IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, March 2026
Impact Factor: 8.2

Multi-Tool Chrome Extension

Prof. Alisha Borkar[1], Bhavana Dhande[2], Tejas Patil[3]
[1]Assistant Professor, Department of Computer Application
[2],[3] PG Scholar Department of Computer Application
K.D.K. College of Engineering, Nagpur, Maharashtra, India
Alisha.borkar@kdkce.edu.in , dhandebanil.mca@kdkce.edu.in tejasfpatil.mca@kdkce.edu.in

Abstract: *Modern web users frequently rely on multiple online tools for tasks such as password generation, hashing, QR code creation, SSL verification, proxy configuration, link analysis, and clipboard management. However, switching between multiple websites for these utilities reduces efficiency and increases security risks. This paper presents a Multi-Tool Chrome Extension that integrates essential web utilities into a single lightweight browser extension. The system provides modules including Proxy Configuration, Hash Generator, Secure Password Generator, QR Code Generator, Clipboard Manager, SSL Certificate Checker, and URL Link Analyzer. The extension follows a modular architecture ensuring scalability, secure processing, and responsive performance within the Chrome browser environment. Experimental testing demonstrates improved user productivity, reduced dependency on third-party websites, and enhanced security awareness. The proposed extension offers a unified, fast, and secure solution for everyday web utility tasks.*

Keywords: Chrome Extension, Web Security, Password Generator, Hashing, SSL Verification, QR Code Generator, Proxy Tool, URL Analyzer

I. INTRODUCTION

Web browsers have evolved into powerful platforms that support extensions designed to improve productivity, usability, and security. With the rapid growth of internet usage, users frequently require access to various utility tools such as password generators, hash converters, SSL certificate checkers, QR code generators, and link analyzers. These tools help users perform important tasks related to data security, authentication, and information sharing. However, most of these utilities are available on separate websites or standalone applications, which forces users to leave their current browsing session and search for the required tool. The need to visit multiple third-party websites creates several challenges. It increases time consumption, interrupts workflow, and may expose sensitive information to untrusted platforms. For example, when users generate passwords or convert sensitive text into hashes on external websites, there is always a risk of data misuse or privacy breaches. Additionally, using different tools across multiple platforms results in a fragmented user experience where users must repeatedly switch tabs and adapt to different interfaces for similar tasks. To address these limitations, the proposed Multi-Tool Chrome Extension integrates several essential utilities into a single browser extension. The system includes tools such as Proxy configuration, Hash generator, Secure password generator, QR code generator, Clipboard manager, SSL certificate checker, and URL link analyzer. By operating directly within the Chrome browser, the extension allows users to access these tools instantly without relying on external websites. This unified approach improves efficiency, enhances security by reducing data exposure, and provides a convenient all-in-one solution for everyday web utility tasks.

II. LITERATURE REVIEW AND MOTIVATION

Several browser extensions and online tools have been developed to provide specific web utilities such as password generators, QR code generators, proxy switchers, and SSL verification tools. These tools are designed to assist users in improving security, managing web data, and performing quick conversions or checks while browsing. For example, password generator extensions help users create strong and secure passwords, QR code generators convert URLs or text

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 4, March 2026 Impact Factor: 8.2

into scannable codes, proxy switchers allow users to change network routing for privacy purposes, and SSL verification tools help users check whether a website connection is secure. While these tools are useful individually, they typically provide only a single function and do not support
multiple utilities within the same platform. Because most extensions focus on one specific task, users often need to install several different extensions to access all required functionalities. This practice leads to several disadvantages. Installing multiple extensions increases browser memory consumption and can slow down browser performance. In addition, each extension requires separate permissions, which may introduce potential security vulnerabilities if the extension is not properly maintained or trusted. Another challenge is the inconsistency in user interfaces and workflows across different extensions, which makes it difficult for users to manage them efficiently. Furthermore, these tools usually operate independently and do not share data or integrate with each other, limiting their overall usability and convenience.

Research in browser-based security and utility tools emphasizes the importance of modular architecture and local data processing to enhance privacy and system performance. Many studies highlight that performing sensitive operations locally within the browser can reduce the risk of data exposure and improve response time. However, despite these advancements, limited research has focused on developing a unified extension that integrates multiple security and productivity tools into a single platform. This lack of integrated solutions creates an opportunity to design a comprehensive browser extension that combines several commonly used utilities. Therefore, the proposed Multi-Tool Chrome Extension is developed to address this gap by providing a centralized, efficient, and secure environment where users can access multiple web utilities through a single extension interface.

III. PROPOSED SYSTEM ARCHITECTURE

A. System Overview

The Multi-Tool Chrome Extension is designed using a modular architecture. Each functionality operates as an independent module while sharing a common user interface and background service.

The system includes:

1. Proxy Tool
2. Hash Generator
3. Password Generator
4. QR Code Generator
5. Clipboard Manager
6. SSL Certificate Checker
7. Link Analyzer

B. Architectural Layers

1. Presentation Layer

- Built using HTML, CSS, and JavaScript
- Provides popup interface
- User-friendly navigation between modules
- Real-time input validation

2. Application Logic Layer

This layer processes user inputs and executes tool-specific logic:

- Proxy Module: Configures browser proxy settings.
- Hash Module: Generates MD5, SHA-1, SHA-256 hashes.
- Password Module: Creates secure random passwords with customizable length and symbols.
- QR Module: Converts text or URL into QR code image.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 4, March 2026 Impact Factor: 8.2

• Clipboard Module: Saves and retrieves copied text.
• SSL Module: Verifies HTTPS protocol and checks certificate validity.
• Link Analyzer Module: Validates URL format and detects unsafe patterns.

3. Background Service Layer
• Handles persistent storage using Chrome Storage API
• Manages permissions
• Ensures secure processing
• Maintains session data

C. System Workflow
Step 1 – User opens extension Step 2 – Selects desired tool
Step 3 – Provides input (URL, text, password length, etc.) Step 4 – Processing occurs locally or via secure API
Step 5 – Output displayed instantly Step 6 – Optional data saved in storage
This workflow ensures fast response time and smooth interaction.

IV. METHODOLOGY

The development of the Multi-Tool Chrome Extension followed a structured approach to ensure efficient design, secure implementation, and reliable performance. The process began with requirement analysis, where the essential tools required for daily web utility tasks were identified. These tools included proxy configuration, hash generation, password generation, QR code creation, clipboard management, SSL verification, and link analysis. Both functional and non-functional requirements such as usability, performance, and security were considered during this stage. After identifying the requirements, the design phase was conducted where user interface wireframes were created to plan the layout of the extension popup and tool modules. A modular architecture was designed so that each tool operates independently while sharing a common interface, making the extension
scalable and easier to maintain or expand in the future. During the implementation phase, the extension was developed using standard web technologies. HTML was used to create the structural layout of the extension, CSS was used for styling and designing a user-friendly interface, and JavaScript was used to implement the functional logic of each tool. Additionally, Chrome Extension APIs were used to interact with browser features such as storage, permissions, and background processing. Security measures such as input validation, local data processing, and secure storage using Chrome APIs were implemented to protect user data and ensure safe operation. Finally, the system underwent a testing phase, where functional testing was performed for each module to verify correct output, along with performance evaluation and compatibility testing to ensure the extension runs smoothly and efficiently within the Chrome browser environment.

V. RESULTS

The developed extension was tested under various real-world scenarios.
Observations:
• Password generation was instant and customizable.
• Hash generation produced accurate outputs for multiple algorithms.
• QR code generation worked for both text and URLs.
• SSL checker successfully identified secure (HTTPS) and insecure (HTTP) websites.
• Proxy switching functioned correctly with manual configuration.
• Clipboard module stored multiple copied entries efficiently.
• Link analyzer detected invalid and malformed URLs.

DOI: 10.48175/IJARSCT-31838

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, March 2026
Impact Factor: 8.2

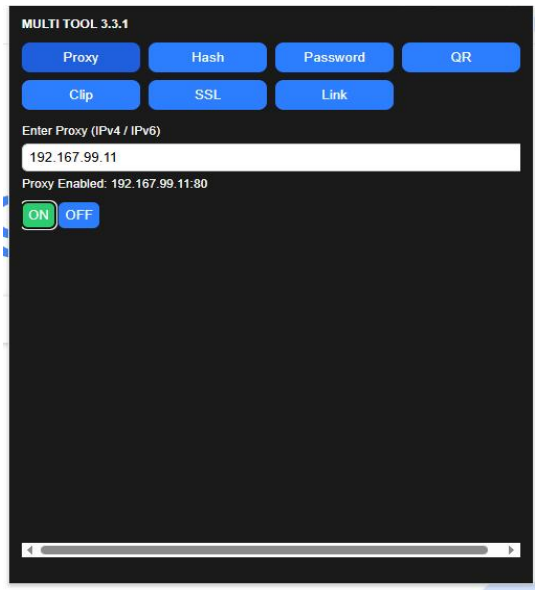

Fig. 1.1

Performance Outcome:

- Lightweight extension size
- Minimal memory usage
- Fast execution time
- Improved user productivity

User feedback indicated that integrating all tools in a single extension significantly improved convenience.

VI. COMPARATIVE ANALYSIS

Feature	Separate Online Tools	Multiple Extensions	Proposed Multi- Tool Extension
Unified Interface	No	No	Yes
Local Processing	Partial	Partial	Yes
Reduced Installation	No	No	Yes
Security Control	Limited	Moderate	High
Performance	Slower	Moderate	Fast
Modular Design	No	No	Yes

The proposed system offers better integration, efficiency, and security compared to traditional approaches.

VII. CONCLUSION

This paper presented the design and implementation of a Multi-Tool Chrome Extension integrating essential web utilities and security features. The system successfully combines Proxy configuration, Hash generation, Password creation, QR code generation, Clipboard management, SSL verification, and URL analysis within a single browser extension.

The modular architecture ensures scalability, secure processing, and improved user experience. Experimental evaluation demonstrates enhanced productivity and reduced reliance on third-party utility websites

.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, March 2026
Impact Factor: 8.2

Future Enhancements

Integration of Encryption/Decryption tool, Dark mode UI enhancement, Malware URL detection using API, Cloud sync for clipboard history, Two-factor authentication support

REFERENCES

[1] Google Chrome Developers, "Chrome Extensions Documentation," — official guide for building browser extensions including APIs, manifest files, and permissions.
[2] Mozilla Developer Network (MDN), "WebExtensions API," — cross-browser extension development guide covering storage, tabs, and scripting APIs.
[3] OWASP Foundation, "OWASP Top 10 Web Application Security Risks," — essential security practices for protecting browser extensions and web tools.
[4] William Stallings, Cryptography and Network Security: Principles and Practice, — foundational concepts for hashing, encryption, and secure communication in tools like password generators and hash utilities.
[5] Internet Engineering Task Force (IETF), "RFC 6234 – Secure Hash Algorithms (SHA)," — standard specifications for SHA hashing used in hash generator tools.
[6] World Wide Web Consortium (W3C), "Web Cryptography API," — provides cryptographic operations used in browser-based tools like encryption and secure password generation.
[7] Google, "Clipboard API and Permissions API Documentation," — used for clipboard management and permission handling in multi-tool extensions.
[8] React Documentation, "React – Building User Interfaces," — used for creating interactive extension popup UI.
[9] Node.js and NPM Documentation, — for managing dependencies and backend logic in extension development workflows.