### A Micro Project report on

PERFORM AUTOMATION TESTING FOR VOICE BASED INPUT IN GOOGLE SEARCH ENGINE

Submitted to the CMR Institute of Technology in partial fulfillment of the requirement for the award of the Laboratory of

## AUTOMATED TESTING TOOL LAB

**Of**

## II B.Tech II Semester In

**Computer Science and Engineering (AI&ML)**

### Submitted by

|  |  |
| --- | --- |
| Y.RAJA REDDY | 22R01A66C9 |
| J.VASANTHI | 23R05A6607 |
| K.PRANAY | 23R05A6608 |
| K.SHIVA | 23R05A6609 |
| K.SRIKAR | 23R05A6610 |
| M.ADITHYA | 23R05A6611 |
| M.RAKESH | 23R05A6612 |

Under the Guidance of

**Mrs. K. Jyothi**

**(Assistant Professor, Dept. of CSE(AIML))**



**CMR INSTITUTE OF TECHNOLOGY (UGC AUTONOMUS)**

**(Approved by AICTE, Affiliated to JNTU, Kukatpally, Hyderabad) Kandlakoya, Medchal Road, Hyderabad.**2021-2022

## CMR Institute of Technology

**(UGC AUTONOMOUS)**

**(Approved by AICTE, Affiliated to JNTU, Kukatpally, Hyderabad) Kandlakoya, Medchal Road, Hyderabad.**



**CERTIFICATE**

This is to certify that a Micro Project entitled with: “Perform automation testing on Voice based search engine on Google Search Engine”.

Submitted By

|  |  |
| --- | --- |
| Y.RAJA REDDY | 22R01A66C9 |
| J.VASANTHI | 23R05A6607 |
| K.PRANAY | 23R05A6608 |
| K.SHIVA | 23R05A6609 |
| K.SRIKAR | 23R05A6610 |
| M.ADITHYA  M.RAKESH | 23R05A6611  23R05A6612 |

In partial fulfillment of the requirement for award of the Automated Testing Tool Laboratory of II B. Tech II Semester in CSE(AI&ML) to the CMRIT, Hyderabad is a record of a bonafide work carried out under our guidance and supervision.

|  |  |  |
| --- | --- | --- |
| **Signature of Faculty** | **Signature of Course coordinator** | **Signature of HOD** |
| **Mrs. K. Jyothi** | **Mrs. V. Vijaya** | **Prof .P. Pavan Kumar** |
| **(Assistant Professor)** | **(Assistant Professor)** |  |

# ACKNOWLEDGEMENT

We are extremely grateful to **Dr. M. Janga Reddy**, **Director**, **Dr. B. Satyanarayana**, **Principal** and **Prof. P. Pavan Kumar**, **Head of Department**, Dept of Computer Science and Engineering(AI&ML), CMR Institute of Technology for their inspiration and valuable guidance during entire duration.

We are extremely thankful to our Automated Testing Tool faculty in-charge **Mrs. K. Jyothi**, Dept of Computer Science and Engineering, CMR Institute of Technology for his constant guidance, encouragement and moral support throughout the project.

We express our thanks to all staff members and friends for all the help and coordination extended in bringing out this Project successfully in time.

Finally, we are very much thankful to our parents and relatives who guided directly or indirectly for successful completion of the project.

|  |  |
| --- | --- |
| Y.RAJA REDDY | 22R01A66C9 |
| J.VASANTHI | 23R05A6607 |
| K.PRANAY | 23R05A6608 |
| K.SHIVA | 23R05A6609 |
| K.SRIKAR | 23R05A6610 |
| M.ADITHYA  M. RAKESH | 23R05A6611  23R05A6612 |

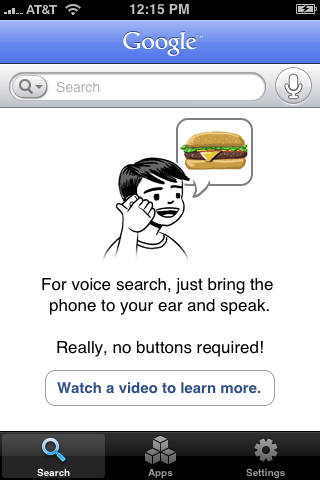
## CONTENTS

|  |  |
| --- | --- |
| Particulars | Page No. |
| 1. Introduction | 5-6 |
| 2. Algorithm | 7 |
| 3. Requirements(S/W and H/W) | 8 |
| 4. Implementation | 9-13 |
| 5. Results | 14 |
| 6. Conclusion | 15 |
| 7. References | 16 |

1. **INTRODUCTION:**

2.3 Google Search by Voice:

Mobile web search is a rapidly growing area of interest. Internet-enabled smartphones account for an increasing share of the mobile devices sold throughout the world, and most models offer a web browsing experience that rivals desktop computers in display quality. Users are increasingly turning to their mobile devices when doing web searches, driving efforts to enhance the usability of web search on these devices. 5 Although mobile device usability has improved, typing search queries can still be cumbersome, error-prone, and even dangerous in some usage scenarios.

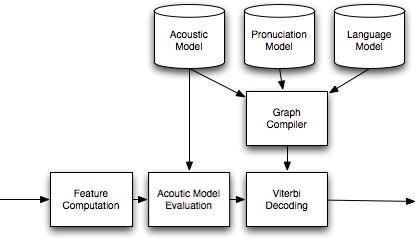


Google search by Voice for iPhone. In November 2008 we introduced Google Mobile App (GMA) for iPhone (Figure 4) that included a search by voice feature. GMA search by voice extended the paradigm of multi-modal voice search from searching for businesses on maps to searching the entire world wide web. In the next few sections we discuss the technology behind these efforts and some lessons we have learned by analyzing data from our users.

The goal of Google search by Voice is to recognize any spoken search query. Table 1 lists some example queries, hinting at the great diversity of inputs we must accommodate. Unlike

GOOG-411, which is very domain-dependent, Google search by Voice must be capable of handling anything that Google search can handle.

Table 1: Example queries to Google search by Voice.



Google search by Voice. For each key area of acoustic modeling and language modeling we will describe some of the challenges we faced as well as some of the solutions we have developed to address those unique challenges.

Using our voice to access information has been part of science fiction ever since the days of Captain Kirk talking to the Star Trek computer. Today, with powerful smartphones and cloud based computing, science fiction is becoming reality. In this chapter we give an overview of Google Search by Voice and our efforts to make speech input on mobile devices truly ubiquitous.

This chapter is a case study of the development of Google Search by 1 Voice - a step toward our long term vision of ubiquitous access. While the integration of speech input into Google search is a significant step toward more ubiquitous access, it posed many problems in terms of the performance of core speech technologies and the design of effective user interfaces.

Work is ongoing - the problems are far from solved. However, we have, at least, achieved a level of performance such that usage is growing rapidly, and many users become repeat users.

Searching for information by voice has been part of our every day lives since long before the internet became prevalent. It was already the case thirty years ago that, if you needed information for a local business, the common approach was to dial directory assistance (411 in the US) and ask an operator for the telephone number. 800-GOOG-411 [2] is an automated system that uses speech recognition and web search to help people find and call businesses.

## ALGORITHM:

Step1: Set path for ChromeDriver

System.*setProperty*("webdriver.chrome.driver", "C:\\Users\\Charan\\Downloads\\chromedriver\_win32\\chromedriver.exe");

Step2: Create an Object for ChromeDriver. WebDriver driver=**new** ChromeDriver();

Step3: Maximize the window

Step4: Open Google Search Engine. driver.get("https://[www.google.co.in/](http://www.google.co.in/)");

Step5: Create an object to perform click operation on mic WebElement mic=driver.findElement(By.*className*("XDyW0e"));

mic.click();

Step6: After clicking on mic a pop up notification will be displayed to enable permission to access mic

Step7: Use Robot Class to perform keyEvents to allow permission to turn on mic

Step8: Now you can perform voice based inputs in Google Search Engine.

## REQUIREMENTS:

* + Eclipse
  + Selenium Server
  + Chrome WebDriver\

## IMPLEMENTATION:

Steps we followed to perform automation testing for voice based input in Google Search Engine:

* + We used Eclipse to perform this operation
  + We initially Downloaded selenium server and chromedriver

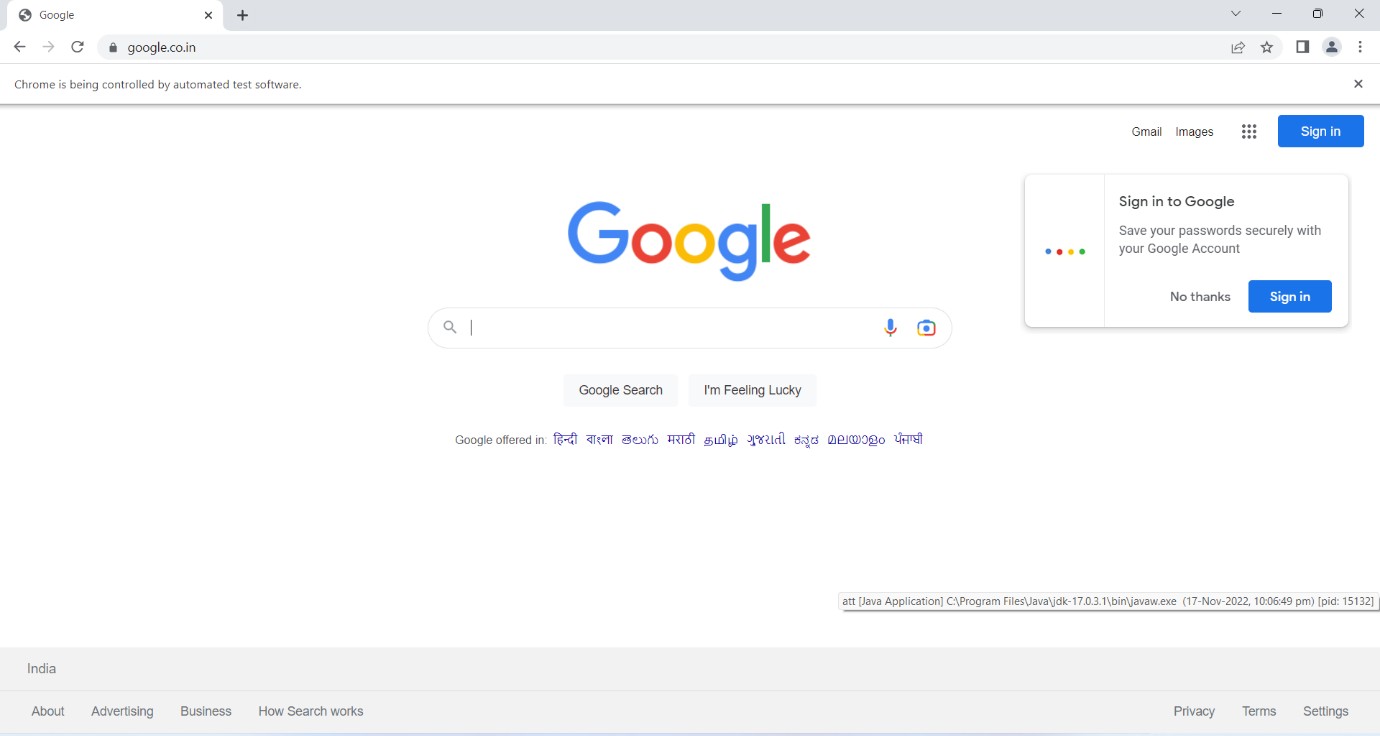
Step1: Set path for ChromeDriver

System.*setProperty*("webdriver.chrome.driver", "C:\\Users\\Charan\\Downloads\\chromedriver\_win32\\chromedriver.exe");

Step2: Create an Object for ChromeDriver. WebDriver driver=**new** ChromeDriver();

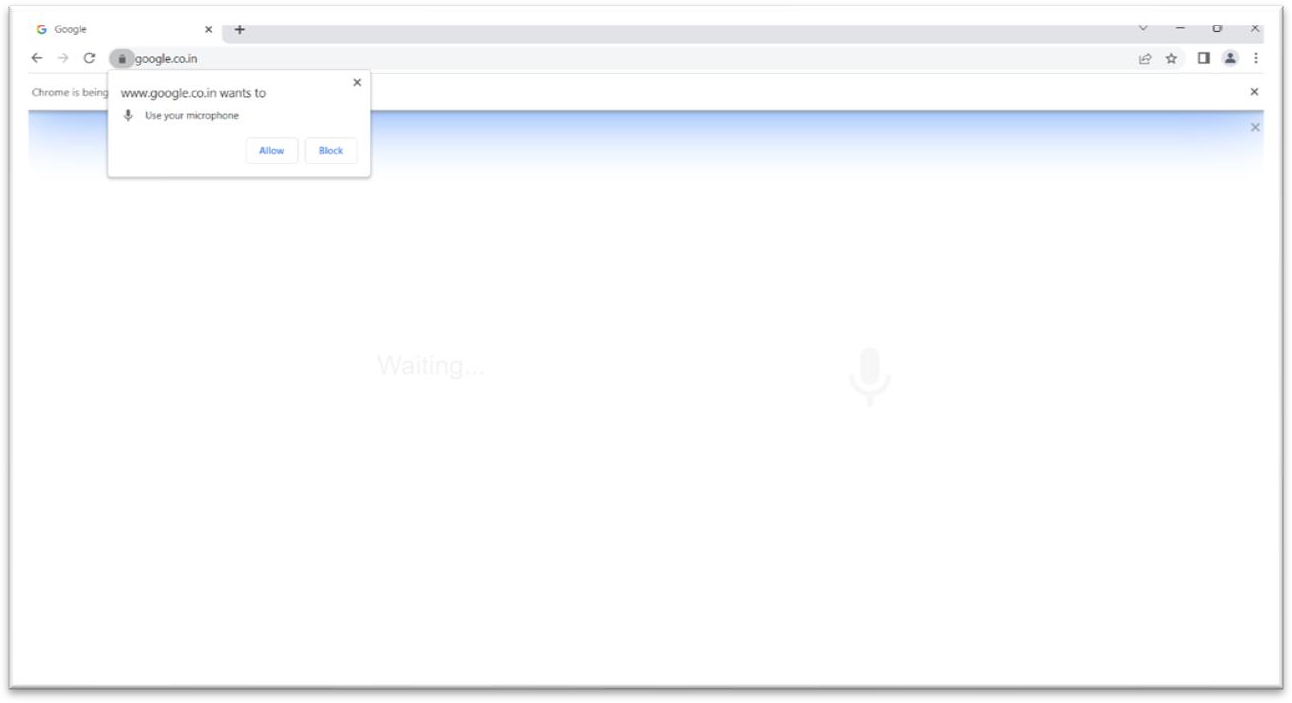
Step3: Maximize the window

Step4: Open Google Search Engine. driver.get("https://[www.google.co.in/](http://www.google.co.in/)");

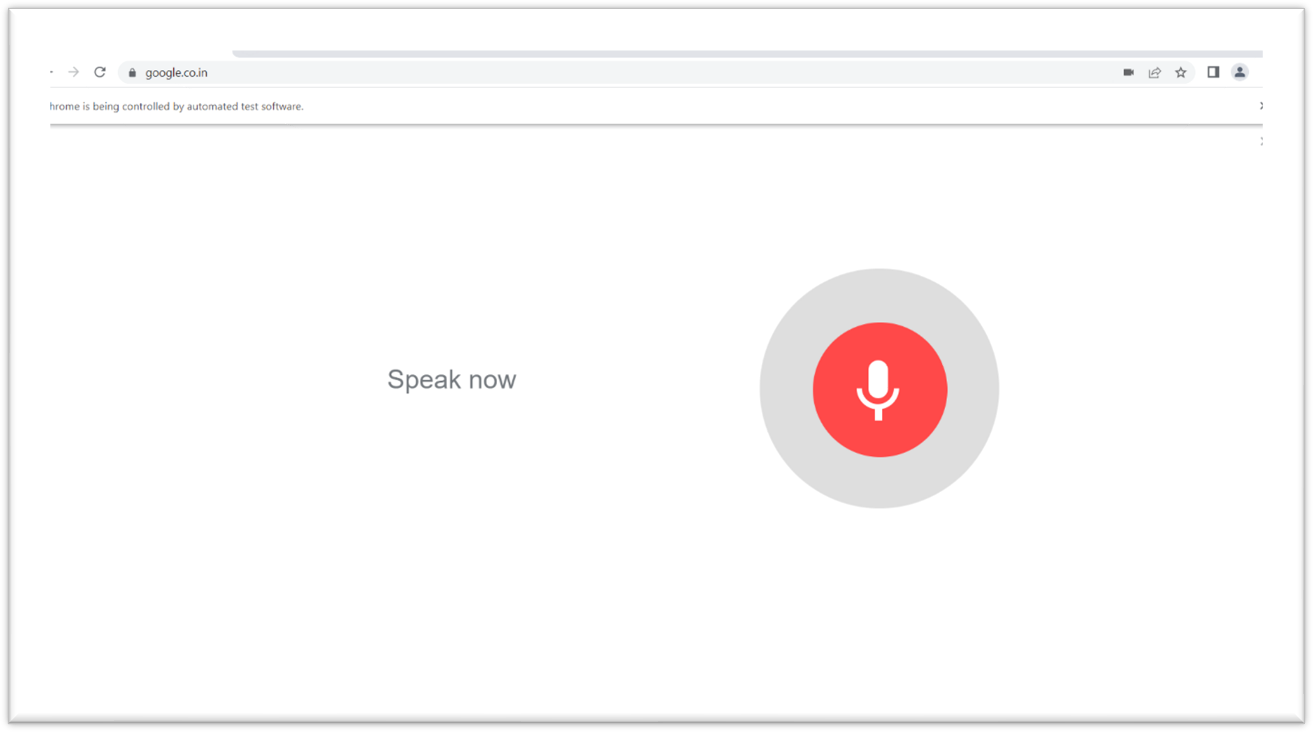


Step5: Create an object to perform click operation on mic driver.findElement(By.*className*("XDyW0e")).click();

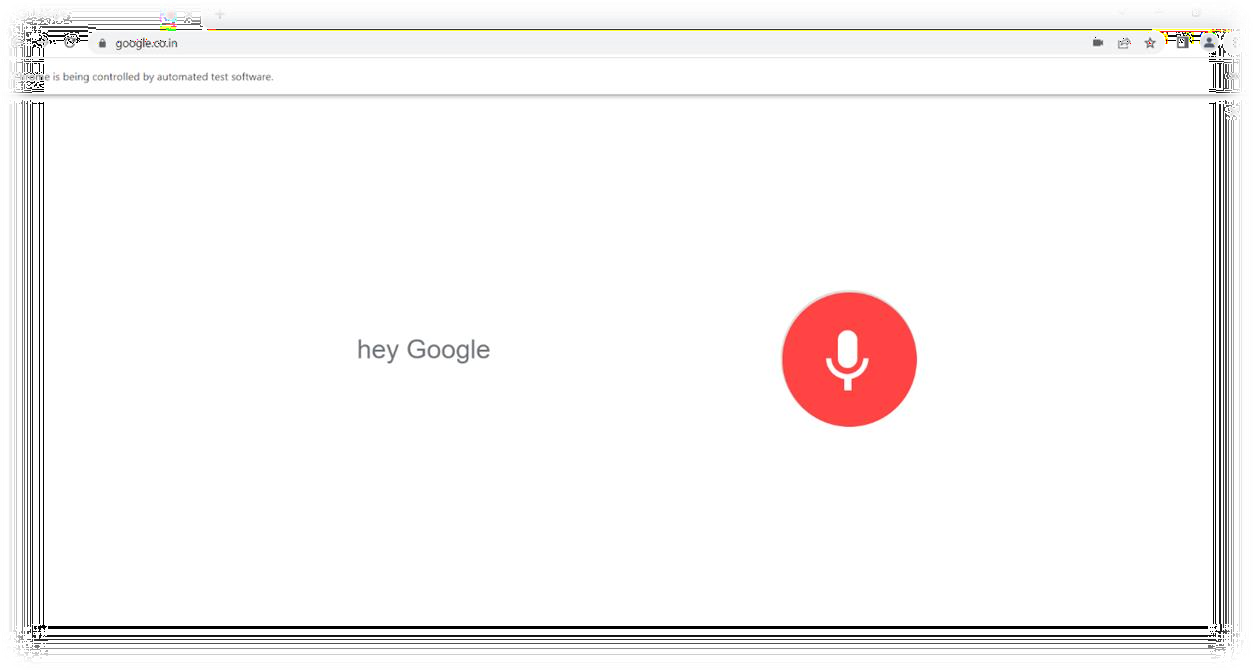
Step6: After clicking on mic a pop up notification will be displayed to enable permission to access mic



Step7: Use Robot Class to perform keyEvents to allow permission to turn on mic



Step8: Now you can perform voice based inputs in Google Search Engine.



#### The full source code we performed is:

**import** java.awt.Robot;

**import** java.awt.event.KeyEvent;

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.WebElement;

**import** org.openqa.selenium.chrome.ChromeDriver;

**public class** att {

**public static void** main(String[] args) **throws** Exception {

// **TODO** Auto-generated method stub System.*setProperty*("webdriver.chrome.driver",

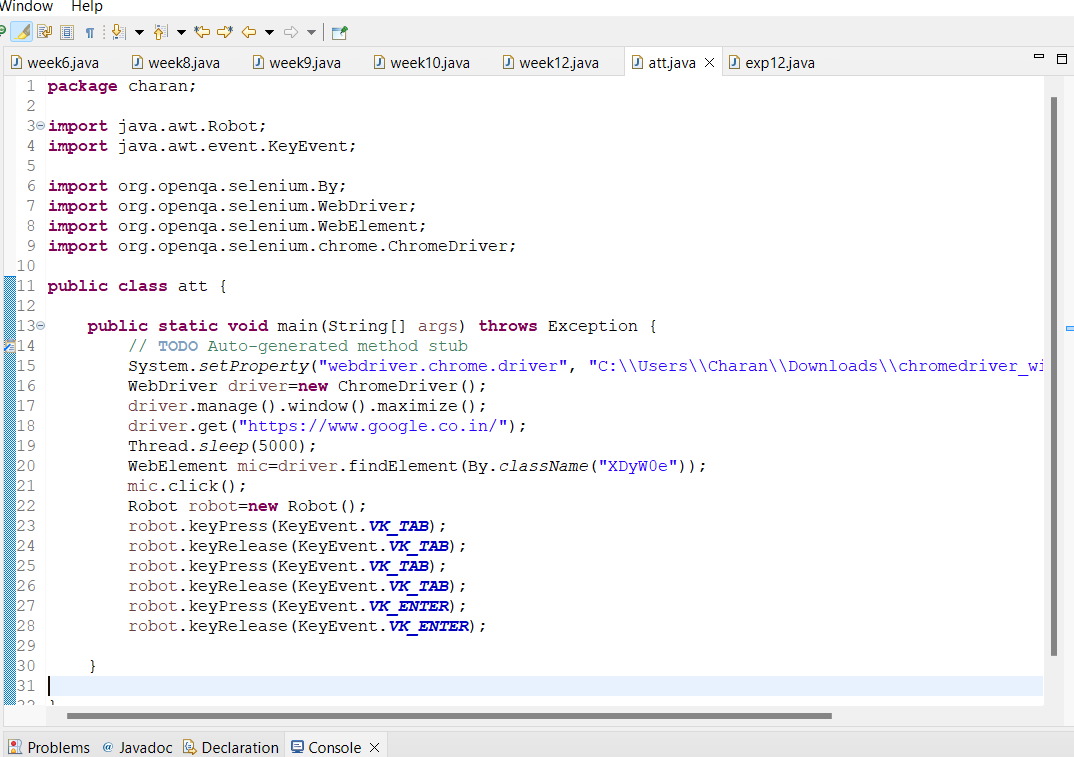
"C:\\Users\\Charan\\Downloads\\chromedriver\_win32\\chromedriver.exe"); WebDriver driver=**new** ChromeDriver(); driver.manage().window().maximize(); driver.get("https://[www.google.co.in/](http://www.google.co.in/)"); Thread.*sleep*(5000);

WebElement mic=driver.findElement(By.*className*("XDyW0e")); mic.click();

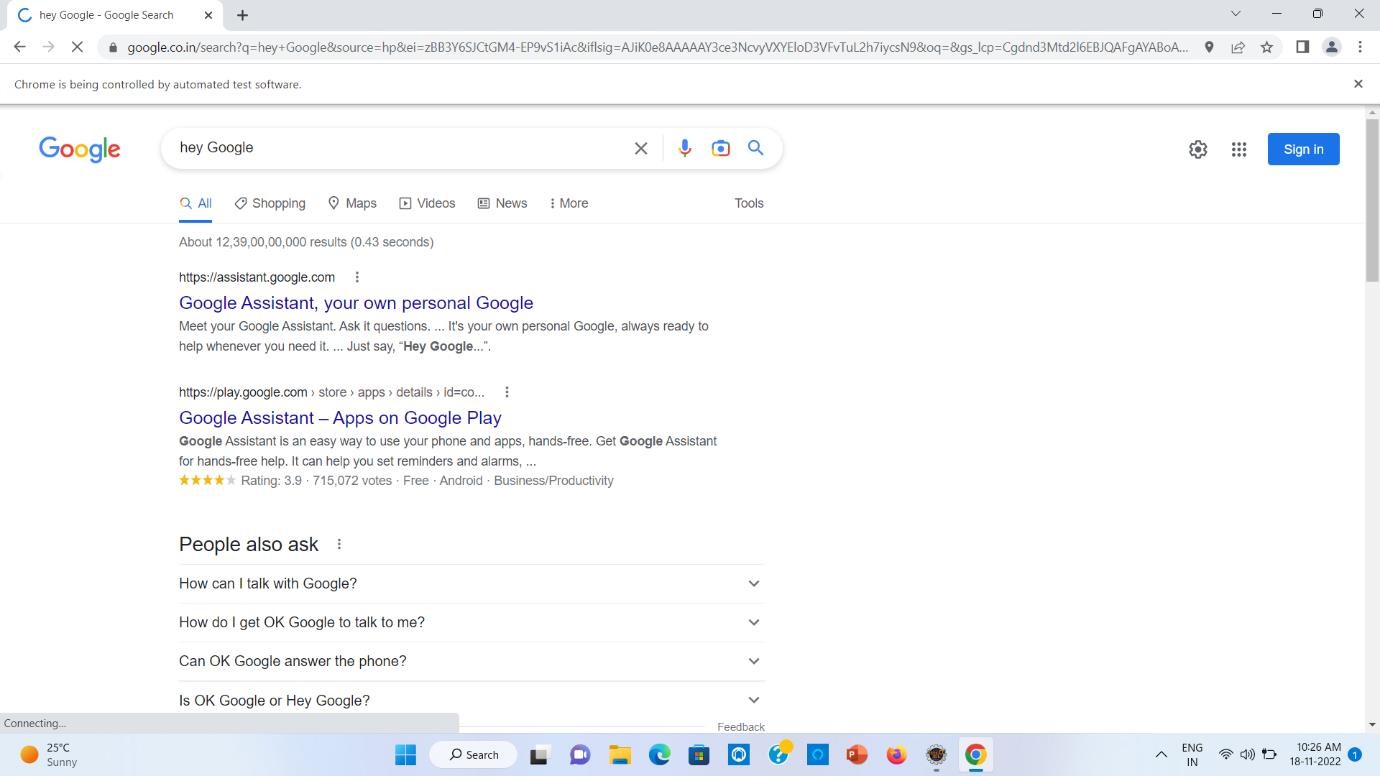
Robot robot=**new** Robot(); robot.keyPress(KeyEvent.***VK\_TAB***); robot.keyRelease(KeyEvent.***VK\_TAB***); robot.keyPress(KeyEvent.***VK\_TAB***); robot.keyRelease(KeyEvent.***VK\_TAB***); robot.keyPress(KeyEvent.***VK\_ENTER***); robot.keyRelease(KeyEvent.***VK\_ENTER***);

}

}



## RESULT:



1. **CONCLUSION:**

* Using specialized software for automated testing has many advantages.
* Reducing routine work
* Increasing testing accuracy
* Saving time and money

#### REFERENCES:

* <https://www.geeksforgeeks.org/>
* https://[www.java.org/](http://www.java.org/)
* <https://www.selenium.dev/downloads/>
* https://chromedriver.chromium.org/downloads