let's take an example of the applications which cannot be run on every platforms like windows, android, Linux, etc. Applications made in visual basic is only supported for the windows, one cannot access this applications from the mobile phones, etc. So, here are some hardware and software specifications which are mandatory for the application to get operated.

HARDWARE

Hardware is a term that refers to all the physical parts that make up a computer. The internal hardware devices that make up the computer. Various devices which are essentials to form a hardware is called as components.

Following are the hardware specifications that is required to develop this project is as follows:

Computer components like Monitor, Keyboard, Mouse, CPU, microphone, (For using IVA), camera

Minimum 8 GB ram for smooth working of application.

1TB Hard Disk or More. CD ROM

Drive.

Wi-Fi Adaptor or an active internet connection.

SOFTWARE

Software can be termed as the group of instruction or command used by the computer to accomplish the given task. It can be said as a set of instructions or programs instructing a computer to do specific task. Software in general term is used to describe the computer programs.

Following are the software specifications that is required to develop this project is as follows:

Language Used (Front End) – Python 3.10.2

Database Used (Back End) – MySQL, MySQL WORKBENCH

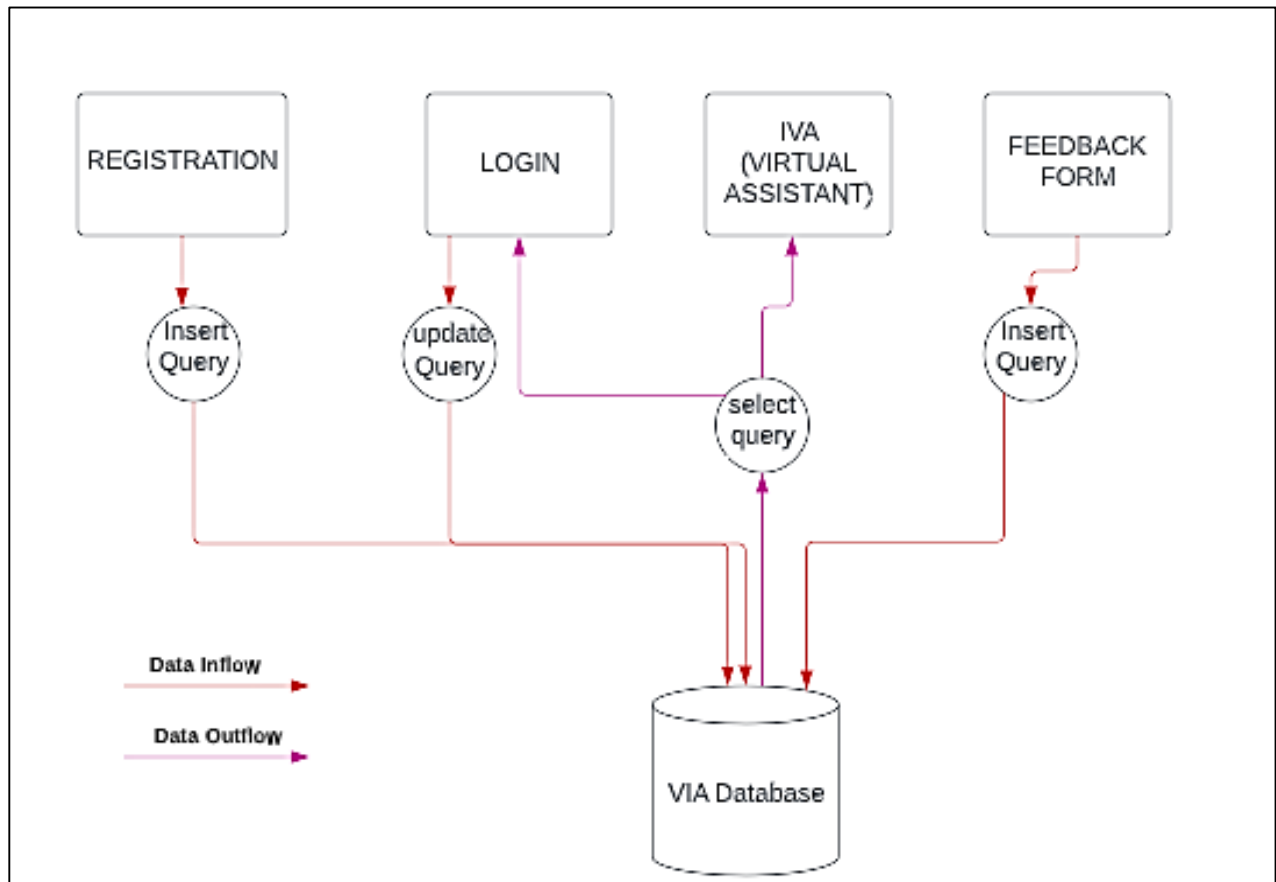
Supporting Languages Used – Python 3.10.2

IDEs Used – Pycharm

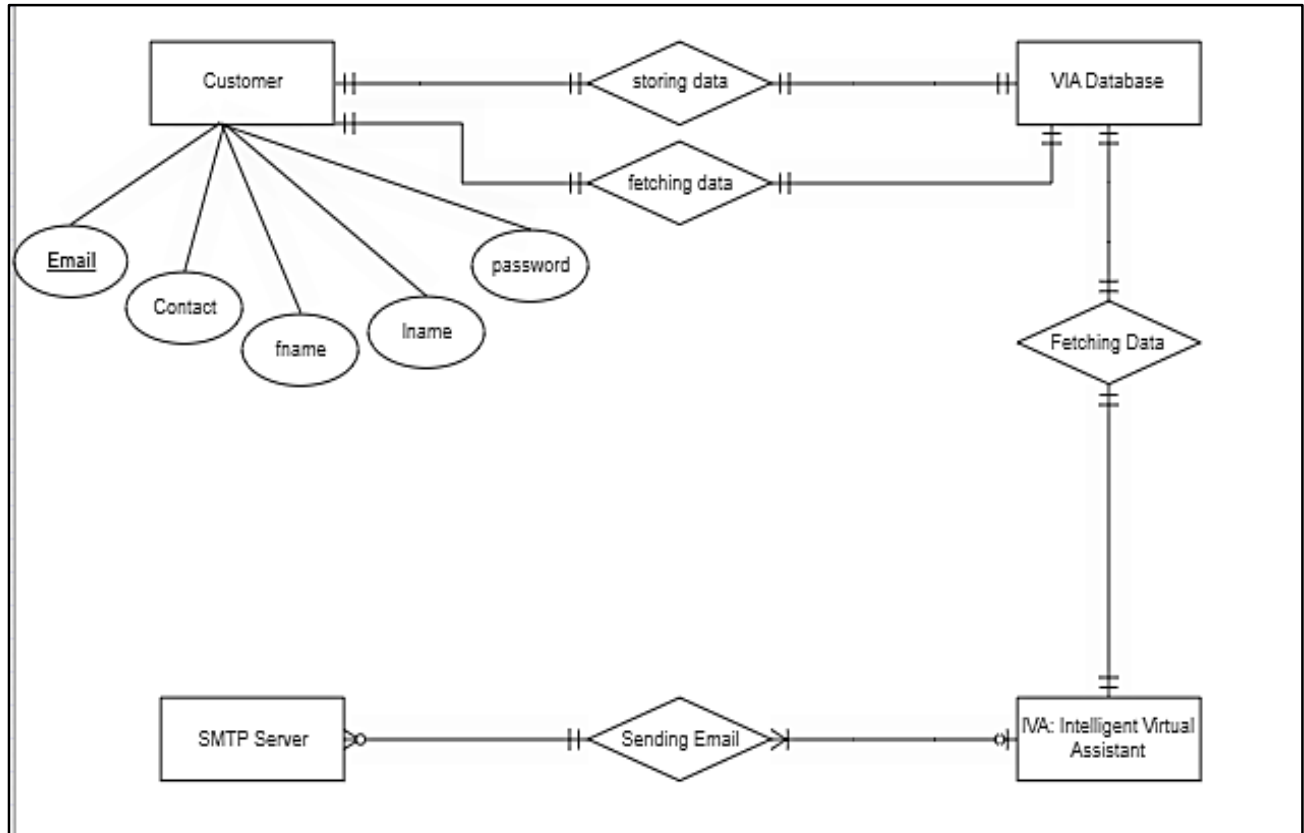
Third-Party Software – Audacity sound editor, Free Alarm Clock Desktop application.

Detailed System Analysis

DATA FLOW DIAGRAM:

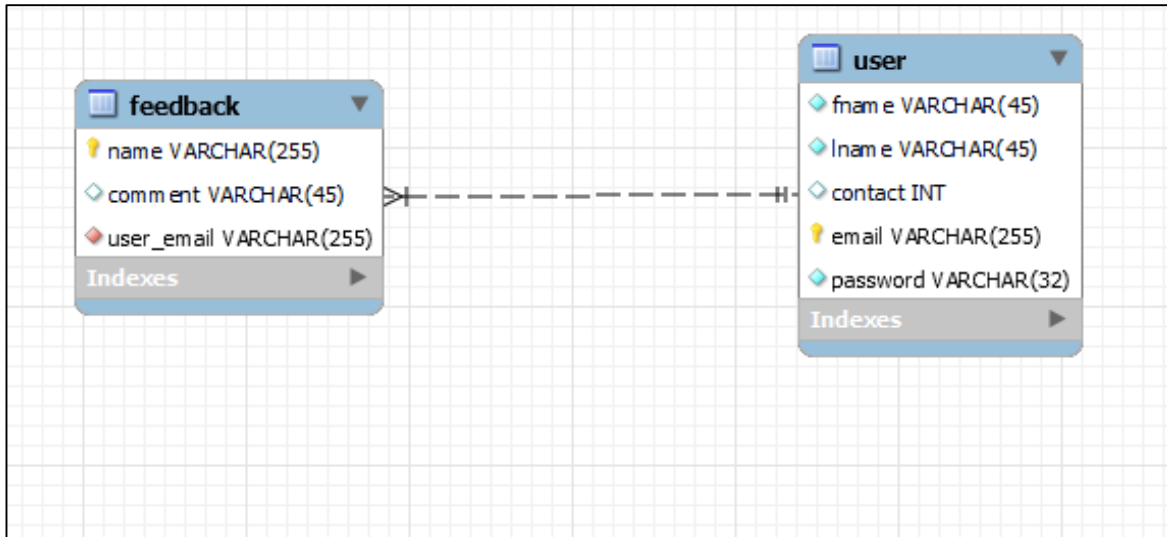


ENTITY RELATIONSHIP DIAGRAM:



DATA STRUCTURES AND TABLES:

Information		fname	lname	contact	email	password
Table: user		qjkg	vnddix	1234556662	233@gmail	123456
Columns:		Chinmayee	Dongare	1234999999	chinu@gmail.com	12
fname	varchar(45)	Kilua	Zoldyck	112233445	Kilua@outlook.com	12333
lname	varchar(45)	Manalee	Gedam	1234567899	manalee@gmail.com	1234
contact	int	mr.	Dragon	444444444	mr._hoarder@gmial.com	44
<u>email</u>	varchar(255) PK	nana	patekar?	1223344556	nana_pats@gmail.com	11
password	varchar(32)	Neha	M	123456778	neha@gmail.com	33
		person	lname	222222222	qq	12
		tt	tt	333333334	r33	333
		Blue	Sea	222222222	Seaeu@gmail.com	22
		Mahima	Dwivedi	1234567899	tanu@gmail.com	1234
		Tanushree	Biroley	1234555678	TanushreeB@gmail.com	123
		rr	rr	333333333	ww	1
		Zahed Ah...	Khwajah	123445678	ZahedBhai@gmail.com	11
		NULL	NULL	NULL	NULL	NULL



Information

Table: feedback

Columns:

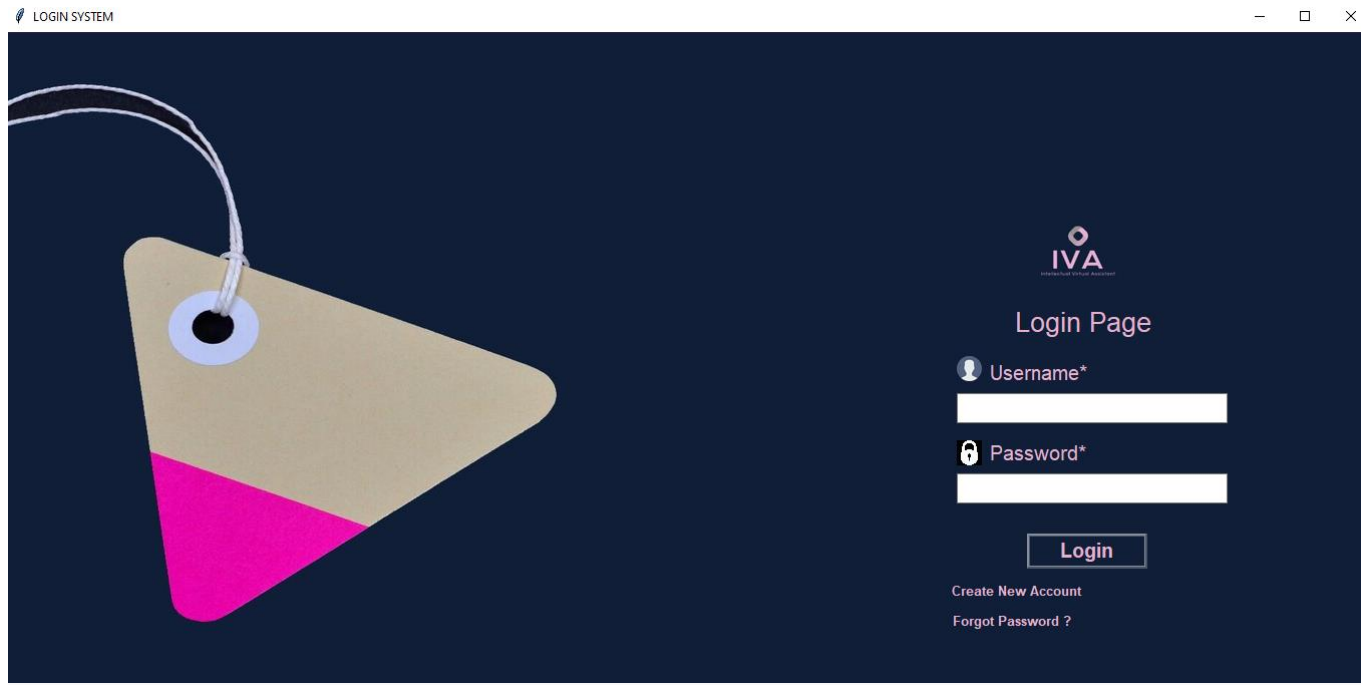
name	comment	user_email
fg	yup	asd@gmail.com
if	if i break my back today?	asd@gmail.com
jhhh	uuu	asd@gmail.com
mahima	working for the weekend	asd@gmail.com
Manalee	TEST	asd@gmail.com
q	test	asd@gmail.com
qwwq	comment	qr@gmail.com
uii	will you pat my back tomorrow	asd@gmail.com
uuuu	but that weekend never came	asd@gmail.com

name varchar(255) PK
 comment varchar(45)
 user_email varchar(45)

SYSTEM **DESIGN**

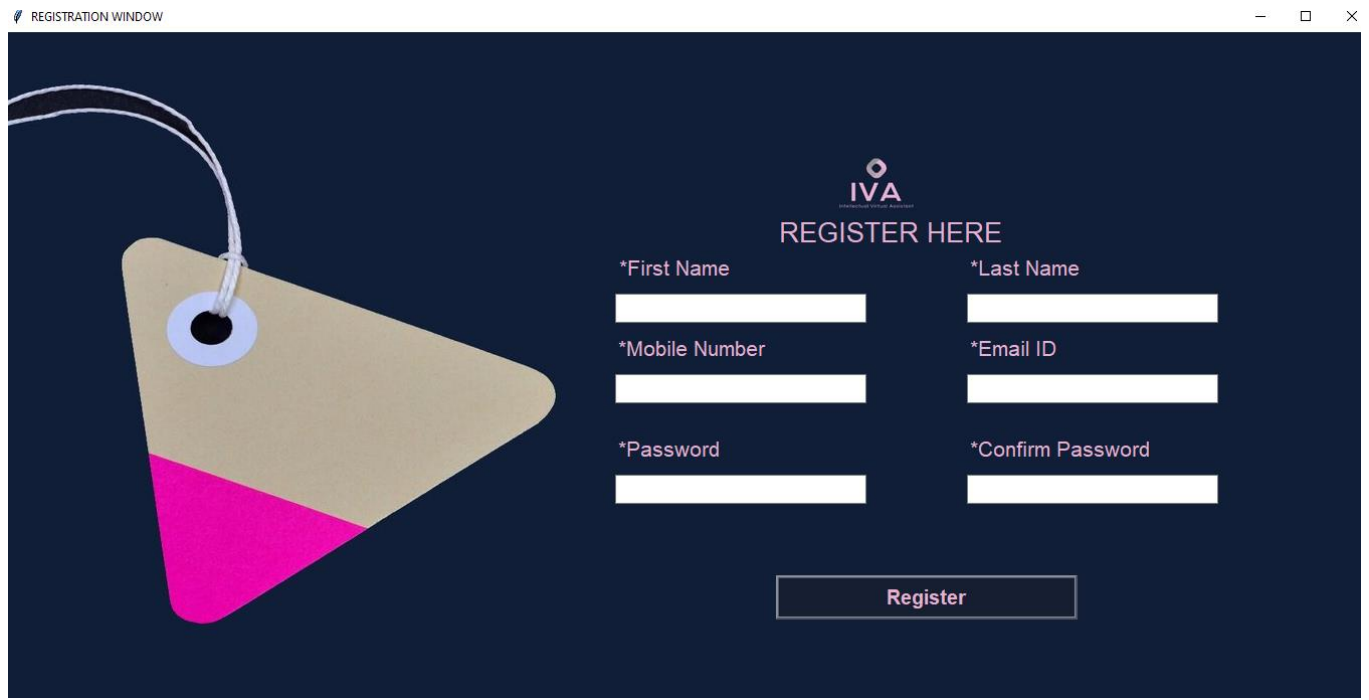
Form Design

Login Form:



The screenshot shows a login form titled "LOGIN SYSTEM" in a window. On the left is a decorative image of a triangular tag with a white string and a hole, split into tan and pink sections. On the right, the IVA logo is at the top, followed by the text "Login Page". Below this are two input fields: "Username*" and "Password*", each with a corresponding icon (a person and a lock). A "Login" button is positioned below the password field. At the bottom, there are two links: "Create New Account" and "Forgot Password?".


Registration Form:



The screenshot shows a registration form titled "REGISTRATION WINDOW" in a window. On the left is the same decorative triangular tag image as in the login form. On the right, the IVA logo is at the top, followed by the text "REGISTER HERE". Below this are six input fields arranged in two columns: "*First Name", "*Last Name", "*Mobile Number", "*Email ID", "*Password", and "*Confirm Password". A "Register" button is located at the bottom center.

Feedback Form

IVA FEEDBACK SYSTEM

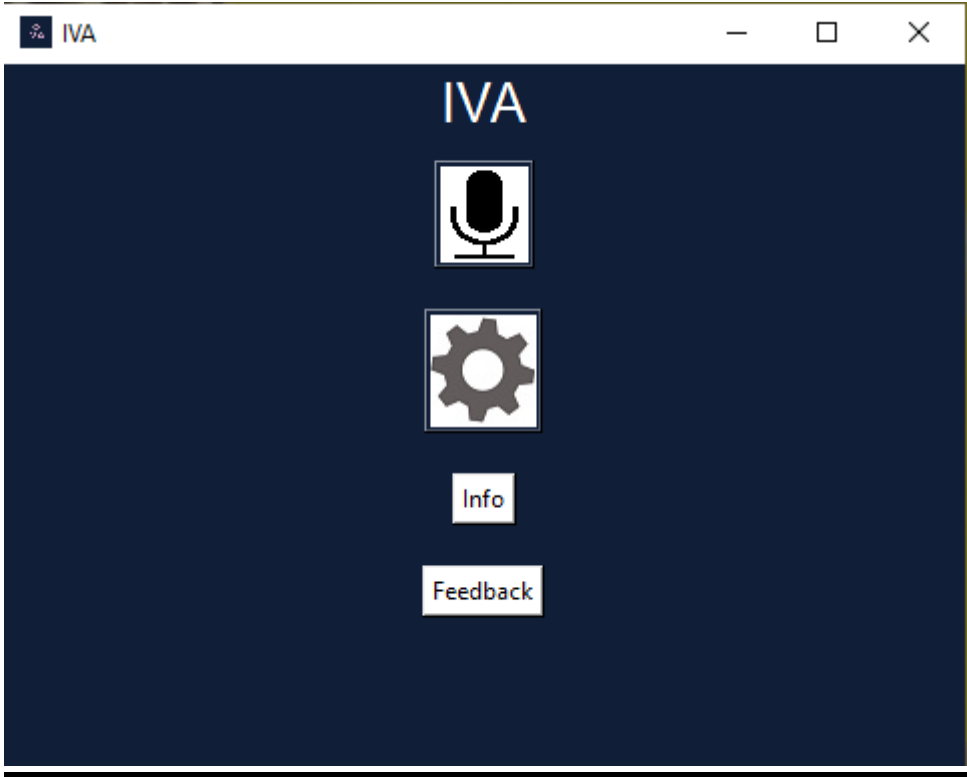
 **Thank you for visiting!**

*Name: *Email(same as earlier):

*Comments:

Submit Clear

IVA GUI:



SOURCE CODE:

Front_End.py File (Login and registration with MySQL

workbench connectivity):

```
import importlib
from tkinter import ttk
from tkinter import messagebox
import mysql.connector
from tkinter import *
from PIL import ImageTk, Image
#from Virtual_Assistant import main_screen
```

```
# import splash_screen
```

```
def main():
    # splash_screen.splash_root.destroy()
    MainScreen = Tk()
    app = Log_win(MainScreen)
    MainScreen.mainloop()
    #win.iconbitmap('app_icon.ico')
```

```
class Log_win:
```

```

def __init__(self, root):
    self.app = None
    self.root = root
    self.root.title("LOGIN SYSTEM")
    self.root.geometry("{0}x{1}+0+0".format(root.winfo_screenwidth(),
root.winfo_screenheight()))

    self.backImg = ImageTk.PhotoImage(file="user (1).jpg")
    label1_backImg = Label(self.root, image=self.backImg)
    label1_backImg.place(x=0, y=0, relwidth=1, relheight=1)

    logFrame = Frame(self.root, bg="#101e38")
    logFrame.place(x=910, y=170, width=340, height=450)

    img1 = Image.open("logo.jpg")
    img1 = img1.resize((110, 110), Image.ANTIALIAS)
    self.photoimage1 = ImageTk.PhotoImage(img1)

    labimg1 = Label(image=self.photoimage1, bg="#101e38", borderwidth=0)
    labimg1.place(x=1021, y=168, width=100, height=100)

    head = Label( logFrame, text="Login Page", font=("Helvetica", 20), fg="#e8b1d1",
bg="#101e38")
    head.place(x=95, y=100)

# Label

```

```

    Uname = lb1 = Label( logFrame, text="Username*", font=("Helvetica", 15),
fg="#e8b1d1", bg="#101e38")
    Uname.place(x=70, y=155)

    self.txtuser = ttk.Entry( logFrame, font=("Helvetica", 15, "bold"))
    self.txtuser.place(x=40, y=190, width=270)

    password = lb1 = Label( logFrame, text="Password*", font=("Helvetica", 15),
fg="#e8b1d1", bg="#101e38")
    password.place(x=70, y=235)

    self.txtpass = ttk.Entry( logFrame, font=("Helvetica", 15, "bold"), show='*')
    self.txtpass.place(x=40, y=270, width=270)

# Icon Image
Uimg = Image.open("user.png")
Uimg = Uimg.resize((25, 25), Image.ANTIALIAS)
self.photoimage2 = ImageTk.PhotoImage(Uimg)

labimg1 = Label(image=self.photoimage2, bg="#101e38", borderwidth=0)
labimg1.place(x=950, y=323, width=25, height=25)

img3 = Image.open("pass.png")
img3 = img3.resize((25, 25), Image.ANTIALIAS)
self.photoimage3 = ImageTk.PhotoImage(img3)

labimg1 = Label(image=self.photoimage3, bg="#101e38", borderwidth=0)

```

```
labimg1.place(x=950, y=405, width=25, height=30)
```

```
# Button
```

```
loginbtn = Button( logFrame, text="Login", font=("Helvetica", 15, "bold"), bd=3,  
relief=RIDGE, fg="#e8b1d1",
```

```
bg="#101e38", command=self.login)
```

```
loginbtn.place(x=110, y=330, width=120, height=35)
```

```
registerbtn = Button( logFrame, text="Create New Account",  
command=self.registration_window, font=("Helvetica", 10, "bold"),
```

```
borderwidth=0, fg="#e8b1d1", bg="#101e38")
```

```
registerbtn.place(x=20, y=375, width=160)
```

```
forgetbtn = Button( logFrame, text="Forgot Password ?",  
command=self.forgot_pass, font=("Helvetica", 10, "bold"),
```

```
borderwidth=0, fg="#e8b1d1", bg="#101e38")
```

```
forgetbtn.place(x=15, y=405, width=160)
```

```
def registration_window(self):
```

```
self.new_window = Toplevel(self.root)
```

```
self.app = Register(self.new_window)
```

```
def login(self):
```

```
if self.txtuser.get() == "" or self.txtpass.get() == "":
```

```
    messagebox.showerror("Error", "Username or Password cannot be empty.")
```

```
elif self.txtuser.get() == "Manalee" and self.txtpass.get() == "Mahima":
```

```

        messagebox.showinfo("Success", "Login Successful.")
else:
    conn = mysql.connector.connect(host="localhost", user="root",
password="Mahima123$", database="via")
    my_cursor = conn.cursor()
    my_cursor.execute("select * from user where Email=%s and Password=%s", (
        self.txtuser.get(),
        self.txtpass.get()
    ))
    row = my_cursor.fetchone()
    if row == None:
        messagebox.showerror("Error", "Invalid Username and Password")
    else:
        further = messagebox.askyesno("YesNo", "Do you want to continue?")
        if further > 0:
            self.new_window = Toplevel(self.new_window)

            #exec("import " + "Virtual_Assistant")
            #Toplevel(main_screen)
            #__import__("Virtual_Assistant")

            #new_window=importlib.import_module("Virtual_Assistant.py")

            #new_window = Toplevel(main_screen)

        else:
            exit()

```

```

conn.commit()
conn.close()

# Forgot Password

def forgot_pass(self):
    if self.txtuser.get() == "":
        messagebox.showerror("Error", "Please enter the email")
    else:
        conn = mysql.connector.connect(host="localhost", user="root",
password="Mahima123$",
                                database="via")
        my_cursor = conn.cursor()
        query = ("select * from user where Email=%s")
        value = (self.txtuser.get(),)
        my_cursor.execute(query, value)
        row = my_cursor.fetchone()
        # print(row)

        if row == None:
            messagebox.showerror("Error", "Please Enter the valid Username")
        else:
            conn.close()
            self.root2 = Toplevel()
            self.root2.title("Forgot Password")
            self.root2.geometry("340x450+610+170")

```

```

l = Label(self.root2, text="Forgot Password", font=("Helvetica", 20, "bold"),
fg="#e8b1d1", bg="#101e38")
l.place(x=0, y=10, relwidth=1)

new_mob = Label(self.root2, text="Enter Mobile Number", font=("Helvetica",
16, "bold"), fg="#e8b1d1", bg='#101e38')
new_mob.place(x=50, y=90)

self.new_mob_entry = ttk.Entry(self.root2, font=("Helvetica", 15))
self.new_mob_entry.place(x=50, y=130, width=250)

new_pass = Label(self.root2, text="Enter New Password", font=("Helvetica",
16, "bold"), fg="#e8b1d1", bg='#101e38')
new_pass.place(x=50, y=170)

self.new_pass_entry = ttk.Entry(self.root2, font=("Helvetica", 15), show='*')
self.new_pass_entry.place(x=50, y=210, width=250)

forgotbtn = Button(self.root2, text="Reset Password", font=("Helvetica", 9,
"bold"), bd=3, relief=RIDGE,
fg="#e8b1d1", bg='#101e38', command=self.reset_pass)
forgotbtn.place(x=110, y=280, width=120)

def reset_pass(self):
    if self.new_mob_entry.get() == "":
        messagebox.showerror("Error", "Enter Registered Mobile Number")

```



```

elif self.new_pass_entry.get() == "":
    messagebox.showerror("Error", "Enter New Password")

else:
    conn = mysql.connector.connect(host="localhost", user="root",
password="Mahima123$",
                                database="via")
    my_cursor = conn.cursor()
    query = ("select * from user where Email=%s and Password=%s")
    value = (self.txtuser.get(), self.txtpass.get())
    my_cursor.execute(query, value)
    row = my_cursor.fetchone()
    if row != None:
        messagebox.showerror("Error", "Enter valid Information")
    else:
        query = ("update user set Password=%s where Email=%s ")
        value = (self.new_pass_entry.get(), self.txtuser.get())
        my_cursor.execute(query, value)

        conn.commit()
        conn.close()
        messagebox.showinfo("Info", "Your Password has been Reset,Please Login
new Password")

```

```

class Register:

```

```

def __init__(self, root):
    self.root = root
    self.root.title("REGISTRATION WINDOW")
    self.root.geometry("{0}x{1}+0+0".format(root.winfo_screenwidth(),
root.winfo_screenheight()))

    # Variable
    self.var_fname = StringVar()
    self.var_lname = StringVar()
    self.var_contact = StringVar()
    self.var_email = StringVar()
    self.var_pass = StringVar()
    self.var_cpass = StringVar()

    self.bg = ImageTk.PhotoImage(file="user (1).jpg")
    bg_lab1 = Label(self.root, image=self.bg)
    bg_lab1.place(x=0, y=0, relwidth=1, relheight=1)

    frame = Frame(self.root, bg="#101e38", )
    frame.place(x=920, y=375, width=680, height=550, anchor=CENTER)

    img1 = Image.open("logo.jpg")
    img1 = img1.resize((110, 110), Image.ANTIALIAS)
    self.photoimage1 = ImageTk.PhotoImage(img1)

```

```

labimg1 = Label(frame, image=self.photoimage1, bg="#101e38", borderwidth=0)
labimg1.place(x=240, y=0, width=100, height=100)

# Label
reg_lbl = Label(frame, text="REGISTER HERE", font=("Helvetica", 20),
fg="#e8b1d1", bg="#101e38")
reg_lbl.place(x=190, y=80)

# Row1
# Firstname
fname = Label(frame, text="*First Name", font=("Helvetica", 16), fg="#e8b1d1",
bg="#101e38")
fname.place(x=31, y=120)

self.fname_entry = ttk.Entry(frame, textvariable=self.var_fname, font=("Helvetica",
15))
self.fname_entry.place(x=30, y=160, width=250)

# Lastname
lname = Label(frame, text="*Last Name", font=("Helvetica", 16), fg="#e8b1d1",
bg="#101e38")
lname.place(x=380, y=120)

self.lname_entry = ttk.Entry(frame, textvariable=self.var_lname, font=("Helvetica",
15))
self.lname_entry.place(x=380, y=160, width=250)

```

```

# Row2
contact = Label(frame, text="*Mobile Number", font=("Helvetica", 16),
fg="#e8b1d1", bg="#101e38")
contact.place(x=30, y=200)
self.txt_contact = ttk.Entry(frame, textvariable=self.var_contact, font=("Helvetica",
15))
self.txt_contact.place(x=30, y=240, width=250)

email = Label(frame, text="*Email ID", font=("Helvetica", 16), fg='#e8b1d1',
bg="#101e38")
email.place(x=380, y=200)
self.txt_email = ttk.Entry(frame, textvariable=self.var_email, font=("Helvetica", 15))
self.txt_email.place(x=380, y=240, width=250)

# Row3
pwd = Label(frame, text="*Password", font=("Helvetica", 16), fg="#e8b1d1",
bg="#101e38")
pwd.place(x=30, y=300)
self.txt_pwd = ttk.Entry(frame, textvariable=self.var_pass, font=("Helvetica", 15),
show='*')
self.txt_pwd.place(x=30, y=340, width=250)

cpwd = Label(frame, text="*Confirm Password", font=("Helvetica", 16),
fg="#e8b1d1", bg="#101e38")
cpwd.place(x=380, y=300)
self.txt_cpwd = ttk.Entry(frame, textvariable=self.var_cpass, font=("Helvetica", 15),
show='*')

```

```

self.txt_cpwd.place(x=380, y=340, width=250)

# button
self.var_check = IntVar()
self.checkbtn = Checkbutton(frame, text="I Agree To The Terms And COnditions.",
variable=self.var_check, font=("Helvetica", 15), fg="white", bg="#101e38", onvalue=1,
offvalue=0)
self.checkbtn.place(x=30, y=380)

# img=Image.open("regbtn.jpg")
# img = img.resize((200, 50), Image.ANTIALIAS)
# self.photoimage=ImageTk.PhotoImage(img)
#
b1=Button(frame,image=self.photoimage,borderwidth=0,cursor="hand2",font=("Arial",
15),bg="white")
# b1.place(x=10,y=420,width=300)
regbtn = Button(frame, text="Register", font=("Helvetica", 15, "bold"), bd=3,
relief=RIDGE, fg="#e8b1d1",
                bg="#161e32", command=self.register_data)
regbtn.place(x=190, y=440, width=300)
def register_data(self):
    if self.var_fname.get() == "" or self.var_lname.get() == "" or self.var_email.get() ==
"" or self.var_contact.get() == "":
        messagebox.showerror("Error", "All fields are required")

    elif self.var_pass.get() != self.var_cpass.get():
        messagebox.showerror("Error", "Password and Confirm Password must be

```

```

same.")
    elif self.var_check.get() == 0:
        messagebox.showerror("Error", "Please agree our Terms & Conditions")
    else:
        conn = mysql.connector.connect(host="localhost", user="root",
password="Mahima123$", database="via")
        my_cursor = conn.cursor()
        query = "select * from user where email=%s"
        value = (self.var_email.get(),)
        my_cursor.execute(query, value)
        row = my_cursor.fetchone()
        if row is None:
            my_cursor.execute("insert into user values(%s,%s,%s,%s,%s)", (
                self.var_fname.get(),
                self.var_lname.get(),
                self.var_contact.get(),
                self.var_email.get(),
                self.var_pass.get()
            ))
        else:
            messagebox.showerror("Error", "User already exist , Please try another email")
        conn.commit()
        conn.close()
        messagebox.showinfo("Success!", "Registered Successfully!!!")
        exit(Register)
if __name__ == '__main__':
    main()

```

Virtual Assistant.py file

```
import os
import subprocess as sp

import Feedback_Form
from Functions.Online_Ops import find_my_ip as ip, get_random_advice, \
    get_trending_movies, \
    get_weather_report, send_whatsapp_message
from Functions.os_ops import open_Chrome, open_cmd, open_Adobe_Reader, \
    open_Microsoft_Edge, open_Clock, open_Camera, open_Sound_recorder,
open_Skype, \
    open_Computer_Management, \
    open_Control_Panel, open_Bluetooth, open_Paint, open_On_Screen_Keyboard,
open_Snipping_Tool, open_Task_manager, \
    open_Telegram
import requests
import winshell
import ctypes
import wolframalpha
import speech_recognition as sr # To convert speech into text
import pyttsx3 # To convert text into speech
import datetime # To get the date and time
import wikipedia # To get information from wikipedia
import webbrowser # To open websites
import os # To open files
import time # To calculate time
import subprocess # To open files
from tkinter import * # For the graphics
import pyjokes # For some really bad jokes
from playsound import playsound # To playsound
import keyboard # To get keyboard
import requests, json
from Feedback_Form import Feedback
name_file = open("Assistant_name", "r")
name_assistant = name_file.read()

engine = pyttsx3.init('sapi5')
engine.setProperty('volume', 1.0)
engine.setProperty('rate', 190)
```

```
voices = engine.getProperty('voices')
engine.setProperty('voice', voices[1].id)
```

```
def speak(text):
    engine.say(text)
    print(name_assistant + " : " + text)
    engine.runAndWait()
```

```
def wishMe():
    hour = datetime.datetime.now().hour
```

```
    if hour >= 0 and hour < 12:
```

```
        speak("Hello,Good Morning")
```

```
    elif hour >= 12 and hour < 18:
```

```
        speak("Hello,Good Afternoon")
```

```
    else:
```

```
        speak("Hello,Good Evening")
```

```
def get_audio():
```

```
    r = sr.Recognizer()
```

```
    audio = "
```

```
    with sr.Microphone() as source:
```

```
        print("Listening")
```

```
        playsound("assistant_on.wav")
```

```
        audio = r.listen(source, phrase_time_limit=3)
```

```
        playsound("assistant_off.wav")
```

```
        print("Stop.")
```

```
    try:
```

```
        text = r.recognize_google(audio, language='en-US')
```



```
print('You: ' + ':' + text)
return text
```

```
except:
    speak("I did not understand. Could you please repeat?")
    return "None"
```

```
def note(text):
    date = datetime.datetime.now()
    file_name = str(date).replace(":", "-") + "-note.txt"

    with open(file_name, "w") as f:
        f.write(text)

    subprocess.Popen(["notepad.exe", file_name])
```

```
def date():
    now = datetime.datetime.now()
    month_name = now.month
    day_name = now.day
    month_names = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August',
'September', 'October',
                    'November', 'December']
    ordinalnames = ['1st', '2nd', '3rd', '4th', '5th', '6th', '7th', '8th', '9th', '10th', '11th', '12th',
'13th',
                    '14th', '15th', '16th', '17th', '18th', '19th', '20th', '21st', '22nd', '23rd', '24rd',
'25th',
                    '26th', '27th', '28th', '29th', '30th', '31st']

    speak("Today is " + month_names[month_name - 1] + " " + ordinalnames[day_name -
1] + '.')
```

```
wishMe()
```

```
def Process_audio():
```

```

run = 1
if __name__ == '__main__':
    while run == 1:

        statement = get_audio().lower()
        results = ""
        run += 1

        if "hello" in statement or "hi" in statement:
            wishMe()

        elif "good bye" in statement or "ok bye" in statement or "stop" in statement:
            speak('Your personal assistant ' + name_assistant + ' is shutting down, Good
bye')
            root.destroy()
            break

        elif 'wikipedia' in statement:
            try:

                speak('Searching Wikipedia...')
                statement = statement.replace("wikipedia", "")
                results = wikipedia.summary(statement, sentences=3)
                speak("According to Wikipedia")
                wikipedia_screen(results)
            except:
                speak("Error")

        elif 'joke' in statement:
            speak(pyjokes.get_joke())

        elif 'youtube' in statement:
            webbrowser.open_new_tab("https://www.youtube.com")
            speak("youtube is open now")
            time.sleep(5)

        elif 'google' in statement:
            webbrowser.open_new_tab("https://www.google.com")
            speak("Google chrome is open now")
            time.sleep(5)

```

elif 'gmail' in statement:

```
webbrowser.open_new_tab("mail.google.com")
speak("Google Mail open now")
time.sleep(5)
```

elif 'netflix' in statement:

```
webbrowser.open_new_tab("netflix.com/browse")
speak("Netflix open now")
```

elif 'prime video' in statement:

```
webbrowser.open_new_tab("primevideo.com")
speak("Amazon Prime Video open now")
time.sleep(5)
```

elif 'news' in statement:

```
news =
webbrowser.open_new_tab("https://timesofindia.indiatimes.com/city/mangalore")
speak('Here are some headlines from the Times of India, Happy reading')
time.sleep(6)
```

elif 'cricket' in statement:

```
news = webbrowser.open_new_tab("cricbuzz.com")
speak('This is live news from cricbuzz')
time.sleep(6)
```

elif 'corona' in statement:

```
news =
webbrowser.open_new_tab("https://www.worldometers.info/coronavirus/")
speak('Here are the latest covid-19 numbers')
time.sleep(6)
```

elif 'time' in statement:

```
strTime = datetime.datetime.now().strftime("%H:%M:%S")
speak(f"the time is {strTime}")
```

elif 'date' in statement:

```
date()
```

elif 'who are you' in statement or 'what can you do' in statement:

```
    speak(
        'I am ' + name_assistant + ' your personal assistant. I am programmed to
        minor tasks like opening youtube, google chrome, gmail and search wikipedia etcetra')
```

```
    elif "who made you" in statement or "who created you" in statement or "who
    discovered you" in statement:
```

```
        speak("I was built by Manalee Gedam and Mahima Dwivedi")
```

```
    elif 'note' in statement:
```

```
        statement = statement.replace("make a note", "")
```

```
        note(statement)
```

```
    elif 'note this' in statement:
```

```
        statement = statement.replace("note this", "")
```

```
        note(statement)
```

```
    elif 'command prompt' in statement or 'cmd' in statement:
```

```
        open_cmd()
```

```
    elif 'stack overflow' in statement:
```

```
        speak("Here you go to Stack Over flow. Happy coding")
```

```
        webbrowser.open_new_tab("https://stackoverflow.com/")
```

```
    elif "calculate" in statement:
```

```
        app_id = "3U74V7-Q293UYLW64"
```

```
        client = wolframalpha.Client(app_id)
```

```
        indx = statement.lower().split().index('calculate')
```

```
        statement = statement.split()[indx + 1:]
```

```
        res = client.query(' '.join(statement))
```

```
        answer = next(res.results).text
```

```
        print("The answer is " + answer)
```

```
        speak("The answer is " + answer)
```

```
    elif 'search' in statement or 'play' in statement:
```

```
        statement = statement.replace("search", "")
```

```
        statement = statement.replace("play", "")
```

```
        webbrowser.open(statement)
```

```
    elif 'camera' in statement:
```

```
        open_Camera()
```

elif 'chrome' in statement:
 open_Chrome()

elif 'open adobe reader' in statement:
 open_Adobe_Reader()

elif ' open microsoft edge' in statement:
 open_Microsoft_Edge()

elif 'open clock' in statement:
 open_Clock()

elif 'open sound recorder' in statement:
 open_Sound_recorder()

elif 'open skype' in statement:
 open_Skype()

elif 'open computer management' in statement:
 open_Computer_Management()

elif 'open control panel' in statement:
 open_Control_Panel()

elif 'open bluetooth' in statement:
 open_Bluetooth()

elif 'open paint' in statement:
 open_Paint()

elif 'open on screen keyboard' in statement:
 open_On_Screen_Keyboard()

elif 'open snipping tool' in statement:
 open_Snipping_Tool()

elif 'open task manager' in statement:
 open_Task_manager()

```
elif 'open telegram' in statement:  
    open_Telegram()
```

```
elif 'lock window' in statement:  
    speak("locking the device")  
    ctypes.windll.user32.LockWorkStation()
```

```
elif 'shutdown system' in statement:  
    speak("Hold On a Sec ! Your system is on its way to shut down")  
    subprocess.call('shutdown / p /f')
```

```
elif 'empty recycle bin' in statement:  
    winshell.recycle_bin().empty(confirm=False, show_progress=False,  
    sound=True)  
    speak("Recycle Bin Recycled")
```

```
elif "locate" in statement:  
    speak("What place would you like me to locate?")  
    statement = get_audio()  
    location = statement  
    speak("User asked to Locate")  
    speak(location)  
    url = "https://www.google.com/maps/search/?api=1&query=" + location + ""  
    if location == 'None':  
        speak("I did not understand. Try again later")  
        break  
    else:  
        webbrowser.open(url)
```

```
elif "restart" in statement:  
    subprocess.call(["shutdown", "/r"])
```

```
elif "log off" in statement or "sign out" in statement:  
    speak("Make sure all the application are closed before sign-out. you will be  
given 10 seconds.")  
    time.sleep(10)  
    subprocess.call(["shutdown", "/l"])
```

```
elif "what is" in statement or "who is" in statement:
```

```

client = wolframalpha.Client("3U74V7-Q293"
                             "UYLW64")
res = client.query(statement)
try:
    print(next(res.results).text)
    speak(next(res.results).text)
except StopIteration:
    speak('No results found')
    print("No results")

elif 'ip address' in statement:
    ip_address = ip()
    speak(
        f'Your IP Address is {ip_address}.\n For your convenience, I am printing it
on the screen.')
    print(f'Your IP Address is {ip_address}')

elif "send whatsapp message" in statement:
    speak(
        'On what number should I send the message? Please state the receivers
number: ')
    number = get_audio()
    speak('Please verify if ' + number + ' is correct')
    statement.get_audio()
    if 'no' in statement:
        speak('Oops. Something went wrong. Please try again later.')
        return 'None'
        break
    elif 'yes' or 'ok' in statement:
        speak("What is the message?")
        message = get_audio().lower()
        send_whatsapp_message(number, message)
        speak("I've sent the message.")
    else:
        speak("Sorry. i did not get that")
        break

elif 'advice' in statement:
    speak(f"This is my advice for you.")
    advice = get_random_advice()

```

```

    speak(advice)
    print(advice)

elif 'trending movies' in statement:
    speak(f"Some of the trending movies are: {get_trending_movies()}")
    print(*get_trending_movies(), sep='\n')

elif 'weather' in statement:
    ip_address = ip()
    city = requests.get(f"https://ipapi.co/{ip_address}/city/").text
    speak(f"Getting weather for your city {city}")
    weather, temperature, feels_like = get_weather_report(city)
    speak(f"The current temperature is {temperature}, but it feels like
{feels_like}")
    speak(f"Also, the weather report talks about {weather}")
    print(
        f"Description: {weather}, \n Temperature: {temperature}, but it feels like
{feels_like}")

else:
    speak("I could not understand what you are saying.")
    break

speak(results)

def change_name():
    name_info = name.get()

    file = open("Assistant_name", "w")

    file.write(name_info)

    file.close()

    settings_screen.destroy()

    root.destroy()

```



```

def change_name_window():
    global settings_screen
    global name

    settings_screen = Toplevel(root)
    settings_screen.title("Settings")
    settings_screen.geometry("300x300")
    settings_screen.iconbitmap('app_icon.ico')

    name = StringVar()

    current_label = Label(settings_screen, text="Current name: " + name_assistant)
    current_label.pack()

    enter_label = Label(settings_screen, text="Please enter your Virtual Assistant's name
below")
    enter_label.pack(pady=10)

    Name_label = Label(settings_screen, text="Name")
    Name_label.pack(pady=10)

    name_entry = Entry(settings_screen, textvariable=name)
    name_entry.pack()

    change_name_button = Button(settings_screen, text="Ok", width=10, height=1,
command=change_name)
    change_name_button.pack(pady=10)

def info():
    info_screen = Toplevel(root)
    info_screen.title("Info")
    info_screen.iconbitmap('app_icon.ico')
    info_screen.configure(bg="#101e38")

    creator_label = Label(info_screen, text="Created by Manalee and Mahima", )
    creator_label.pack()

    for_label = Label(info_screen, text="For Final Project", )
    for_label.pack()

```

```
keyboard.add_hotkey("F4", Process_audio)
```

```
def wikipedia_screen(text):
```

```
    wikipedia_screen = Toplevel(root)
```

```
    wikipedia_screen.title(text)
```

```
    wikipedia_screen.iconbitmap('app_icon.ico')
```

```
    wikipedia_screen.configure(bg='#101e38')
```

```
    message = Message(wikipedia_screen, text=text, foreground="#e8b1d1",  
background="white", font=("Helvetica", 12))
```

```
    message.pack()
```

```
def feedback():
```

```
    fb=os.startfile('Feedback_Form.py')
```

```
def main_screen():
```

```
    global root
```

```
    root = Tk()
```

```
    root.title(name_assistant)
```

```
    root.geometry("480x350")
```

```
    root.iconbitmap('app_icon.ico')
```

```
    root.configure(bg='#101e38')
```

```
    name_label = Label(text=name_assistant, width=300, bg='#101e38', fg="white",  
font=("Helvetica", 20))
```

```
    name_label.pack()
```

```
    microphone_photo = PhotoImage(file="microphone.png")
```

```
    microphone_button = Button(image=microphone_photo, command=Process_audio,  
bg="#101e38")
```

```
    microphone_button.pack(pady=10)
```

```
    settings_photo = PhotoImage(file="settings.png")
```

```
    settings_button = Button(image=settings_photo, command=change_name_window,
```

```
bg="#101e38")
    settings_button.pack(pady=10)

    info_button = Button(text="Info", command=info, fg="black", bg="white")
    info_button.pack(pady=10)

    Feedback_button = Button(text="Feedback", command= Feedback_Form.main,
fg="black", bg="white")
    Feedback_button.pack(pady=10)

    root.mainloop()

main_screen()
```

Online Ops.py File:

```
#from main import password
import pywhatkit as kit
import requests
from decouple import config
from wikipedia import wikipedia
import json

NEWS_API_KEY = '5e631d1074d64e45865164b3e2c0c05a'
OPENWEATHER_APP_ID = 'f672955be24df985d3978a3f8f599b05'
TMDB_API_KEY = '1ee888214a2117c629dd9c1cf3e2cdde'

def find_my_ip():
    ip_address = requests.get('https://api64.ipify.org?format=json').json()
    return ip_address["ip"]

def get_latest_news():
    news_headlines = []
    res = requests.get(
        f"https://newsapi.org/v2/top-
        headlines?country=in&apiKey={NEWS_API_KEY}&category=general").json()
    articles = res["articles"]
```

```
for article in articles:
    news_headlines.append(article["title"])
return news_headlines[:5]
```

```
def search_on_wikipedia(query):
    results = wikipedia.summary(query, sentences=2)
    return results
```

```
def play_on_youtube(video):
    kit.playonyt(video)
```

```
def search_on_google(query):
    kit.search(query)
```

```
def send_whatapp_message(number, message):
    kit.sendwhatmsg_instantly(f"+91 {number}", message)
```

```
def get_weather_report(city):
    news_headlines = []
    res = requests.get(
```

```
f"http://api.openweathermap.org/data/2.5/weather?q={city}&appid={OPENWEATHER_
```

```
APP_ID}&units=metric").json()
    weather = res["weather"][0]["main"]
    temperature = res["main"]["temp"]
    feels_like = res["main"]["feels_like"]
    return weather, f"{temperature}°C", f"{feels_like}°C"
```

```
def get_trending_movies():
    trending_movies = []
    res =
requests.get(f"https://api.themoviedb.org/3/trending/movie/day?api_key={TMDB_API_K
EY}").json()
    results = res["results"]
    for r in results:
        trending_movies.append(r["original_title"])
    return trending_movies[:5]
```

```
def get_random_jokes():
    headers = {
        'Accept': 'application/json'
    }
    res = requests.get("https://icanhazdadjoke.com/", headers=headers).json()
    return res["joke"]
```

```
def get_random_advice():
```

```
res = requests.get("https://api.adviceslip.com/advice").json()
return res['slip']['advice']
```

Feedback Form.py

```
import tkinter
from tkinter import *
from tkinter import ttk
from tkinter import messagebox
from PIL import ImageTk
import mysql.connector

#form design

class Feedback:

    #constructor
    def __init__(self, master):
        master.title('IVA FEEDBACK SYSTEM')
        master.resizable(False, False)
        master.configure(background='#101e38')
        # e063bd
        self.style = ttk.Style()
        self.style.configure('TFrame', background='#101e38', foreground='#e063bd')
        self.style.configure('Tbutton', background='#101e38')
        self.style.configure('TLabel', background='#101e38', font=('Helvetica', 11),
        foreground='white')
        self.style.configure('header.TLabel', font=('Helvetica', 18, 'bold'),
        foreground='#e8b1d1', background='#101e38')

        self.frame_header = ttk.Frame(master) # header frame
        self.frame_header.pack()
```

```

self.logo = ImageTk.PhotoImage(file="logo.jpg")
ttk.Label(self.frame_header, image=self.logo).grid(row=0, column=0, rowspan=1)
ttk.Label(self.frame_header, text="Thank you for visiting!",
style='header.TLabel').grid(row=0, column=1)
    #ttk.Label(self.frame_header, text="We're glad you chose to use IVA. \nPlease tell
us how it was!",
        # justify='center', foreground='white', background='#101e38').grid(row=1,
column=1)

self.frame_content = ttk.Frame(master) # content frame
self.frame_content.pack()

    ttk.Label(self.frame_content, text='*Name:', font=('Helvetica', 10),
foreground='#e8b1d1', background='#101e38').grid(row=0, column=0, padx=5,
sticky='sw')
    ttk.Label(self.frame_content, text='*Email(same as earlier):', font=('Helvetica', 10),
foreground='#e8b1d1', background='#101e38').grid(row=0, column=1, padx=5,
sticky='sw')
    ttk.Label(self.frame_content, text='*Comments:', font=('Helvetica', 10),
foreground='#e8b1d1', background='#101e38').grid(row=2, column=0, padx=5,
sticky='sw')

self.entry_name = ttk.Entry(self.frame_content, width=24)
self.entry_email = ttk.Entry(self.frame_content, width=24)
self.text_comments = Text(self.frame_content, width=50, height=10,
font=('Helvetica', 10))

self.entry_name.grid(row=1, column=0, padx=5)
self.entry_email.grid(row=1, column=1, padx=5)
self.text_comments.grid(row=3, column=0, columnspan=2, padx=5)

    ttk.Button(self.frame_content, text='Submit', command=self.submit).grid(row=5,
column=0, padx=6, sticky='e')
    ttk.Button(self.frame_content, text='Clear', command=self.clear).grid(row=5,
column=1, padx=6, sticky='w')

#Data handling

def submit(self):
    if self.entry_name.get() == "" or self.text_comments.get("1.0",'end-1c') == "" or

```



```

self.entry_email.get() == "":
    messagebox.showerror("Error", "All fields are required")
else:
    conn = mysql.connector.connect(host="localhost", user="root",
password="Mahima123$", database="via")
    my_cursor = conn.cursor(buffered=True)
    query = "select * from feedback where user_email=%s"
    value = (self.entry_email.get(),)
    my_cursor.execute(query, value)
    row = my_cursor.fetchone()
    my_cursor.execute("insert into feedback values(%s,%s,%s)", (
        self.entry_name.get(),
        self.text_comments.get("1.0",'end-1c'),
        self.entry_email.get(),
    ))
    conn.commit()
    conn.close()
    messagebox.showinfo(title='Submit', message='Thank you for your Feedback!
Your Comments are Submitted!')
    print('Name: {}'.format(self.entry_name.get()))
    print('Email: {}'.format(self.entry_email.get()))
    print('Comments: {}'.format(self.text_comments.get(1.0, 'end')))
    self.clear()

#clear button
def clear(self):
    self.entry_name.delete(0, 'end')
    self.entry_email.delete(0, 'end')
    self.text_comments.delete(1.0, 'end')

#calling function
def main():
    root = tkinter.Toplevel()
    feedback = Feedback(root)
    root.mainloop()

if __name__ == "__main__": main()

```

os_ops.py:

```
import os
import subprocess as sp
import pyautogui
from pyautogui import screenshot

paths = {
    'notepad': r"C:\Windows\notepad.exe",
    'Chrome': r"C:\Program Files (x86)\Google\Chrome\Application\chrome.exe",
    '#cmd': r"C:\Windows\System32\cmd.exe",
    'PyCharm': r"C:\ProgramData\Microsoft\Windows\Start
Menu\Programs\JetBrains\PyCharm Community Edition 2021.3.2.lnk",
    'Spotify': r"C:\Program
Files\WindowsApps\SpotifyAB.SpotifyMusic_1.180.699.0_x86__zpdnekdrzrea0\Spotify.
exe",
    'Adobe_Reader': r"C:\Program Files (x86)\Adobe\Reader 11.0\Reader\AcroRd32.exe",
    'Microsoft_Edge': r"C:\Program Files (x86)\Microsoft\Edge\Application\msedge.exe",
    'Calculator': r"C:\Windows\System32\calc.exe",
    'Clock': r"C:\Program Files (x86)\FreeAlarmClock\FreeAlarmClock.exe",
    '#Camera': r"C:\Program
Files\WindowsApps\Microsoft.WindowsCamera_2021.105.10.0_x64__8wekyb3d8bbwe\
WindowsCamera.exe",
    'Sound_recorder': r"C:\Program Files\Audacity\Audacity.exe",
    'Video_Launcher': r"C:\Program
Files\WindowsApps\Microsoft.Windows.Photos_2021.21090.10008.0_x64__8wekyb3d8
bbwe\VideoProjectsLauncher.exe",
    'Skype': r"C:\Program
Files\WindowsApps\Microsoft.SkypeApp_15.80.194.0_x86__kzf8qxf38zg5c\Skype\Sky
pe.exe",
    'Computer_Management': r"C:\Windows\System32\CompMgmtLauncher.exe",
    'Control_Panel': r"C:\Windows\System32\control.exe",
    'Bluetooth': r"C:\Windows\System32\ffsquirr.exe",
    'Paint': r"C:\Windows\System32\mspaint.exe",
    'On_Screen_Keyboard': r"C:\Windows\System32\osk.exe",
    'Snipping_Tool': r"C:\Windows\System32\SnippingTool.exe",
    'Task_manager': r"C:\Windows\System32\Taskmgr.exe",
    'Telegram': r"C:\Users\ASUS\AppData\Roaming\Telegram Desktop\Telegram.exe"
```

```
}

def open_notepad():
    os.startfile(paths['notepad'])

def open_Chrome():
    os.startfile(paths['Chrome'])

def open_cmd():
    os.system('start cmd')

def open_PyCharm():
    os.startfile(paths['PyCharm'])

def open_Spotify():
    os.startfile(paths['Spotify'])

def open_Adobe_Reader():
    os.startfile(paths['Adobe_Reader'])

def open_Microsoft_Edge():
    os.startfile(paths['Microsoft_Edge'])

def open_Calculator():
    os.startfile(paths['calculator'])

def open_Clock():
    sp.Popen(paths['Clock'])

def open_Camera():
    sp.run('start microsoft.windows.camera:', shell=True)

def open_Sound_recorder():
    os.startfile(paths['Sound_recorder'])

#def open_Video_Launcher():
    # sp.startfile(paths['Video_Launcher'])
```

```

def open_Skype():
    os.startfile(paths['Skype'])

def open_Computer_Management():
    os.startfile(paths['Computer_Management'])

def open_Control_Panel():
    os.startfile(paths['Control_Panel'])

def open_Bluetooth():
    os.startfile(paths['Bluetooth'])

def open_Paint():
    os.startfile(paths['Paint'])

def open_On_Screen_Keyboard():
    os.startfile(paths['On_Screen_Keyboard'])

def open_Snipping_Tool():
    os.startfile(paths['Snipping_Tool'])

def open_Task_manager():
    os.startfile(paths['Task_manager'])

def open_Telegram():
    os.startfile(paths['Telegram'])

class note:
    def Note(self,data):
        date=datetime.datetime.now()
        filename=str(date).replace(':', '-')+'-note.txt'
        a=os.getcwd()
        if not os.path.exists('Notes'):
            os.mkdir('Notes')
        os.chdir(a+r'\Notes')
        with open(filename,'w') as f:
            f.write(data)
        sp.Popen(['notepad.exe',filename])
        os.chdir(a)

```

```

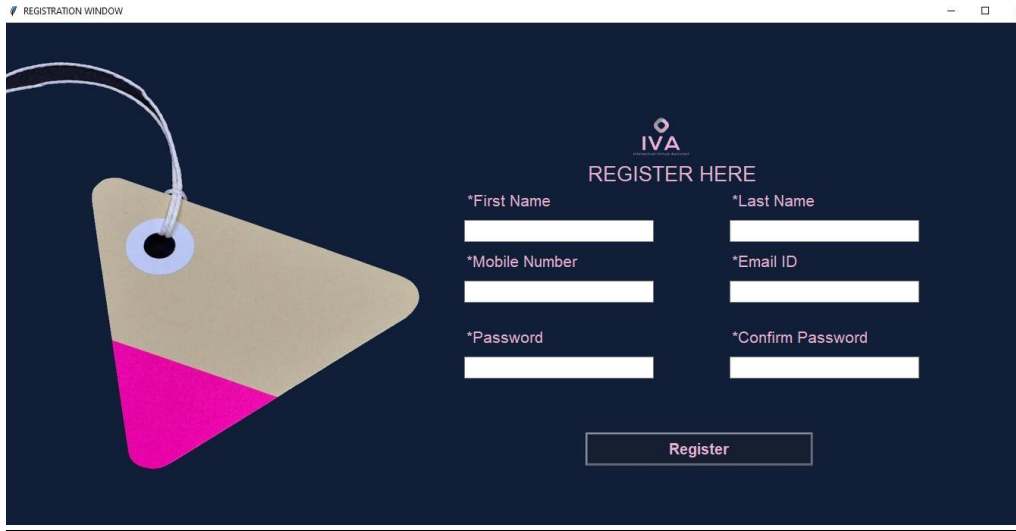
#class screenshot:
# def takeSS(self):
#     #img_captured = pyautogui.screenshot
#     #a = os.getcwd()
#     #if not os.path.exists("Screenshots"):
#     #     # os.mkdir("Screenshots")
#     #os.chdir(a + "\Screenshots')
#     #ImageName = 'screenshot-' + str(datetime.datetime.now()).replace(':', '-') + '.png'
#     #img_captured.save(ImageName)
#     #os.startfile(ImageName)
#     #os.chdir(a)

#class camera:
# @property
# #def takePhoto(self):
# # self.videoCaptureObject = cv2.VideoCapture(0)
# # self.result = True
# # a=os.getcwd()
# #if not os.path.exists("Camera"):
# #     # os.mkdir("Camera")
# #os.chdir(a+"\Camera')
# #self.ImageName="Image-"+str(datetime.datetime.now()).replace(':', '-')+".jpg"
# #while(self.result):
# #     # self.ret,self.frame = self.videoCaptureObject.read()
# #     # cv2.imwrite(self.ImageName,self.frame)
# #     # self.result = False
# #self.videoCaptureObject.release()
# #cv2.destroyAllWindows()
# #os.chdir(a)
# #playsound.playsound("camera-shutter-click.mp3")
# #return "Camera\\"+self.ImageName

```

Input/output screen

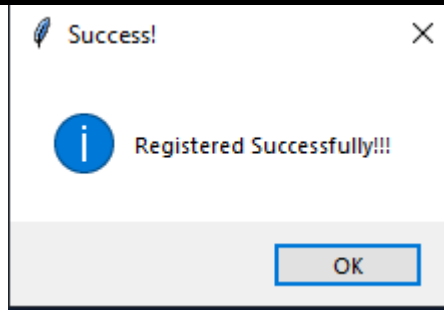
Registration Form output



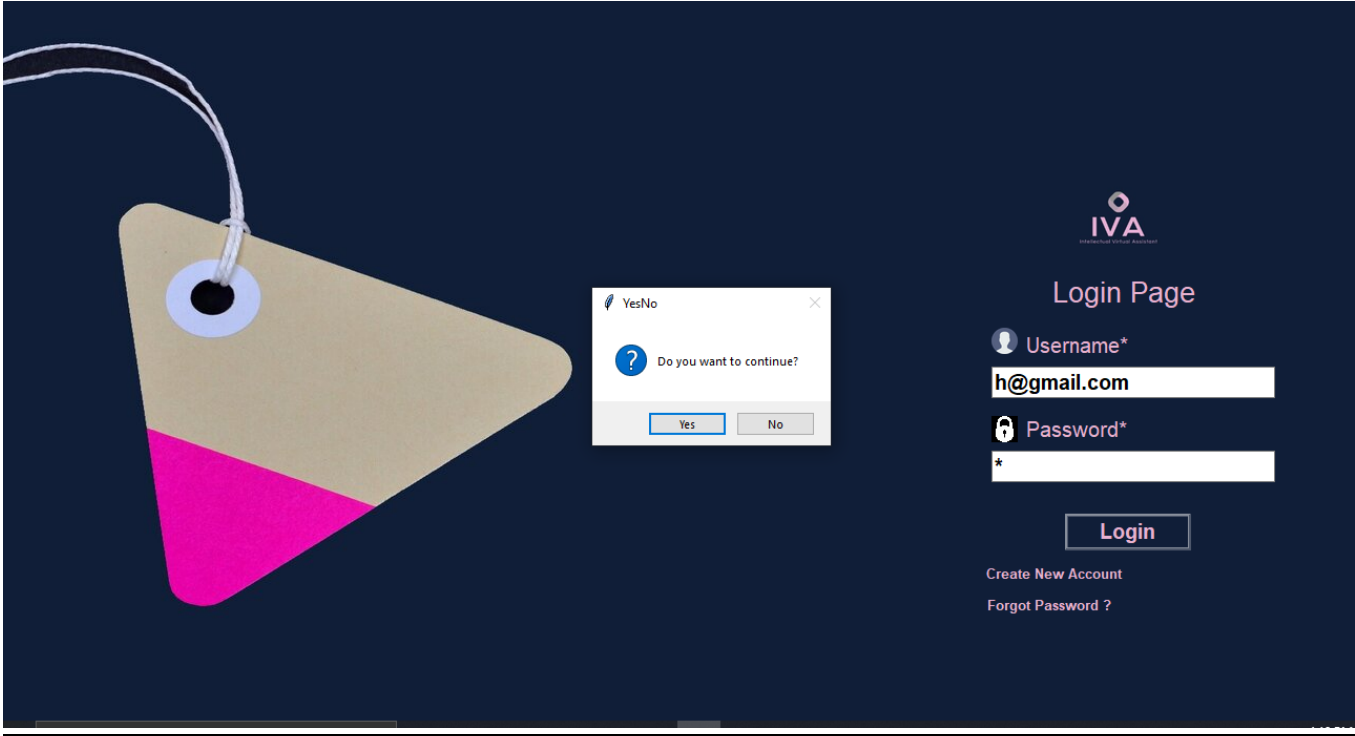
The screenshot shows a registration form window with a dark blue background. On the left, there is a graphic of a brown paper tag with a white string and a hole, and a pink triangle at the bottom. The form is titled "REGISTER HERE" and includes the following fields:

- *First Name
- *Last Name
- *Mobile Number
- *Email ID
- *Password
- *Confirm Password

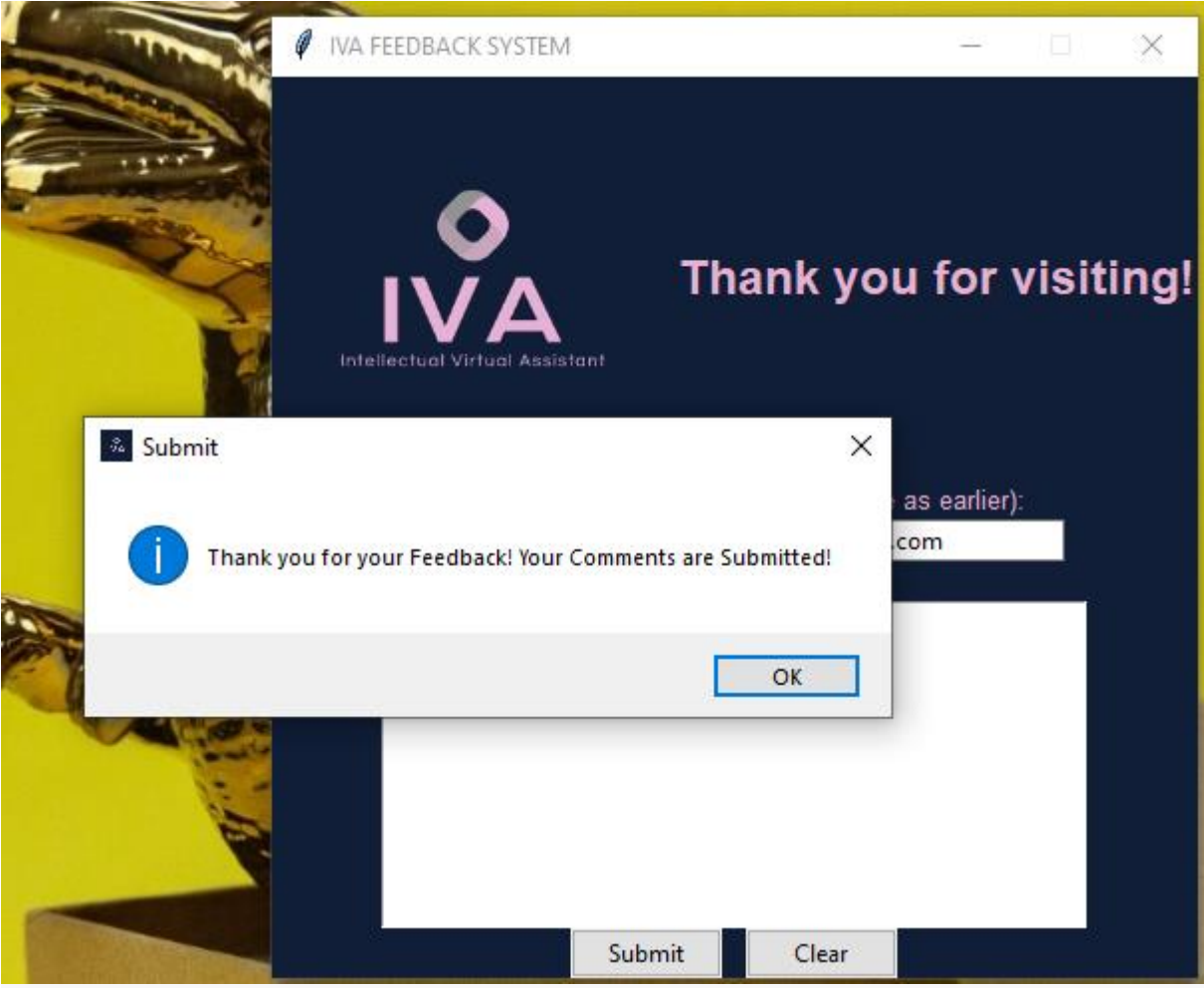
A "Register" button is located at the bottom center of the form.



Login Form Output

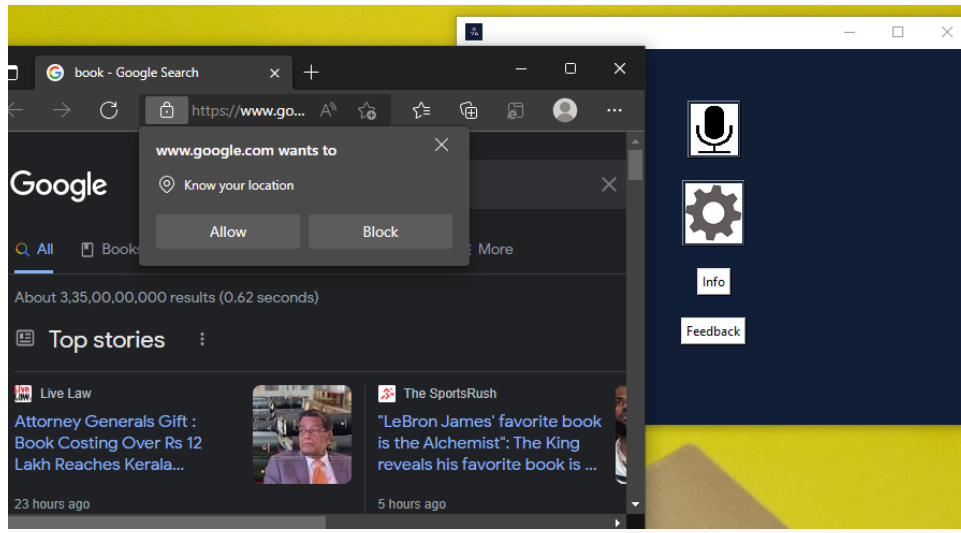


Feedback Form Output

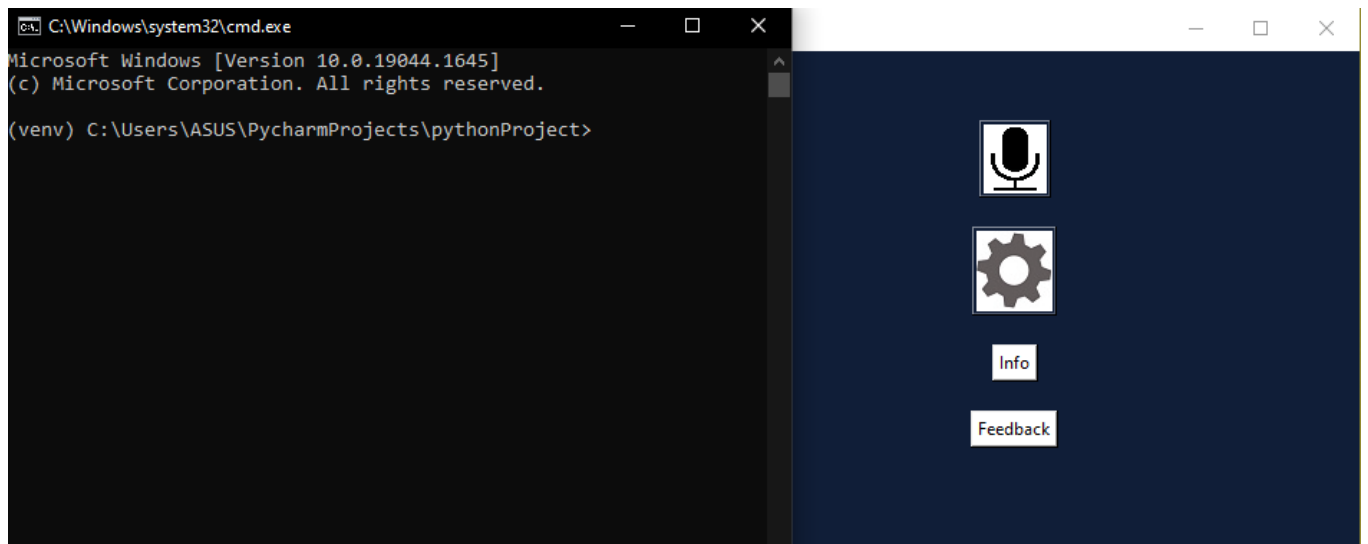


Output screen of IVA functionalities:

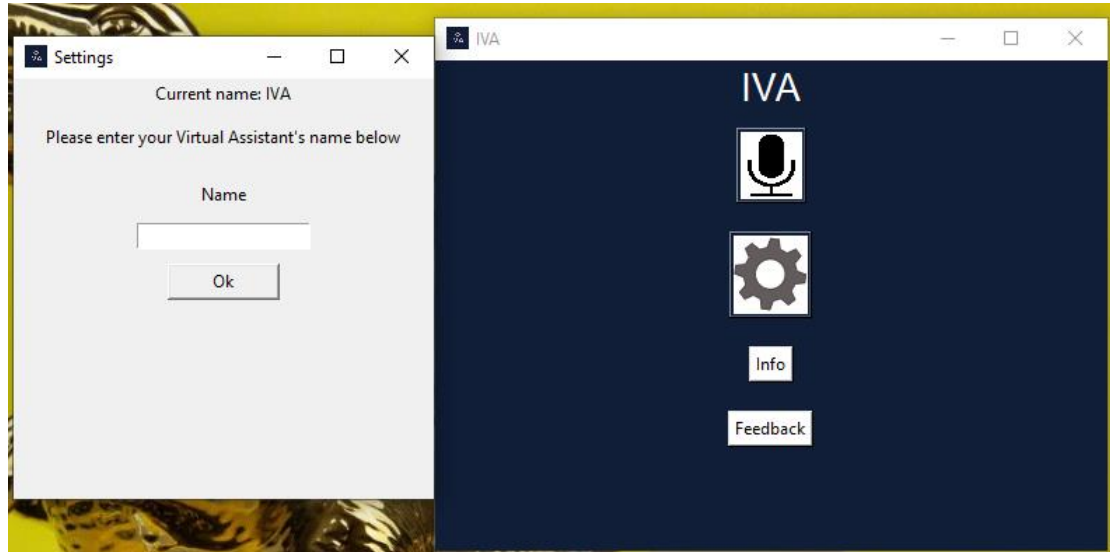
Search function:



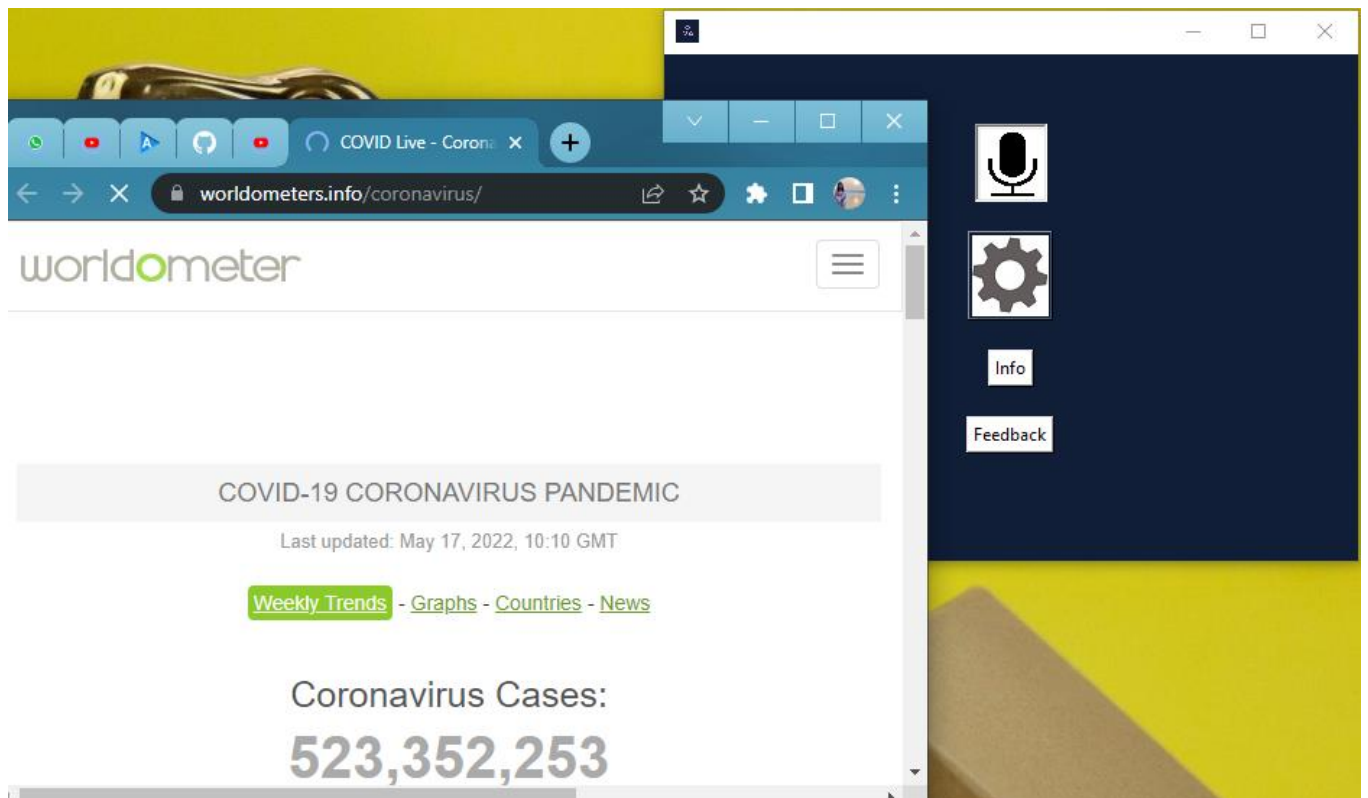
Opening command prompt:



Change name screen:



Showing latest covid_19 numbers:



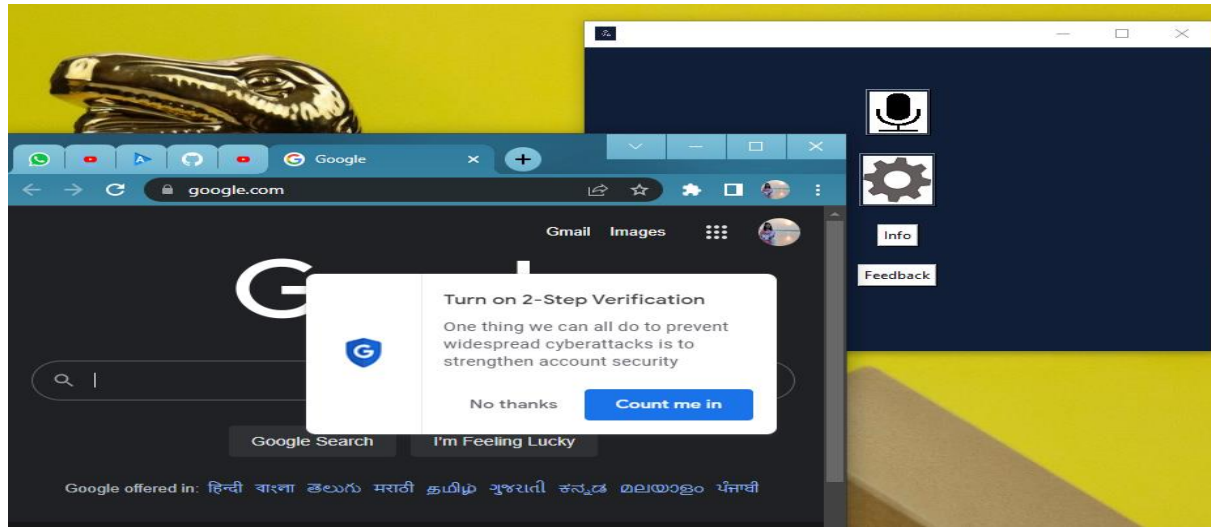
Showing cricket news:

The screenshot shows the Cricbuzz website in a browser window. The browser's address bar displays "https://www.cri...". The website features a header with the Cricbuzz logo and navigation links for "Live Scores", "Schedule", "Archives", "News", "Series", "Teams", and "Videos". Below the header, there is a "MATCHES" section with links for "SL vs BAN - Live", "DC vs PBKS - DC Won", "MI vs SRH - Preview", and "RR vs LSG - RR W". A "FEATURED MATCHES" section is visible, listing scores for SL (397), BAN (272-3), DC (159-7 (20)), and PBKS (142-9 (20)). It also mentions "Mumbai Indians", "Sunrisers Hyde", and "Tuesday, 17 May". A sidebar on the right contains icons for a microphone, settings, "Info", and "Feedback".

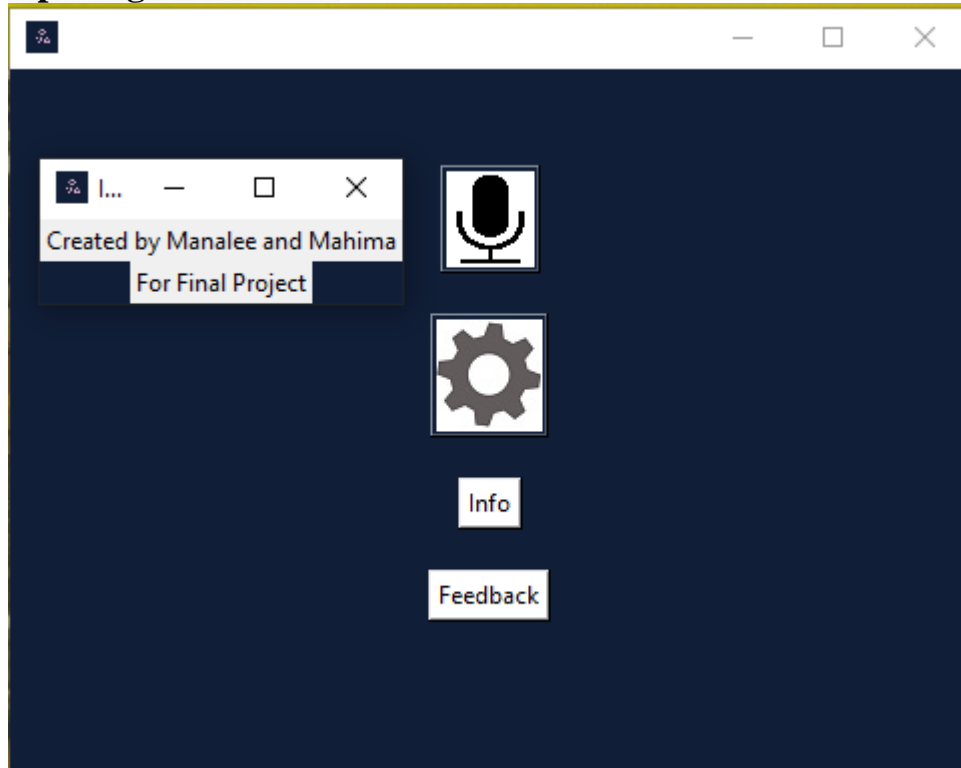
Opening gmail:

The screenshot shows the Gmail website landing page in a browser window. The browser's address bar displays "https://www.go...". The Gmail logo is visible, along with the text "For work" and buttons for "Sign in" and "Get Gmail". The main heading reads "Secure, smart, and easy to use email". Below this, it says "Get more done with Gmail. Now integrated with Google Chat, Google Meet, and more, all in one place." At the bottom, there is a cookie notice: "This site uses cookies from Google to deliver its services and to analyze traffic." with buttons for "Learn more" and "OK, Got it". A sidebar on the right contains icons for a microphone, settings, "Info", and "Feedback".

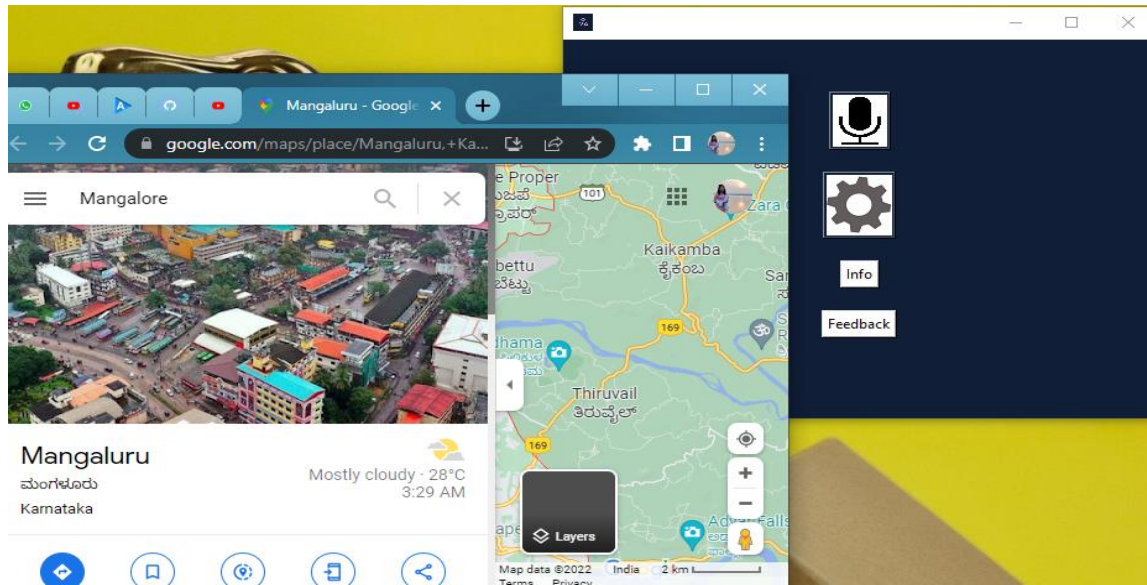
Opening google:



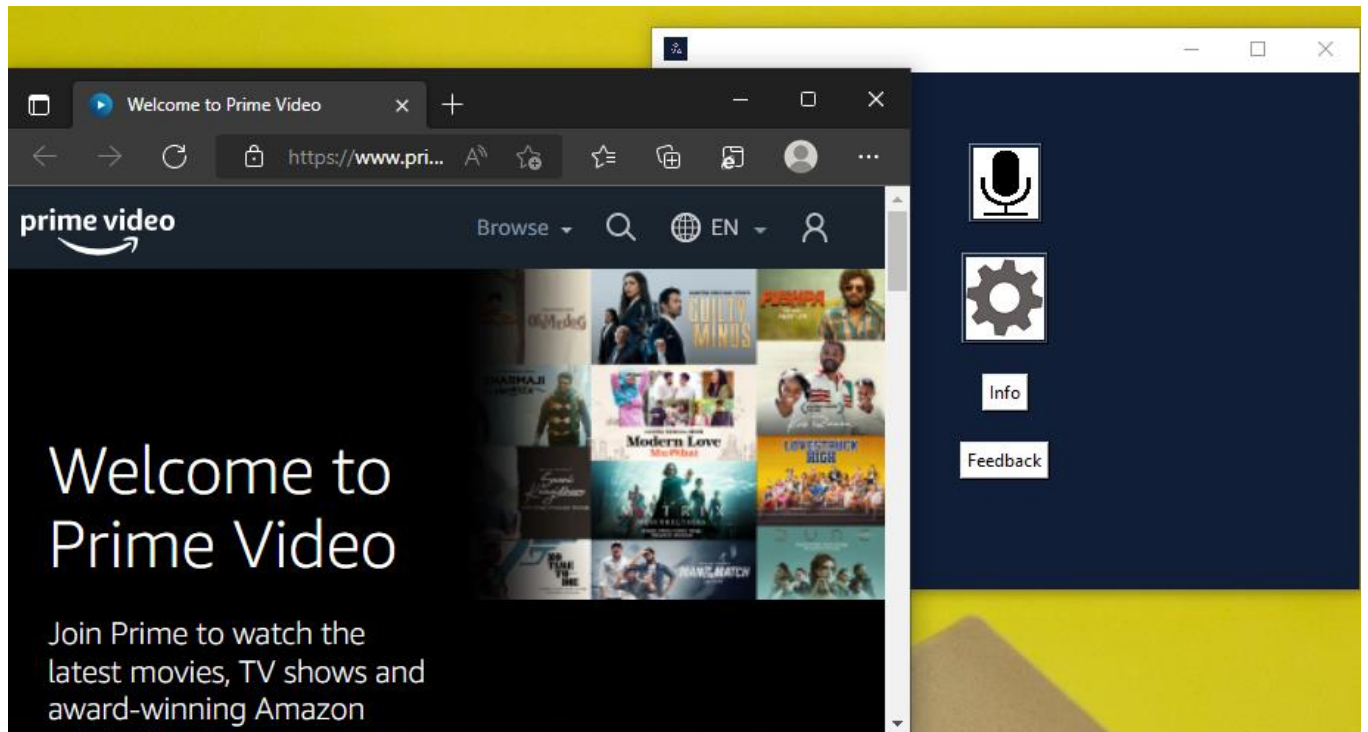
Opening info screen:



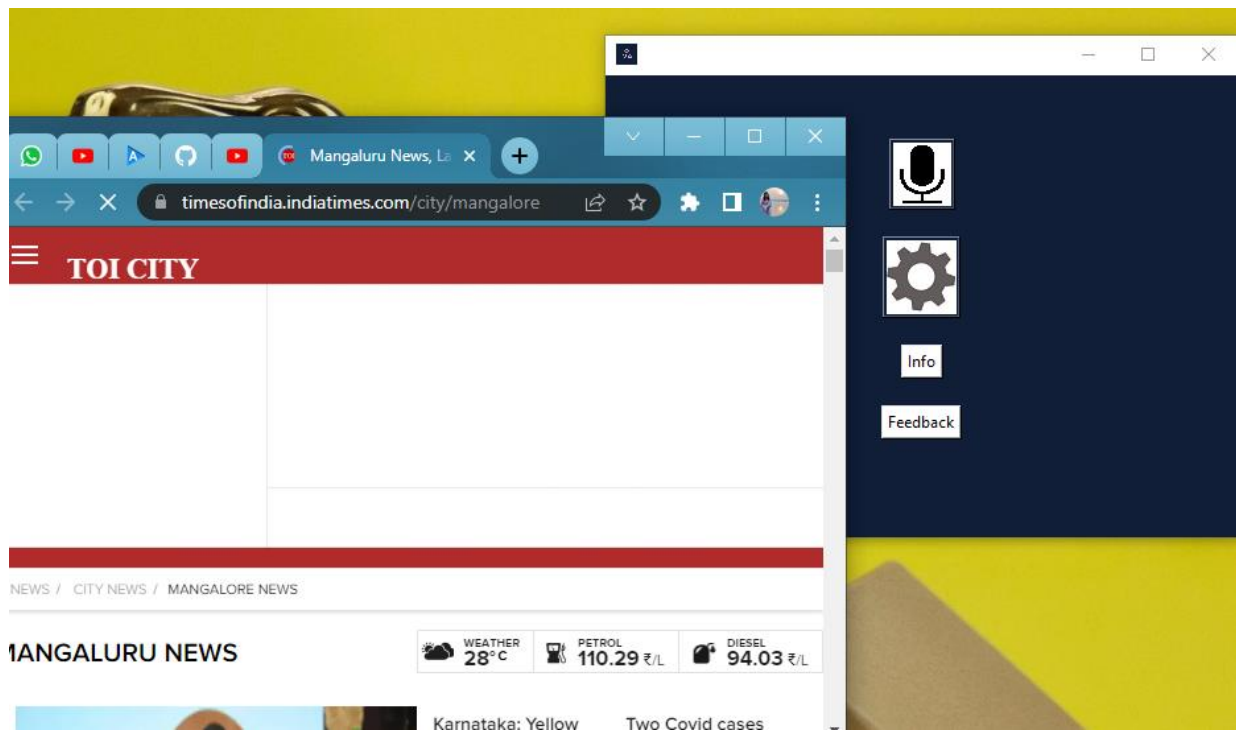
Locate function:



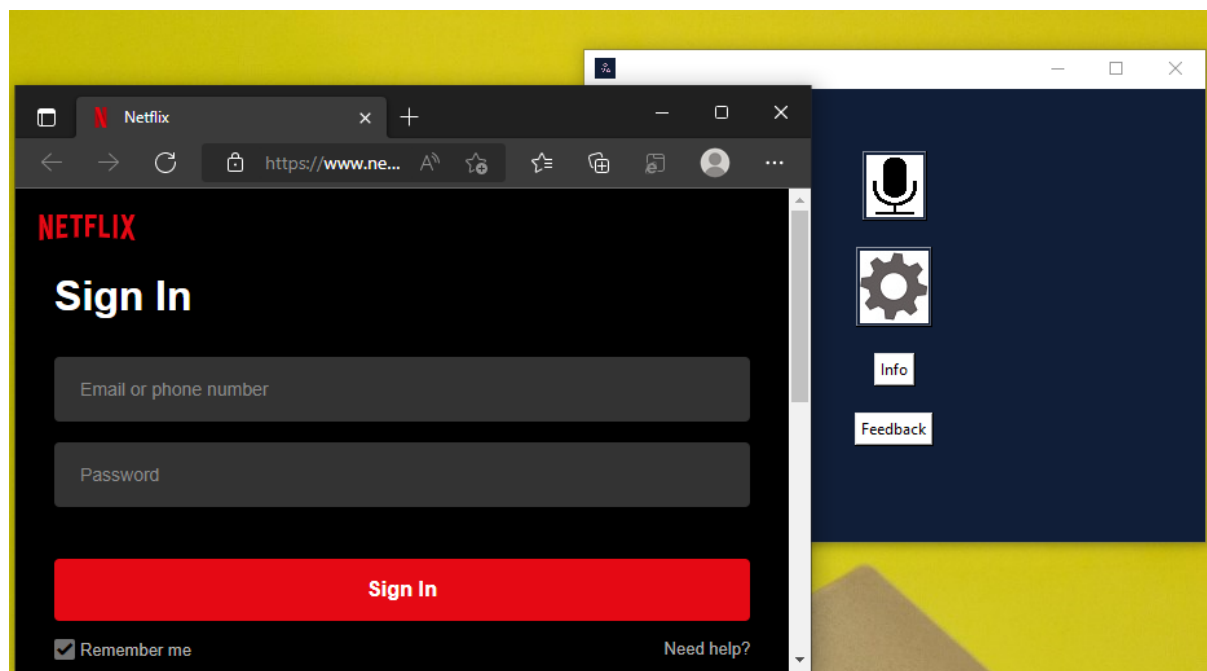
Opening amazon prime video:



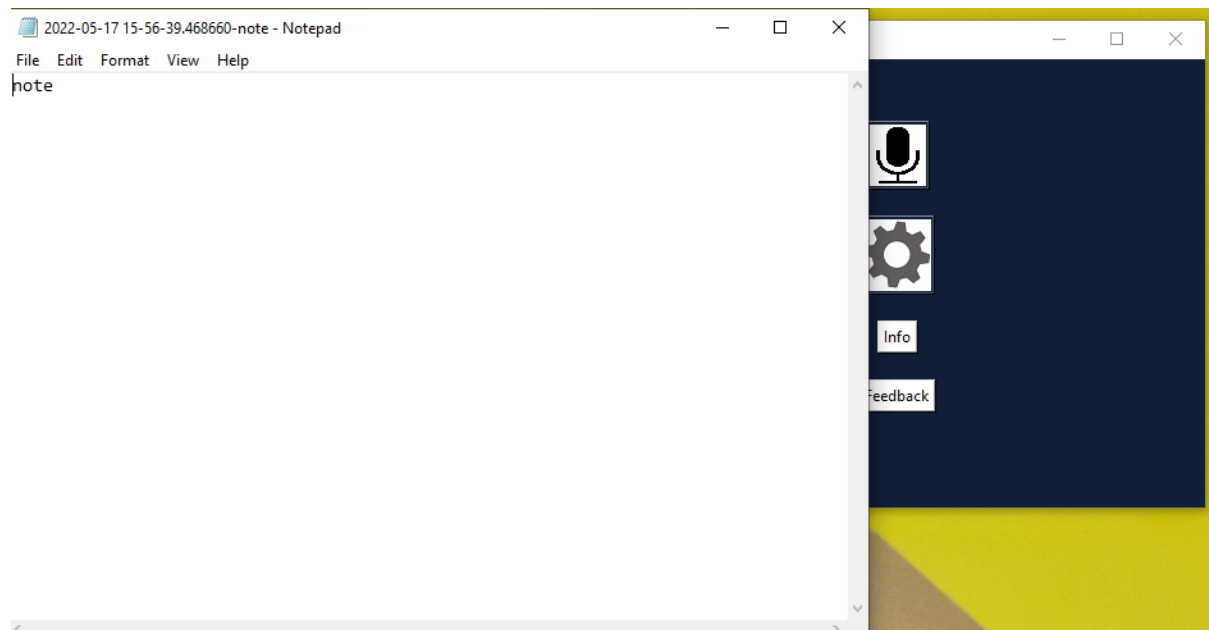
Opening news:



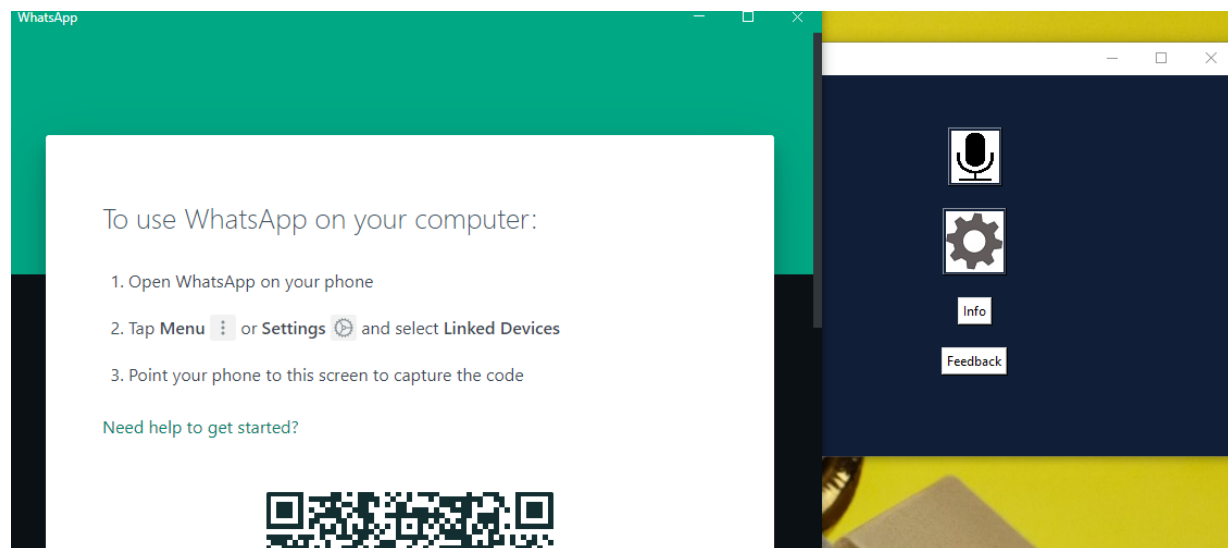
Opening netflix:



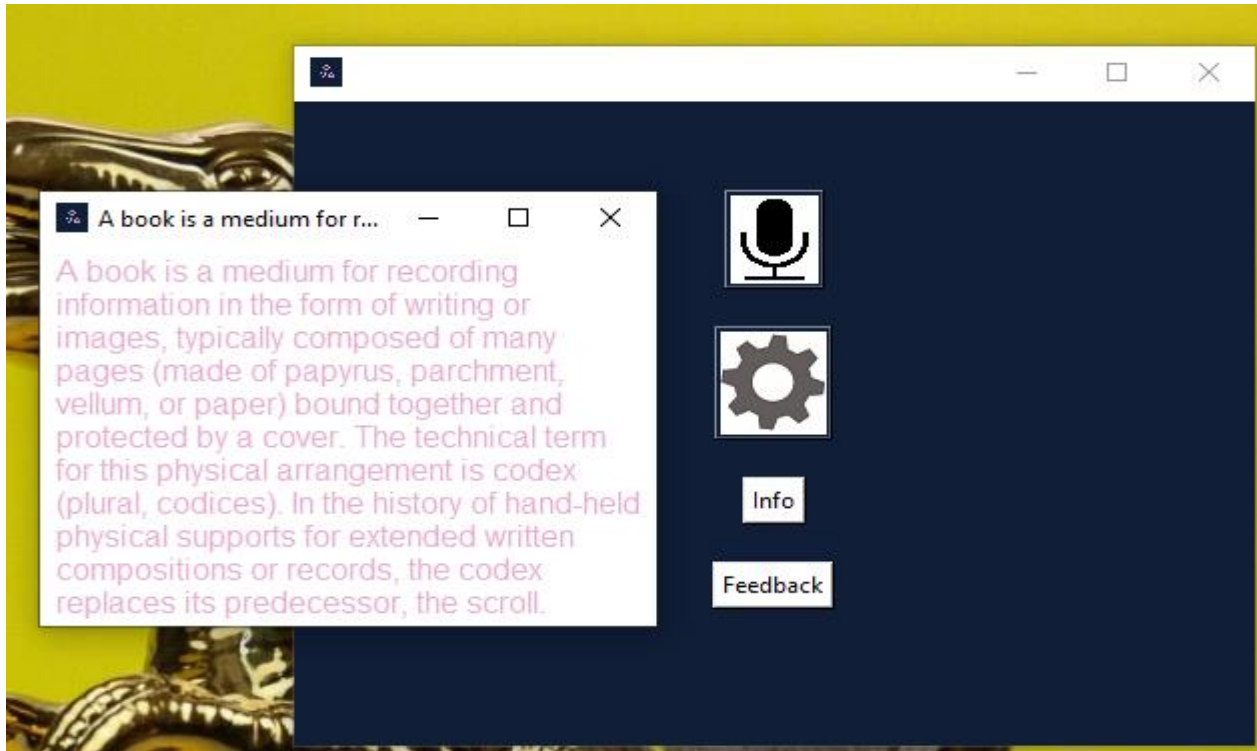
Creating a note:



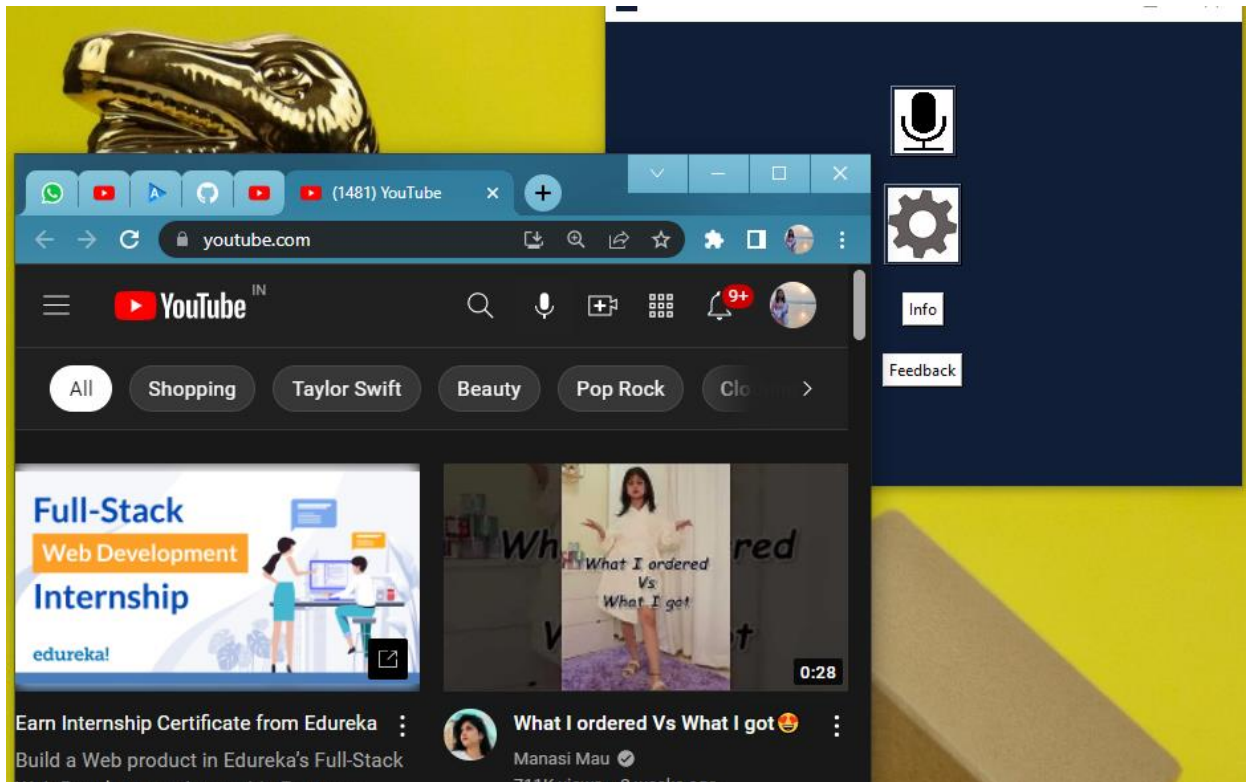
Sending a whatsapp message:



Wikipedia Query:



Opening youtube:



TESTING AND **VALIDATION CHECK**

Testing and validation check:

Validation testing in software engineering is in place to determine if the existing system complies with the system requirements and performs the dedicated functions for which it is designed along with meeting the goals and needs of the organization.

The process of evaluating software during the development process or at the end of the development process to determine whether it satisfies specified business requirements.

Validation Testing ensures that the product actually meets the client's needs. It can also be defined as to demonstrate that the product fulfills its intended use when deployed on appropriate environment.

Whenever any particular software is tested then the main motive is to check the quality against the defects being found.

The developers fix the bugs and the software is rechecked to make sure that absolutely no bugs are left out in that. This not only shoots the product's quality but also its user acceptance.

- To ensure customer satisfaction
- To be confident about the product
- To fulfil the client's requirement until the optimum capacity
- Software acceptance from the end-user

Client-side validation is an initial check and an important feature of good user experience; by catching invalid data on the client-side, the user can fix it straight away. If it gets to the server and is then rejected, a noticeable delay is caused by a round trip to the server and then back to the client-side to tell the user to fix their data.

TEST CASES:

- **Test Case 1**

Test Title: Response Time

Test ID: T1

Test Priority: High

Test Objective: To make sure that the system respond back time is efficient.

Description:

Time is very critical in a voice based system. As we are not typing inputs, we are speaking them. The system must also reply in a moment. User must get instant response of the query made.

- **Test Case 2**

Test Title: Accuracy

Test ID: T2

Test Priority: High

Test Objective: To assure that answers retrieved by system are accurate as per gathered data.

Description:

A virtual assistant system is mainly used to get precise answers to any question asked. Getting answer in a moment is of no use if the answer is not correct. Accuracy is of utmost importance in a virtual assistant system.

- **Test Case 3**

Test Title: Approximation

Test ID: t3

Test priority: Moderate

Test Objective: To check approximate answers about calculations.

Description:

There are times when mathematical calculation requires approximate value. For example, if someone asks for value of PI the system must respond with approximate value and not the accurate value. Getting exact value in such cases is undesirable.

- **Test Case 4**

Test Title: Validation of Input Fields and Buttons.

Test ID: t4

Test Priority: High

Test Objective: Validation and testing of the input fields and buttons embedded in the web forms.

Description:

The forms used for login and Sign-In page must be validated so that users can only enter appropriate data in the given input fields and can submit or cancel their data as per their need.

- **Test Case 5**

Test Title: Database Connectivity

Test ID: t5

Test Priority: Medium

Test Objective: To test the connectivity between database and application.

Description:

The data entered by users must be inserted in the database as per its nature and the assigned column. The information must be stored securely and as per the schema of the database as intended by the developer.

SYSTEM
SECURITY
MEASURES

System security measures:

Security of a computer system is a crucial task. It is a process of ensuring confidentiality.

A system is said to be secure if its resources are used and accessed as intended under all the circumstances, but no system can guarantee absolute security from several of the various threats and unauthorized access.

Security System Goals –

Henceforth, based on the above breaches, the following security goals are aimed:

- **INTEGRITY:**

The objects in the system mustn't be accessed by any unauthorized user & any user not having sufficient rights should not be allowed to modify the important system files and resources.

- **SECURITY:**

The objects of the system must be accessible only to a limited number of authorized users. Not everyone should be able to view the system files.

- **AVAILABILITY:**

All the resources of the system must be accessible to all the authorized users' i.e. only one user/process should not have the right to hog all the system resources. If such kind of situation occurs, denial of service could happen. In this kind of situation, malware might hog the resources for itself & thus preventing the legitimate processes from accessing the system resources.

Security measures taken:

- **Strong passwords:**

This first measure is taken that users may use special characters in their passwords and password length must be 8 characters.

- **Confidentiality:**

If any users is sharing their personal details in login form it will be secure safely as only users can access such information.

IMPLEMENTATION,
EVOLUTION
AND
MAINTENANCE

IMPLEMENTATION, EVALUATION AND MAINTENANCE:

IMPLEMENTATION:

The software implementation stage involves the transformation of the software technical data package (TDP) into one or more fabricated, integrated, and tested software configuration items that are ready for software acceptance testing. Software implementation refers to the process of adopting and integrating a software application into a business or an individual's workflow (as a part of a digital transformation strategy). The primary activities of software implementation include the:

- Fabrication of software units to satisfy structural unit specifications.
- Assembly, integration, and testing of software components into a software configuration item.
- Prototyping challenging software components to resolve implementation risks or establish a fabrication proof of concept.
- Dry-run acceptance testing procedures to ensure that the procedures are properly delineated and that the software product (software configuration items (CIs and computing environment) is ready for acceptance testing.

The implementation process includes planning, organizing, developing procedures, training the users, developing forms for data collection, developing files for storage of data, testing the system, cutover, and

documenting the system. The process is further explained as such:

- **Organizing for Implementation:**

In this stage, the responsibilities regarding the implementation of software are assigned to the respective persons for efficiency and quick as well as concise and clear implementation of software.

- **Developing Procedures for Implementation:**

Proper procedure should be established for the implementation of the system developed. These procedure help guide the developer(s) through the process of installation and implementation. Procedures should be established for the following purposes:

- Evaluation and selection of hardware
- Purchase or development of software
- Implementation of MIS in parts

- **Hardware and Software Acquisition:**

Acquisition of hardware and software has to be made for the physical installation of networks. Hardware includes computers and other peripherals, while software includes the operating system, applications, and programming languages. Another major managerial decision involved in this stage is whether to buy the hardware or lease it from the vendors. Capital expenditure analysis, prestige, usage, anticipated replacement schedule, and vendor's options are some of the factors that are taken into consideration to decide whether to buy the hardware or lease it.

- **Develop Forms for Data Collection:**

Organizations generate or collect a vast amount of data from both internal and external sources. This data must be stored in standardized formats. To do so, daily activities should be recorded in specifically designed forms. Even in organizations where sophisticated technology is used for recording of data, some forms need to be used. These may be formats in which data is stored. Forms are used not only for input/output of data but also for transfer of data between processes. However, in such small-scale projects, one must collect data on a basic level or create own, small datasets or data items.

E.g.: Creating dummy database entries to observe if the software functions as needed.

- **Changeover:**

Changeover is the point at which the new system replaces the old system. This entails activities like the physical transfer of files, furniture, and other office equipment and also the movement of people. Even after component and subsystem testing, problems cannot be ruled out.

During this period, forms and programs may have to be changed and some employees may be transferred. It is at this stage that the practical realities of the system are reflected. These cannot be found out during the pre-acceptance testing.

- **Documenting the System:**

Documentation is the process of maintaining written reports describing the scope, purpose, information flow, and operating procedures of the system. It is required when there is need for troubleshooting, replacement of the subsystem, training operating personnel, and upgrading the system. Maintaining documents helps organizations in several ways:

- New personnel can be trained easily with the help of documents.
- Designers who are not familiar with the system can refer to the documents to understand the system.
- Designers who are not familiar with the system can refer to the documents to understand the system.
- Designers who are not familiar with the system can refer to the documents to understand the system.

EVALUATION:

After the implementation phase, another stage in project development is Evaluation. Evaluation during a program's implementation may examine whether the program is successfully recruiting and retaining its intended participants, using training materials that meet standards for accuracy and clarity, maintaining its projected timelines, coordinating efficiently with other ongoing programs and activities, and meeting applicable legal standards. Evaluation during program implementation could be used to inform mid-course corrections to program implementation (formative evaluation) or to shed light on implementation processes (process evaluation). After keeping the project in the working condition for the sometime, all the errors that are showing in the computer program should be removed. The programmer needs to correct them so that same errors should not be repeated. We should also get the feedback from the user which are using it and ask them whether, it is user friendly or not. After evaluating the Program and satisfying the needs of the user the program is maintained fully.

It has been evaluated that the developed IVA software is the basic in its functionalities and has a huge scope of development in the future as well.

MAINTENANCE:

Maintenance of software can include software upgrades, repairs, and fixes of the software if it breaks.

Software applications often need to be upgraded or integrated with new systems the customer deploys. It's often necessary to provide additional testing of the software or version upgrades. During the maintenance phase, errors or defects may exist, which would require repairs during additional testing of the software. Monitoring the performance of the software is also included during the maintenance phase.

Once the system is deployed, and customers start using the developed system, following 3 activities occur:

- **Bug fixing** – bugs are reported because of some scenarios which aren't tested at all.
- **Upgrade** – Upgrading the application to the newer versions of the software.
- **Enhancement** – Adding some new features into the existing software.

In the case of IVA, it's very basic in its functionalities. Hence, it has many avenues available for its development. One of the bugs available is that it relies on a few third-party software for clock and other basic processes. This is a bug that can be solved at a later time.

Suggested updates would be addition of an activation phrase, Addition of other functionalities such as being able to create new folders in systems, sending out distress signals to relevant authorities in case of emergencies, voice recognition for added security, etc.

**FUTURE SCOPE
OF THE PROJECT**

FUTURE SCOPE OF THE PROJECT:

- **Removing Dependency upon third-party application:**

The current project is still somewhat reliant upon third party open source applications normally found on desktops. However, in the future, such reliance can be completely eradicated, making VIA self-sufficient.

- **Activation Phrase:**

In the present state of the project, the activation has not been implemented. However, in the future, a specific phrase such as the phrases used in Google Assistant and SIRI can be implemented in this technology as well.\

- **Improved Efficiency:**

The efficiency is good in the present application, but it can be improved using web-scraping in the future updates as well.

- **Voice Recognition Software:**

In the future, voice recognition functionality using AI and Machine learning can be added to provide exclusive access to particular users and personalization for users as well.

- **Better Security:**

In the future with the utilization of firewall, biometric verification, etc. can be used for increasing the security of the personal data provided by the users when using the application.

- **Global Reach:**

In the future, this application can be launched globally. For this reason, independent platforms that can support VIA can be developed as well.

These are some of the functionalities and features we can add to the application to increase its reach and to improve the application as a whole to be more convenient for the users and others.

CONCLUSION

CONCLUSION:

While working on this project, we were able to understand and appreciate the many inner functionalities and delicate brilliance of the languages used in this project. It was an incredible learning opportunity that helped us understand some of the inner workings that come with developing a project.

Our project is to create a virtual assistant that will help making the daily organization tasks faced by people a little easier as well as help them multitask. This project also helps us to understand complicated modules such as pywhatkit, pyaudio, tensorflow, etc. This helped us improve somewhat as aspiring developers ourselves.

We were clear as to what we wanted to achieve, but a little unclear on the execution of it. For this, we had to rely on references and official documentation.

The project is very basic in its nature and can be improved in the future. The current project is primarily help users in basic tasks and to help us (developer's) advance academically and logically in the field of software development.

All the procedures and the above stated technical elements are a culmination of 3 years of education provided by our teachers and mentors.

BIBLIOGRAPHY

BIBLIOGRAPHY –

While developing this project internet was the eternal support.

Following are the websites referred by us which helped us in developing our project:

- academia.edu
- YouTube.com
- TechRepublic.com
- <https://docs.python.org/3/>
- <https://www.php.net/docs.php>
- <https://dev.w3.org/>
- <https://devdocs.io/>
- <https://dev.mysql.com/doc/>
- <https://vixenlabs.co/voice-consumer-index>
- Stackoverflow.com
- google.co.in
- Lucidchart.com
- Go.gliPHY.com
- GeeksForGeeks.com

