IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, May 2022
Impact Factor: 6.252

Screen Save: An Integrated Chrome Extension for Screenshot, Long Screenshot and Screen Recording

Mr. Sudarsanam[1] and Deepan Surya Raj S V[2]
Assistant Professor, Department of Cyber Security[1]
UG Scholar, Department of Computer Science and Engineering[2]
SRM Valliammai Engineering College, Chengalpattu, Tamil Nadu, India

Abstract: *Google Chrome extensions are apps that can be installed in Chrome to alter its functionality. This can include adding new capabilities to Chrome or changing the program's existing behaviour to make it more user-friendly. A screenshot, also known as a screen cap or screengrab, is a digital snapshot that depicts the contents of a computer screen. Screenshots allow you to capture exactly what's on your screen for later sharing or reference. Taking, saving, and sharing screenshots can come in handy. Some argue that the screenshot is the most important aspect of the internet. In this project, a Chrome browser was developed, which has features like taking screenshots of the webpage, taking long screenshots, capturing multiple screenshots, and screen recording. And these captured screenshots can be downloaded in different formats, i.e., jpg, png, pdf. In the multiple screenshot feature, users can choose an option called "combine into a single pdf" so that multiple screenshots can be downloaded in a single pdf. Within the screen recorder, the video can be downloaded in two different formats, i.e., mkv and mp4.*

Keywords: Screenshots, Screen Recorder, DOM, Automatic Evaluation, Chrome API, Long Screenshot, Current Active View.

I. INTRODUCTION

A chrome extension is a small programme that modifies the Chrome browser's experience or adds capabilities. They're made with web technologies including HTML, CSS, and JavaScript. The fundamental goal of an extension is to fulfil a single purpose around which the entire programme is built; it can have several components, but they must all contribute to the program's core goal. The interface of an extension should be as simple as possible, or it can extend to a web page, but the main goal should be to provide good functionality with low overhead. Screenshots let you capture exactly what you're seeing on your screen to share with others or reference later. Taking, saving, and sharing screenshots can be extremely helpful. This project involves creating a Chrome extension that captures screenshots, lengthy screenshots, and screenshots specifically for webpages. This project is also valuable for documentation and education.

A. System Overview

There are four sections in the proposed system architecture (screenshot, multiple screenshot, long screenshot, screen recording). The screenshot module uses the Chrome API to acquire the current view of the current active window in the browser and a simple DOM method to save the image in the appropriate format. The system uses the same chrome api to get the current view of the current active window in the browser for numerous photos at once while taking multiple screenshots. It also uses the standard DOM method to download the numerous photos as a group or individually. The system uses an external snapshot API to capture the entire page in the lengthy screenshot module. gets the picture and saves it to the local storage using the FileSaver.js Module. The system leverages the browser's api to record any desired screen or tab of the browser and can be downloaded using the regular DOM approach in the screen recording module.

II. RELATED WORKS

Neelam Singh Gurjar & et al., (2021) [1]: A browser extension, also known as a plugin or an addon, is a small software application that adds functionality to a web browser. However, security threats are always linked with such software where data can be compromised and ultimately trust is broken. The proposed research work jas developed asecurity model named WebSecAsst, which is a chrome plugin relying on the Machine Learning modelXGBoost and VirusTotal to detect malicious websites visited by the user and to detect whether thefiles downloaded from the internet are Malicious or Safe. During this

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, May 2022
Impact Factor: 6.252

detection, the proposed modelpreserves the privacy of the user's data to a greater extent than the existing commercial chrome extensions.

Asma K. Al-Khamis & et al., (2015) [2]: Electronic transactions rank the top on our daily transactions. Internet became invaluable for government, business, and personal use. This occurred in synchronization with the great increase in online attacks, particularly the development of newest forms from known attacks such as Tabnabbing. Thus, users' confidentiality and personal information must be protected using information security. Tabnabbing is a new form of phishing. The attacker needs nothing to steal credentials except users' preoccupation with other work and exploitation of human memory weakness. The impact of this malicious attempt begins with identity theft and ends with financial loss. That has encouraged some security specialists and researchers to tackle tabnabbing attack, but their studies are still in their infancy and not sufficient. The work done here focuses on developing an effective anti-tabnabbing extension for the Google Chrome browser to protect Internet users from been victims as well as raisetheir awareness. The system developed has a novel significance due to its effectiveness in detecting a tabnabbing attack and the combination of two famous approaches used to combat online attacks. The success of the system was examined by performance measurements such as confusion matrix and ROC. The system produces promising results.

Muhammad Nomani Kabir & et al., (2019) [3]: Web extension is a software that can be installed on a web browser. A web-extension link is displayed as an icon on the toolbar of the browser. Based on browsing activity, the extension works automatically or by clicking the extension icon depending on the functionalities made in the extension software. In this work, we developed a web extension on Google Chrome browser to verify online texts by simply clicking on an extension button. Upon clicking the button, the underlying algorithm in the extension software retrieves the texts from the current web-page being displayed. Verification and authentication of texts are performed by comparing the retrieved texts with text database. According to the comparison, texts are highlighted in colors. We consider authentication of Arabic Hadith texts for a case study. The authentic Hadith texts are highlighted by green color; authentic texts with partial diacritics, by yellow; and unauthentic texts, by red. This technique can also be used to authenticate laws, constitutions and Government documents in any language.

Dhawaleswar Rao CH & Sujan Kumar Saha (2020) [4]: Automatic multiple-choice question (MCQ) generation from a text is a popular research area. Researchers have been attracted toward automatic MCQ generation since the late 90's. Since then, many systems have been developed for MCQ generation. We perform a systematic review of those systems. This paper presents our findings on the review. We outline a generic workflow for an automatic MCQ generation system. The workflow consists of six phases. For each of these phases, we find and discuss the list of techniques adopted in the literature. We also study the evaluation techniques for assessing the quality of the system generated MCQs. Finally, we identify the areas where the current research focus should be directed toward enriching the literature.

Inmaculada Pardines& Marcos Sanchez-Elez et.al., (2014) [5]: This paper presents a proposal for assessing the laboratory sessions of a subject of the first year in computer science degrees. This methodology is based on online short-answer exam questions related to the concepts studied in each session. After analyzing the academic results of a large group of students, we may conclude that this way of evaluating knowledge is precise. The obtained grades neither underestimate nor overestimate the student's work, being similar to the ones achieved in a final exam. Moreover, there is continuous feedback, which allows the teacher to go into detail about those aspects of the subject that students have not understood.

III. EXISTING SYSTEM

A native screenshot tool is a built-in screen recorder in a browser or operating system that allows you to record your webcam, screen, or both at once. For example, both Windows 10 and Windows 11 come with built-in screen capture tools (Snip & Sketch and Snipping Tool), as well as a variety of keyboard shortcuts that make taking a screenshot a breeze. Chrome also includes a built-in screen recorder that may be accessed through keyboard shortcuts. With these integrated screen recording tools, you may record a piece of your screen, your full screen, a tab, or your entire browser.

IV. PROPOSED SYSTEM

In this proposed project, a separate integrated chrome extension for screen shot long screenshot and screen recording is developed, which overcomes the shortcomings of the previous existing system in that there is no need to crop for eliminating the headers in the nav bar, and you can choose your desired format. In the suggested system, the extension can be applied

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, May 2022
Impact Factor: 6.252

to any platform, including Whatsapp, Facebook, and email, with the crucial function of file sharing. The project has been improved with the time-consuming issue shown in the existing project screenshot. This project focuses solely on the webpage, allowing users to be free of headers and the bottom navbar.

V. METHODOLOGY

The suggested system is made up of four parts that work together to create a powerful system. screenshot, multiple screenshot, long screenshot, screen recording. Figure 1 depicts the actions of the previously discussed modules.are shown in the figure 1

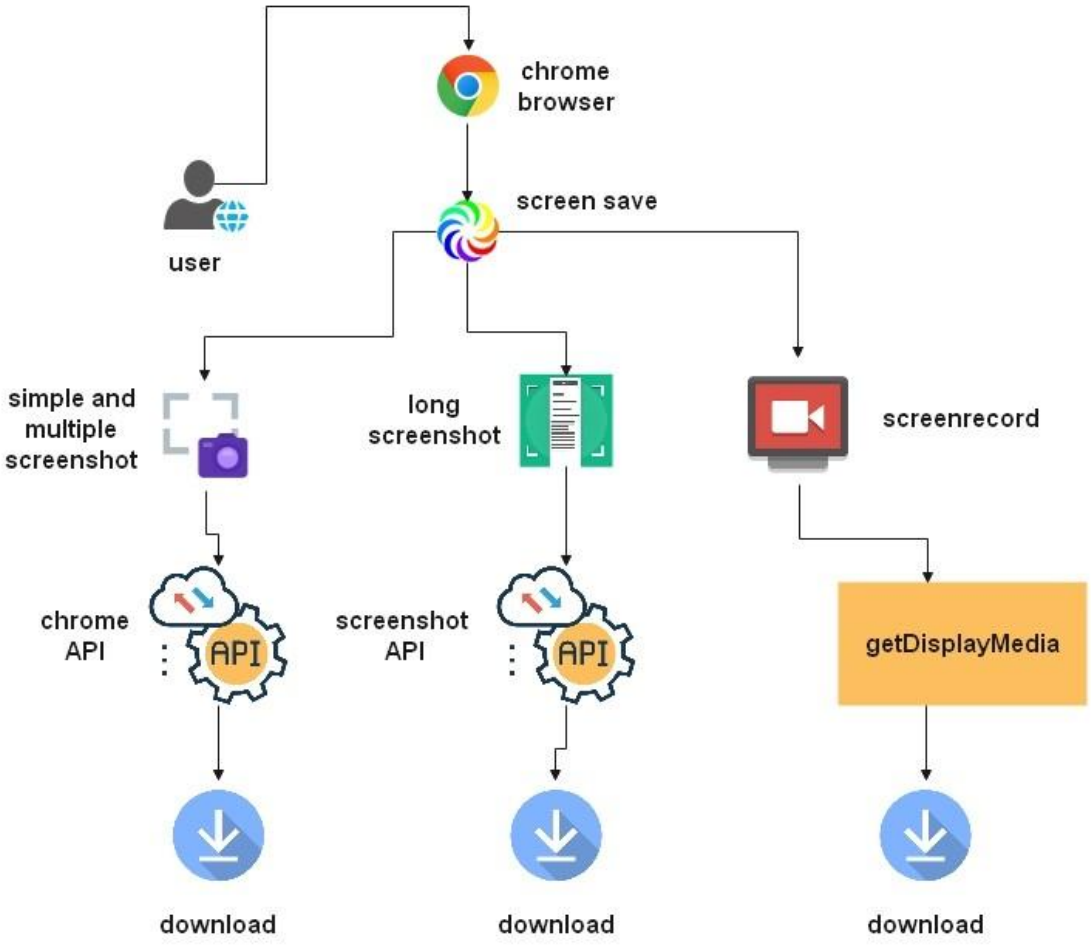

Fig. 1. System Architecture of Proposed System

A. Simple Screenshort Module

This module allows you to capture a screenshot of the current active view of the active webpage. The chrome extension uses the chrome API, which is inherited by default, when the screenshot button is selected after selecting the screenshot format. The chrome API retrieves information about the current active view, or the user's visible region on the webpage. The API sends back a response in base 64 format. A temporary element, an anchor tag, is produced in the HTML document using the DOM, and then a downloadable URL is created using a technique (URL.createObjectURL). The anchor tag then places this link in the href attritube of the anchor tag and downloads it using ".download". As a result, the basic screenshot module is implemented. The behaviour of authentication module is shown in fig. 2

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, May 2022
Impact Factor: 6.252

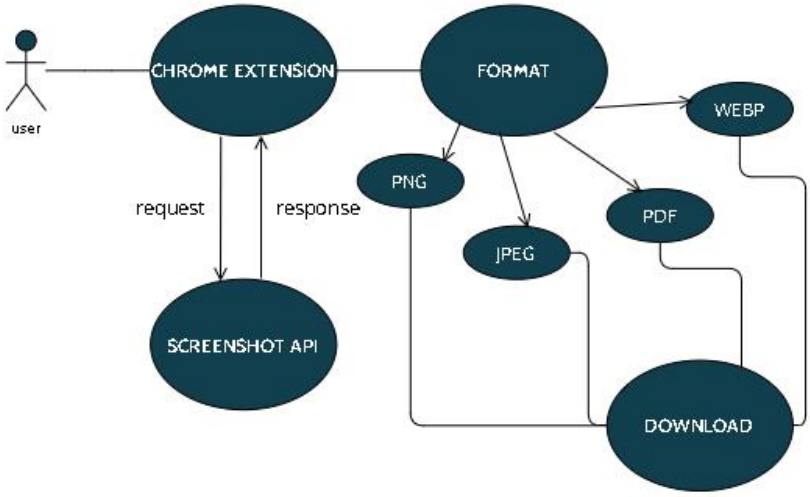

Fig. 2. Simple Screenshot Module

B. Multiple Screenshot Module

The current active view of the current active webpage is comparable to the preceding module. In the simple screenshot module, the action will be completed after the image has been downloaded. However, in this module, the extension will maintain its state even after the screenshot has been taken, and it will do so for any number of screenshot shots. The state is stopped and the download is executed only when the user clicks the "combine into single pdf" or "download individually" options. Follow the same steps as the preview module for the download. The addon makes use of a library for the downloads to merge into a single pdf option.

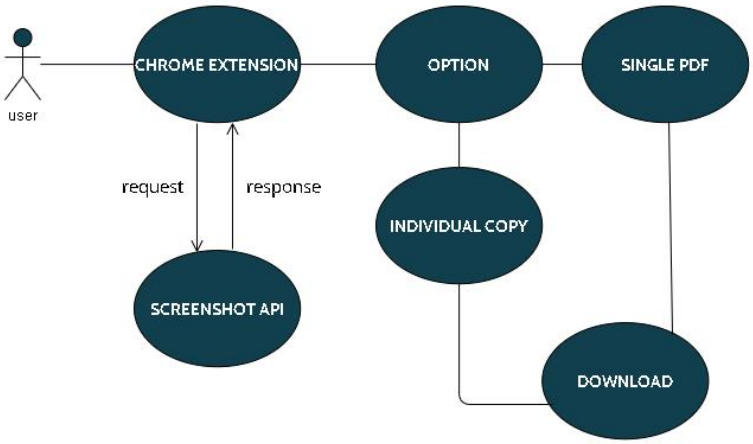

Fig. 3. Multiple Screenshot Module

C. Long Screenshot Module

This module takes a screenshot of the entire document, from the top to the bottom of the page. Through the chrome API, the plugin obtains the current active webpage's link. The 'Screenshot API' is a unique API for taking screenshots. The Screenshot API receives the link to the currently active webpage. In the URL, the format is specified. The screenshot is for the specified website. And the image's response is received in blob format. "URL.createObjectURL" is then used to convert the blob format to a URL. FileSaver.js is a javascript opensource library that is used to download a file with a configurable file name

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, May 2022
Impact Factor: 6.252

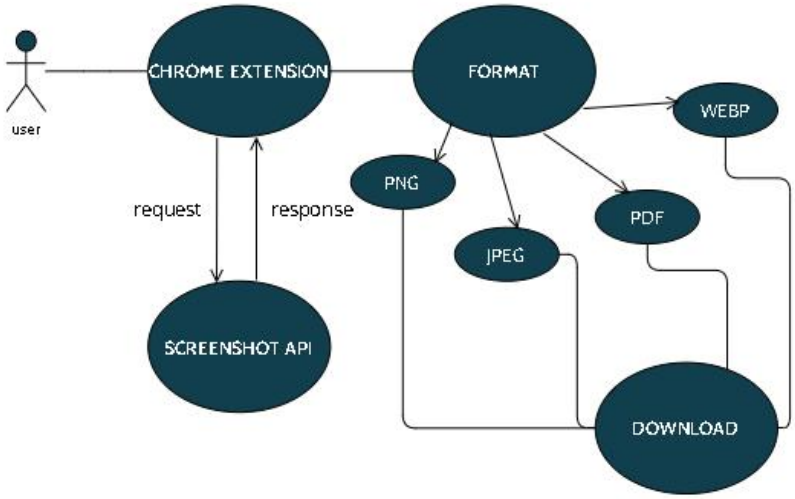

Fig. 4. Long Screenshot Module

D. Screen Recorder Module

The screen of the browser or a specific tab in the browser can be captured and recorded as a video with this module. First, allow the user to choose the input source. We can ask users to select the screen to be recorded using the getDisplayMedia function. We can also select a certain browser tab. getDisplayMedia returns a promise and is an asynchronous method. We'll obtain a recorded stream of data after we resolve the commitment. The extension should be to save the MediaStream, create a MediaRecorder. A stream will be returned by the getDisplayMedia function. We'll establish a MediaRecoreder and record the video using that stream. The MediaRecorder has: A method for starting to capture media, the data that is broadcast and received in the data available event, and to stop recording, use the stop method. Then, When the screen recording is finished, save it. The MediaRecorder's stop method can be used to halt the recording. We get the stop event on the MediaRecorder and deliver the recordedChunks once the recording has finished. We make a blob out of the recordedChunks and download it.

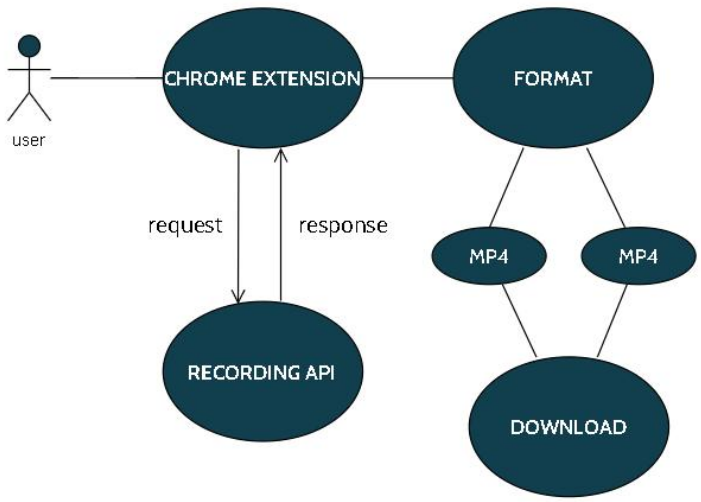

Fig.5. Screen Recorder Module

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, May 2022
Impact Factor: 6.252

VI. IMPLEMENTATION

The code was written using the Visual Studio code editor and the Chrome web browser. It offers developers millions of extensions to make the development process easier and more pleasurable.

This system uses 5 modules.

1. Simple Screenshot
2. Multiple Screenshot
3. Long Screenshot
4. Screen recorder

These components work together to keep the system running smoothly. Each module contributes significantly to the data flow and user experience. In Chapter III, the behaviour of all the modules listed is discussed.

A. HTML:

HTML stands for Hyper Text Markup Language, which is a programming language used to create web pages and apps. Let's look at what the terms Hypertext Markup Language and Web page signify. "Text within Text" is what HyperText signifies. A hypertext is a text that has a link. You have clicked on a hypertext when you click on a link that takes you to a new webpage. HyperText is a technique for connecting two or more web pages (HTML documents). A markup language is a computer language for applying formatting and layout principles to text documents. The markup language makes text more dynamic and interactive. It can convert text into graphics, tables, and links, among other things.

The structure of the HTML is mention below, in the table.1

<!DOCTYPE>	This tag is used to tells the HTML version. This currently tells that the version is HTML 5.
<html>	This is called HTML root element and used to wrap all the code.
<head>	It is the header Head tag contains metadata, title, page CSS etc.
<body>	The body tag is used to contain all of the data on a web page, including words and links. This element contains all of the material that you see rendered in the browser.

Table. 1. Structure of HTML

B. CSS

CSS (Cascading Style Sheets) is a simple design language designed to make the process of making web pages presentable easier. The style and feel of a web page is handled by CSS. You can use CSS to change the text colour, font style, paragraph spacing, how columns are scaled and laid out, what background pictures or colours are used, layout designs, display variants for different devices and screen sizes, and a range of other effects. HTML and CSS design is a must-have talent if you wish to pursue a career as a professional web designer. CSS is simple to learn and understand, but it gives you a lot of power over how an HTML document looks. CSS is frequently used in conjunction with the markup languages HTML or XHTML.

C. JavaScript

JavaScript is an interpreted, lightweight programming language. It is intended for the development of network-centric applications. It works in tandem with and complements Java. Because JavaScript is interwoven with HTML, it is incredibly simple to use. It's free to use and cross-platform. Javascript is the most widely used programming language in the world, making it an excellent choice for programmers. When you learn JavaScript, you can use it to create powerful front-end and back-end software utilising Javascript-based frameworks like jQuery and Node.JS. JavaScript is omnipresent; it's built into every modern web browser, so there's no need to set up a special environment to learn it. JavaScript is supported by Chrome, Mozilla Firefox, Safari, and every other browser available today. JavaScript enables you to create stunningly gorgeous and lightning-fast websites. You can create a website with a console-like appearance and feel to provide the finest Graphical User Experience to your visitors.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, May 2022
Impact Factor: 6.252

D. Screen Capture API

The Screen Capture API extends the existing Media Capture and Streams API by allowing users to choose a screen or a piece of a screen (such as a window) to capture as a media stream. This broadcast can then be captured or shared over the internet with others. The Screen Capture API is rather straightforward to utilise. MediaDevices is its only technique. getDisplayMedia(job)'s is to prompt the user to choose a screen or a piece of a screen to capture as a MediaStream. The Screen Capture API does not have its own interfaces; instead, it extends the existing MediaDevices interface with one method.

E. Screenshot API

It is an external api that is used to take screenshots from a URL that has been requested. The image response is then sent back to the client or server, where it can be shown or downloaded in the specified format, such as PNG, jpeg, pdf, or webp

VII. CONCLUSION

This project focuses on screenshots, extended screenshots, multiple screenshots, and screen recordings. In comparison to the present project, these implementations are more efficient and speedier. This project can be useful for web developers or web bloggers who want to take screenshots or record the screen so that they can document it in the future because the webpage will be updated frequently with newer versions. When the button for screenshot or screen recording is clicked, it provides a quicker option to download the image/video in a single click without the header of the browser and navbar of the windows or any operating system.

A. Future Scope

The only goal of this project is to download the image to the localhost. This project might be enhanced by providing the necessary API for direct sharing of the downloadable screenshot file via Whatsapp, Telegram, and other similar platforms. Any implementation for altering the screenshot is not included. Cropping the image, marking and sketching on the webpage, and so on are all options for editing the screenshot. A marquee tool can be provided for the simple screenshot and video record modules so that the user can crop the necessary section for the screenshot or screenrecord. Taking a long screenshot takes a little time. And this is something that can be improved in future efforts.

ACKNOWLEDGEMENT

We sincerely express our gratitude in depth to our esteemed Founder Chairman & Chancellor Dr. T. R. Paarivendhar, Thiru. Ravi Pachamoothoo Chairman SRM Group & Pro-Chancellor (Admin), Mrs.R.Padmapriya, Correspondent, SRM VEC & Director SRM Group and authorities of SRM VALLIAMMAI ENGINEERING COLLEGE for the patronage on our welfare rooted in the academic year. We express our sincere gratitude to our respected Principal Dr.B.Chidhambararajan, M.E., Ph.D., for his constant encouragement, which has been our motivation to strive towards excellence. We thank our sincere Vice Principal Dr.M.Murugan,M.E.,Ph.D., for his constant encouragement, which has been our motivation to strive towards excellence. We extend our hand of thanks to our Head of the Department, Dr.B.Vanathi, M.E., Ph.D., Professor for her unstinted support. We are grateful to our internal guide Mr. SUDARSANAM, M.Tech, without whose invaluable guidance and encouragement, this project would not have been a success.
We also like to thank all Teaching and non-teaching staff members of our department, for their support during the course of the project. We finally thank our friends and family for their support during the course of the project

REFERENCES

[1]. Neelam Singh Gurjar , Sudheendra S S R, Chejarla Santosh Kumar, and Krishnaveni K. S, "WebSecAsst - A Machine Learning based Chrome Extension", IEEE publication, DOI: 10.1109/ICCES51350.2021.9488953
[2]. Asma K. Al-Khamis, Ayman A. Khalafallah, "Secure Internet on Google Chrome: Client side anti-tabnabbing extension", IEEE publication, DOI: 10.1109/Anti-Cybercrime.2015.7351942
[3]. Muhammad Nomani Kabir, Omar Tayan, Yasser Alginahi ,Md Munirul Hasan, Md Arafatur Rahman, "On the development of a web extension for text authentication on Google Chrome", IEEE publication, DOI: 10.1109/ECACE.2019.8679250

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, May 2022
Impact Factor: 6.252

[4]. Evgeny Pyshkin, Maxim Mozgovoy, Alexander Chisler, Yulia Volkova, "Striving with online addiction with a self-control chrome extension", IEEE publication, DOI: 10.1109/SSCI.2016.7850190 \

[5]. Lujo Bauer, Shaoying Cai, Limin Jia, Timothy Passaro, Yuan Tian, "Analyzing the dangers posed by Chrome extensions", IEEE publication, DOI: 10.1109/CNS.2014.6997485