SOFTWARE REQUIREMENTS SPECIFICATION

(SRS)

PROJECT: QuickShort Browser Extension

Version: 1.0
Status: Final
Architecture: Manifest V3

Prepared For:

Department of Computer Science and Engineering

D.N. Polytechnic, Meerut

Prepared By:

Pratham

1. INTRODUCTION

1.1 Purpose

The purpose of this document is to present a detailed description of the **"QuickShort"** Browser Extension. It outlines the functional and non-functional requirements, system interface, and design constraints. This document serves as the primary reference for the development and validation of the software.

1.2 Scope

"QuickShort" is a client-side browser extension compatible with Chromium-based browsers (Google Chrome, Microsoft Edge). Its primary scope is to streamline the process of sharing URLs by:

1. Interfacing with the **TinyURL API** to shorten links.
2. Providing a **Context Menu** for instant background shortening.
3. Providing a **Popup Interface** for custom alias creation.
4. Automating the clipboard copy process.

1.3 Definitions, Acronyms, and Abbreviations

Term	Definition
SRS	Software Requirements Specification
API	Application Programming Interface (TinyURL)
Manifest V3	The latest extension platform architecture by Google
Service Worker	An event-driven script that runs in the background
DOM	Document Object Model
Context Menu	The menu that appears when a user right-clicks

2. OVERALL DESCRIPTION

2.1 Product Perspective

QuickShort is an add-on software component. It is not a standalone desktop application; it exists within the runtime environment of the web browser. It interacts with the Browser's internal APIs (Tabs, Scripting, Notifications) and communicates externally with the TinyURL server via HTTPS.

2.2 User Characteristics

The target audience includes:

- **General Internet Users:** For sharing clean links on social media.
- **Students & Professionals:** For citing sources in reports or presentations without clutter.
- **Technical Expertise:** Minimal (Requires basic ability to install a browser extension).

2.3 Assumptions and Dependencies

1. **Internet Access:** The user must be online to reach the API endpoint.
2. **Browser Version:** The user must have Chrome (v88+) or Edge installed.
3. **API Availability:** The system relies on `tinyurl.com` being operational.

---

3. FUNCTIONAL REQUIREMENTS

This section describes the specific behaviors and functions the system supports.

3.1 Context Menu Module (Right-Click)

- **FR-01:** The system shall inject a menu item "⚡ Quick Shorten & Copy" into the browser's native context menu.
- **FR-02:** Upon selection, the system shall capture the URL of the active tab.
- **FR-03:** The system shall send an asynchronous request to the TinyURL API.
- **FR-04:** The system shall utilize the `scripting` API to inject a temporary copy command into the active tab.
- **FR-05:** The system shall display a desktop notification confirming the action.

3.2 Popup Interface Module (Foreground)

- **FR-06:** The system shall display a graphical interface when the extension icon is clicked.
- **FR-07:** The system shall automatically populate the input field with the current tab's URL.
- **FR-08:** The system shall allow the user to edit the source URL (e.g., Paste functionality).
- **FR-09:** The system shall allow the user to input a **Custom Alias** string.
- **FR-10:** The system shall validate the API response:
  - If successful: Copy to clipboard and show a success message.
  - If failed (Alias taken): Show a visual error message.

3.3 Error Handling

- **FR-11:** If the input URL is empty, the system shall prevent the API call.
- **FR-12:** If the user is on a restricted page (e.g., `chrome://settings`), the system shall handle the permission error gracefully and notify the user.

4. NON-FUNCTIONAL REQUIREMENTS

4.1 Performance

- **NFR-01 Response Time:** The application shall generate and copy a link in under 2 seconds (assuming standard 4G/Broadband connection).
- **NFR-02 Resource Usage:** The extension shall utilize **Service Workers**, ensuring 0% CPU and RAM usage when the extension is idle.

4.2 Security & Privacy

- **NFR-03 Data Privacy:** The system shall not store, track, or transmit user browsing history to any third-party analytics.
- **NFR-04 Permissions:** The system shall operate on the **Least Privilege Principle**, using `activeTab` permissions to only access data when explicitly triggered by the user.
- **NFR-05 Content Security Policy (CSP):** The system shall not execute remotely hosted code, complying with Manifest V3 security standards.

4.3 Reliability

- **NFR-06:** The system shall provide a fallback notification if the clipboard copy action fails due to browser security focus loss.

4.4 Usability

- **NFR-07:** The user interface shall be clean, utilizing standard system fonts (Segoe UI/Roboto) to blend with the browser environment.
- **NFR-08:** Critical actions (Shorten & Copy) shall be achievable with a single click in the popup.

5. SYSTEM INTERFACES

5.1 Software Interfaces

- **Browser Engine:** Chromium (Google Chrome, Microsoft Edge, Brave).
- **Communication Protocols:** HTTPS (TLS 1.2/1.3) for secure API communication.

5.2 External Interface (API)

The system communicates with the TinyURL API via the following endpoint:

- **Endpoint:** `https://tinyurl.com/api-create.php`
- **Method:** HTTP GET
- **Query Parameters:**
    - `url` : The long URL to be shortened.
    - `alias` : (Optional) The desired custom name.

6 Use Case Summary

1. **Actor:** User.
2. **Action:** Clicks Extension Icon.
3. **System:** Displays Popup with URL pre-filled.
4. **Actor:** Enters Custom Alias -> Clicks "Shorten".
5. **System:** Calls API -> Returns Short Link -> Copies to Clipboard.

7. CONCLUSION

The QuickShort Browser Extension fulfills the requirement for a fast, privacy-focused, and efficient URL management tool. By adopting the Manifest V3 architecture, it ensures long-term compatibility with modern browsers while providing essential utility through its dual-mode interface.