

PROJECT REPORT

Voice Driven Web Application for Desktop



Project Group Number: III-16

Faculty Mentor Name: Dr. Anirban Goswami

Alumni Mentor Name: Soumyadip Sen

Group Members Name and Roll Numbers:

1. Sanu Kumar (13000219043)
2. Amartya Shivam (13000219023)
3. Raj Shekhar Pal (13000219056)
4. Krishna Kumar (13000219020)

PROBLEM STATEMENT

The idea is to implement a voice driven web application. As per the provided problem statement, we need to develop an application which will be like a browser on which we have to open websites like Google, YouTube, Facebook, etc using voice commands. Now the navigations inside this website have to be also voice-driven. If I want to click on some button on the website, I must be able to do it just by voice commands.

According to Google research, 57% of US teens and 43% of adults use voice search at least once a day. Most of all, voice search is preferred during cooking, watching TV, and other such situations, when it is difficult to type text in the usual way, and additional effort is required.

From our study, we made our plan of action in which first we will make an environment where the whole project will work on, which will be on a browser. The next step will be taking the voice as input and converting them into text. This text will act as a command and each specific command will do their specific work on that page. The commands will be used to open several websites. The navigation inside that website will be done by targeting their unique ID's in the web page source code.

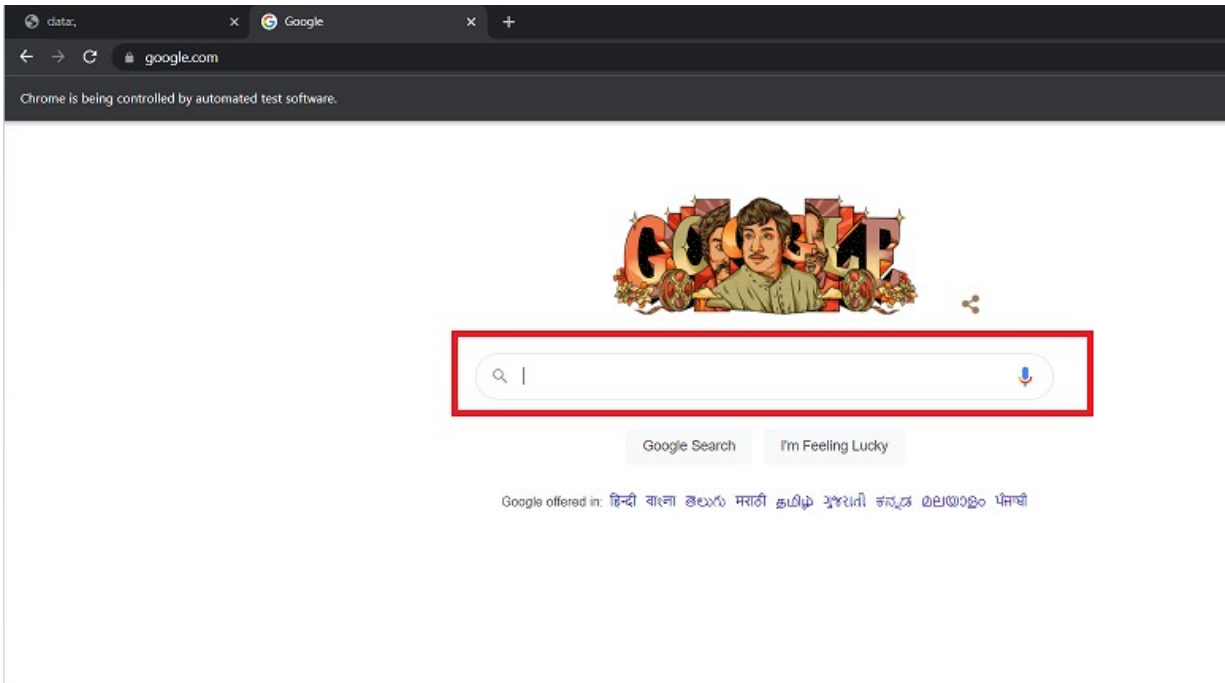
METHODOLOGY

We have divided the work into 5 following parts:

- 1) **Building the environment:** - We will be using Selenium Web driver to open a browser on which we will be navigating websites using voice commands.
- 2) **Speech- to- text:** - Using speech recognition we will be collecting the input in voice form and converting them to text form. This text will be used to find the work the user has to do in the website.
- 3) **Creating a sample webpage:** - We will be also making a form filling webpage using HTML and CSS as a sample. The form will be filled by voice commands given by the user.
- 4) **Navigation:** - We will be navigating websites like Facebook, Google, YouTube, etc. We will be accessing the things on the website by extracting their unique ID's from the web page source. We will be assigning some commands which will target these ID's upon which things will be controlled on the web page.
- 5) **Text-to-Speech:-** To make the system more effective we will be creating a voice bot using pyttsx3. This will take inputs in text form and give output in form of speech. This will help us to know whether our given commands are valid or not and will also speak what are we doing in the web page.

EXPERIMENTAL RESULTS

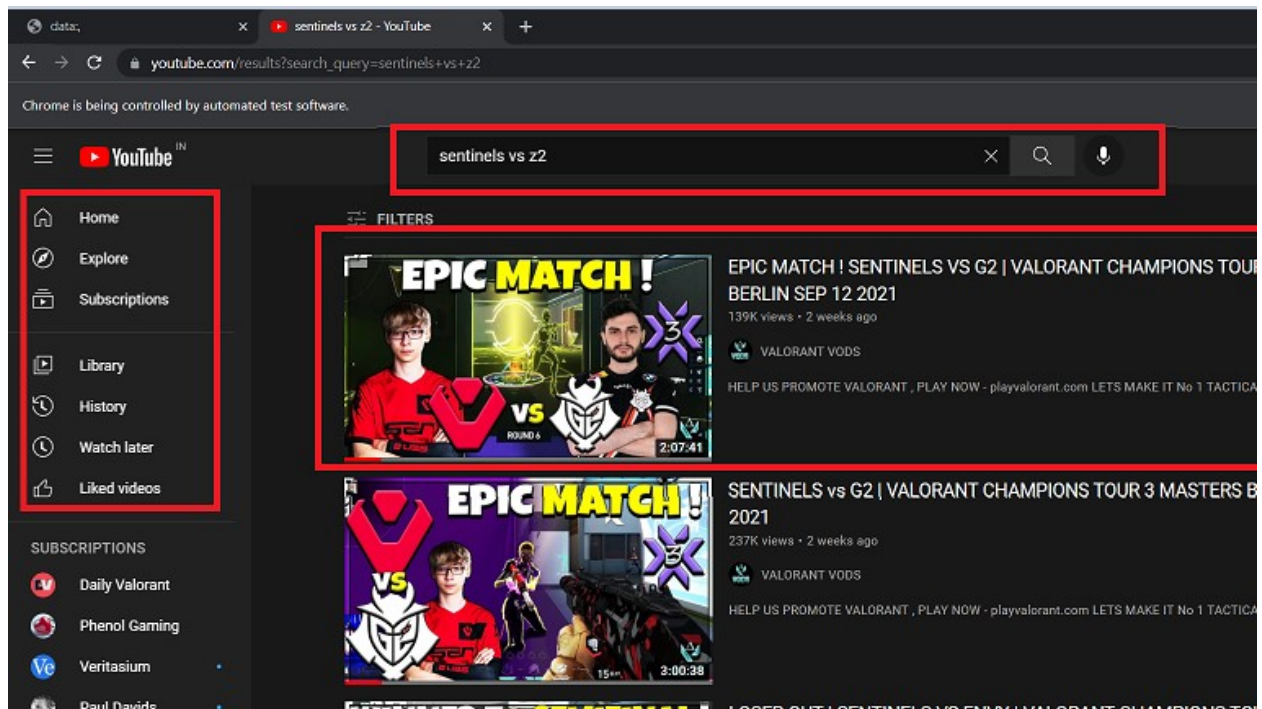
The things which will be controlled by voice are depicted in red boxes:-



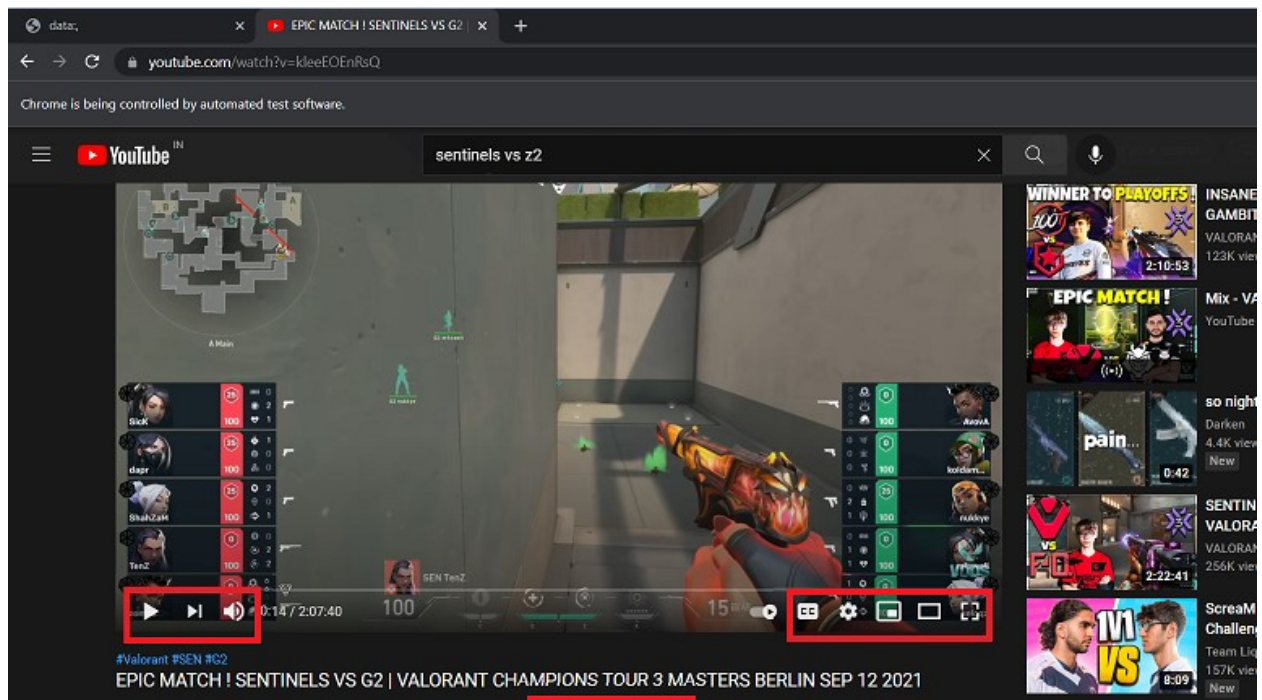
Google Search

A screenshot of the Google sign-in page. It features the Google logo, the text 'Sign in', and 'Use your Google Account'. A red rectangular box highlights the input field for 'Email or phone', which contains a vertical cursor. Below this field is a link for 'Forgot email?'. Further down, there is a link for 'Not your computer? Use Guest mode to sign in private' and a 'Learn more' link. At the bottom, there is a 'Create account' link and a partially visible 'No' button.

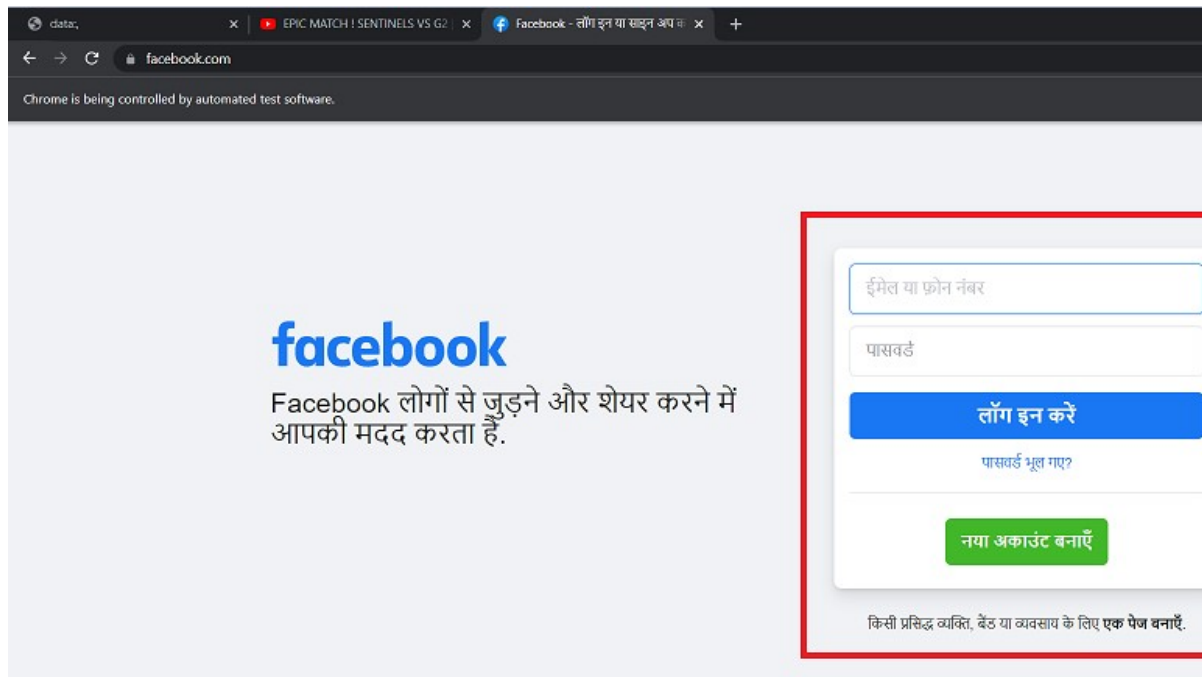
Login Google & YouTube



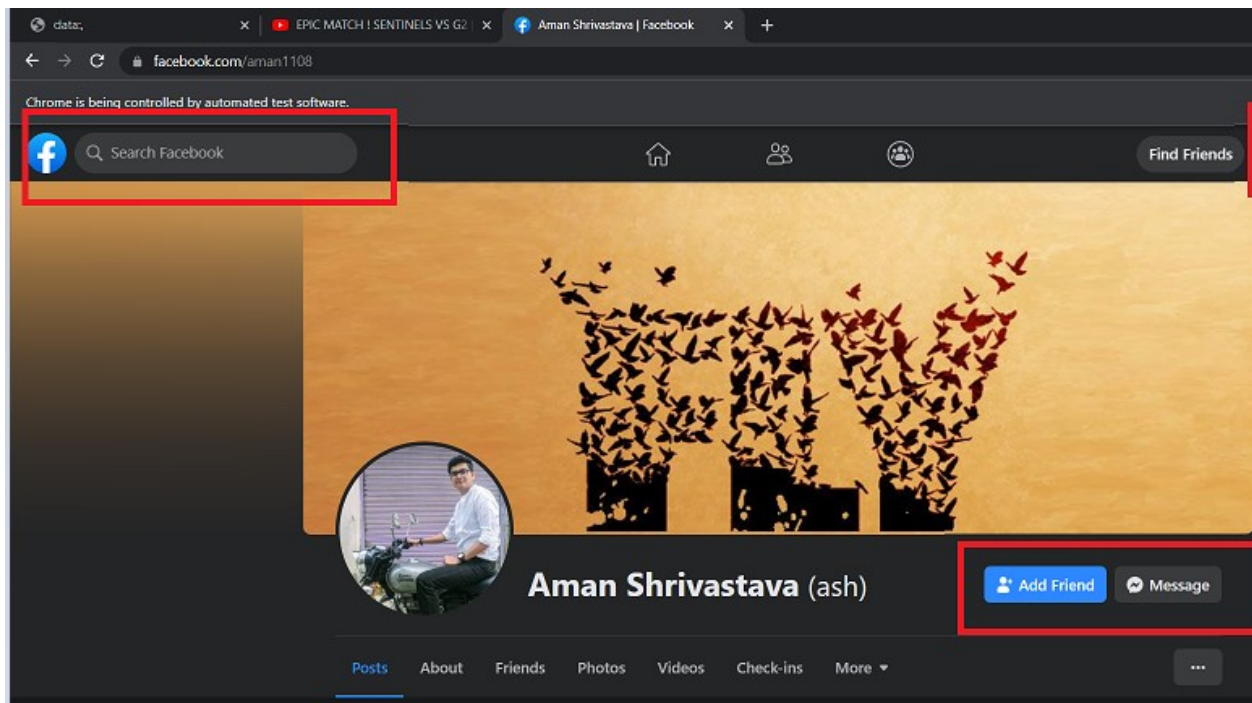
Playing Videos, Accessing the left panel of YouTube
Searching Videos, Log-Out, Log-In



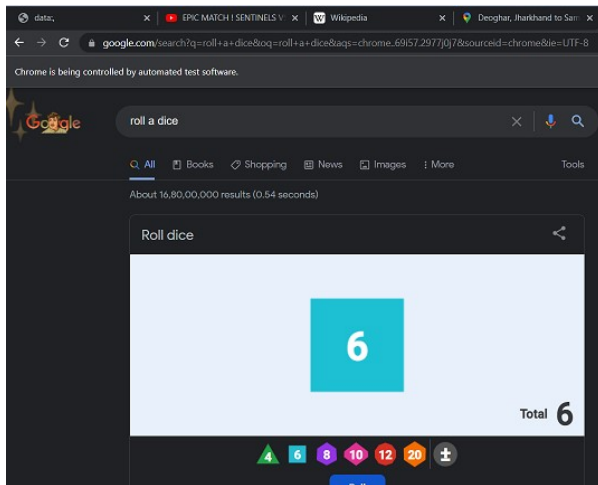
Play, Pause, Mute, Un-mute, Next Video, Theatre Mode, Full Screen,
Like, Dislike, Subscribe



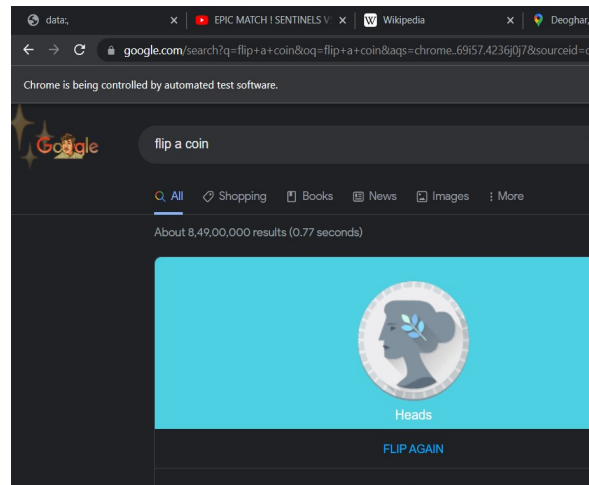
Logging into Facebook



Searching for friends, View Profile, Show Notifications, Open Messenger, Dark-Mode, Light-Mode, Add Friend, Message, Login, Logout



Rolling a Dice



Flipping a Coin

A screenshot of a web browser displaying a "Student Registration Form". The form is overlaid on a background image of a mountain landscape. The form fields include: "Name" (split into "First Name" and "Last Name"), "College", "Email", "Phone" (split into "Area Code" and "Phone Number"), and "Subject" (a dropdown menu with "--Choose option--"). A red rectangular box highlights the form area. The browser's address bar shows the URL "127.0.0.1:5500/project-site.html".

Filling a form (Self made web-page)

More functions:- Refreshing a page, Changing voice of voice assistant, Switch Tabs, Go forward or back, Take a break.

SCOPE OF IMPROVEMENT

There are some areas in which there is scope of improvement like adding more working parts of a website into commands. The efficiency of recognizing voice commands can also be improved. Setting more commands for the voice assistant and the responses can be made more accurate. More websites can be added into the project and voice navigating them. Adding various other languages to the project which will help to navigate the websites in any language a user want.

ARTIFACTS

Repo link: <https://github.com/SanuKumar4245/Voice-Driven-Web-Application.com>

User Manual:

1. First we have to run the python main.py file to open the web browser.
2. The voice assistant will ask for the command. Following are the commands which will help to open websites and navigate through them:-

❖ **Google:-**

- **Open Google:** It will open Google home page in new tab.
- **Login Google:** You can login Google using your ID and Password.
- **Search Google:** You can use the Google search to search anything on Google.

❖ **YouTube:-**

- **Open YouTube:** It will open YouTube home page in new tab.
- **Login YouTube:** You can login YouTube using your mail id and password.
- **Search YouTube:** You can search for any videos in YouTube.
- **YouTube Home:** - You will be redirected to YouTube home page.
- **Explore:** - To go to explore section of YouTube.
- **Subscriptions:** - To go to subscriptions page in YouTube.
- **Library:** - To go to the library section of YouTube.
- **Play the first video:** - upon searching some videos it will play the first video from the below list.
- **Pause the video:** - Pause the playing video.
- **Play the video:** - Resume the paused video.

- **Mute:** - mute the playing video.
- **Un-mute:-** un-mute the muted video.
- **Full screen:** - Play the video in full screen mode or exit from it.
- **Theatre mode:** - Play the video in theatre mode.
- **Exit theatre mode:** - exit from the theatre mode.
- **Like the video:** - Give a like to the playing video.
- **Unlike the video:** - Dislike the playing video.
- **Turn captions:** - turn captions on or off.
- **Next video:** - plays the next video in queue.
- **Subscribe:** - subscribe the channel.
- **Unsubscribe:** - unsubscribe the channel.

❖ **Facebook:-**

- **Open facebook:** - It will open facebook login page in new tab.
- **Login facebook:** - Login facebook using your id and password.
- **Search facebook:** - search for a friend in facebook.
- **Add friend:** - sends friend request to the person whose profile is open, also can cancel the friend request.
- **Message:** - opens the messenger to send message to the person whose profile is open.
- **Show profile:** - opens your profile.
- **Open messenger:** - opens the facebook messenger.
- **Show notifications:** - shows your facebook notifications.
- **Dark mode:** - turns the dark mode on for facebook.
- **Light mode:** - return to light mode of facebook from dark mode.
- **Logout:-** logout from facebook

❖ **Wikipedia:-**

- **Open Wikipedia:** - Wikipedia's home page will be opened in new tab.
- **Search Wikipedia:** - Search anything in Wikipedia.

❖ **Google Maps:-**

- **Open Google map:** - opens Google maps in new tab.
- **Find path:** - it will ask for the source and destination from the user upon which it will show a path between those places.

❖ **Form Filling Web Page:-**

- **Open form:** - A web page will open which will contain a student registration form.
- **Fill:** - the voice assistant will ask the respective columns details from the user and will fill the columns with user commands.

❖ **More Functions:-**

- **Roll dice:** - a new tab will open where a dice will be rolled and the outcome can be seen.
- **Flip coin:** - a new tab will open where a coin will be flipped and the outcome can be seen.
- **Switch tab:** - helps to switch between multiple tabs present in the browser.
- **Go back:** - go back to the previous page .
- **Go forward:** - go to the page where we came back from.
- **Refresh:** - refreshes the page.
- **Change voice:** - changes the voice assistant voice from male to female and vice versa.
- **Take a break:** - if you want the assistant to take a break you can write the amount of time in the console and the assistant will sleep during that duration.
- **Exit:** - closes the browser.

REFERENCES

- Selenium Web-Driver:
<https://www.selenium.dev/documentation/webdriver.com>
- Pyttsx3 (Text to Speech):
<https://pypi.org/project/pyttsx3.com>
- Speech_recognition(Speech to Text):
<https://www.geeksforgeeks.org/speech-recognition-in-python-using-google-speech-api.com>
- Web page making:
https://www.w3schools.com/html/html_css.asp.com