1. Introduction

1.1 Purpose

The Movie SpoilerShield is an innovative software solution designed to enhance the user experience by preventing unwanted exposure to movie spoilers while browsing the internet. This Software Requirements Specification (SRS) document outlines the detailed requirements for the development of the extension. The purpose of this document is to provide a comprehensive understanding of the features, functionalities, and constraints associated with the Movie Spoiler Blocker Chrome Extension.

The scope of this project encompasses the development of a browser extension utilizing advanced technologies such as BERT (Bidirectional Encoder Representations from Transformers), Python, JavaScript, and web scraping techniques. The extension aims to empower users with a tool that intelligently identifies and blocks potential movie spoilers across various websites, allowing them to enjoy online content without the risk of encountering plot revelations.

This SRS focuses on defining the software requirements for the entire Movie Spoiler Blocker Chrome Extension, covering its core functionalities and interactions with the user and external systems. It serves as a foundation for the development team, guiding them in the implementation of a robust and effective solution to enhance the movie-watching experience for users.

1.2 Intended Audience and Reading Suggestions

This Software Requirements Specification (SRS) is crafted to cater to a diverse audience involved in the development, testing, deployment, and management of the Movie Spoiler Blocker Chrome Extension. The primary stakeholders and intended audience for this document include:

1. **Development Team:** Software developers responsible for designing, coding, and testing the Movie Spoiler Blocker Chrome Extension. System architects and designers involved in defining the overall structure and components of the extension.
2. **Quality Assurance (QA) Team:** Testers engaged in verifying and validating the functionality, performance, and security aspects of the extension.
3. **Project Managers:** Individuals overseeing the planning, scheduling, and execution of the development project.
4. **Product Managers:** Those responsible for the overall vision, strategy, and market positioning of the Movie Spoiler Blocker Chrome Extension.
5. **System Administrators:** Individuals responsible for deploying, configuring, and maintaining the extension in various environments.
6. **End Users:** Individuals who will install and utilize the Movie SpoilerShield to enhance their browsing experience.

1.3 Product Scope

The Movie SpoilerShield is a sophisticated solution crafted to mitigate the inadvertent exposure to movie spoilers during online browsing. Its primary objective is to augment user satisfaction

by seamlessly identifying and obstructing potential spoilers across diverse websites. The key facets of the product scope are elucidated as follows:

- **Spoiler Identification:** Leveraging state-of-the-art natural language processing, particularly BERT, the extension adeptly discerns and evaluates textual content to identify potential movie spoilers.

- **Real-time Blocking:** The extension ensures a real-time defense mechanism, intercepting spoilers before they are presented to the user while navigating through various web pages.

- **Compatibility:** Seamlessly integrating with the Google Chrome browser, the extension guarantees a harmonious and user-friendly experience, catering to a broad audience.

- **Efficient Web Scraping:** Employing advanced web scraping techniques, the extension extracts pertinent information from web pages, facilitating accurate identification and timely blocking of spoilers.

- **Scalability and Maintainability:** The software architecture is engineered for scalability, enabling seamless updates and enhancements to accommodate evolving movie-related content and internet technologies.

- **Business Strategy:** This strategically positions itself in the market by addressing a specific pain point for movie enthusiasts—spoiler exposure—thus differentiating itself and contributing to enhanced user engagement and loyalty.

2. Overall Description

2.1 Product Perspective

The Movie SpoilerShield stands as an independent and innovative solution within the domain of web browser extensions. Designed to enhance user experience by preventing inadvertent exposure to movie spoilers during online activities, this extension integrates seamlessly with popular browsers like Google Chrome. Leveraging advanced technologies such as BERT, Python, and Javascript, along with web scraping techniques, it operates as a self-contained component. Unlike being part of an existing product family, it serves as a unique tool, ensuring a spoiler-free browsing experience. In the broader context, the extension acts as a protective layer for users exploring movie-related content, contributing to a seamless and enjoyable online experience.

2.2 Product Functions

The Movie SpoilerShield is designed to deliver a seamless and efficient user experience by performing key functions aimed at preventing accidental exposure to movie spoilers. Its major functions include real-time analysis of webpage content using BERT for spoiler detection, dynamic blocking of spoiler content, customizable user settings for tuning the spoiler sensitivity, automatic updates for maintaining effectiveness. By summarily organizing these functions, the extension ensures users can navigate movie-related content without fear of encountering spoilers, enhancing their overall browsing experience.

2.3 User Classes and Characteristics

The Movie SpoilerShield is anticipated to cater to a diverse user base with varying characteristics. The primary user classes identified for this product are:

1. **Casual Browsers:**

   - **Frequency of Use:** Regular but not necessarily daily.
   - **Characteristics:** Casual users who want to avoid movie spoilers during occasional browsing sessions. They may have a general understanding of web browsers but may not be technically inclined.
   - **Requirements:** Intuitive user interface, easy installation, and minimal configuration. The extension should operate seamlessly in the background without requiring active user management.

2. **Movie Enthusiasts:**

   - **Frequency of Use:** Frequent users engaged in movie-related content.
   - **Characteristics:** Individuals passionate about movies who actively seek information online. They may possess a moderate level of technical expertise and a keen interest in customizing their browsing experience.
   - **Requirements:** Customizable spoiler sensitivity settings, the ability to whitelist trusted websites, and real-time updates for the latest movie data.

3. **Support Services:**

   - **Frequency of Use:** As needed.
   - **Characteristics:** Users who may require assistance with installation, troubleshooting, or general inquiries. This class includes individuals seeking support services for technical issues, ensuring that the extension operates smoothly.
   - **Requirements:** Accessible and responsive customer support channels, detailed documentation, and a user-friendly troubleshooting guide. The extension should provide clear error messages and diagnostics to facilitate efficient support.

4. **Developers:**

   - **Frequency of Use:** Regular, with a focus on customization and extension enhancements.
   - **Characteristics:** Individuals with a strong technical background, including proficiency in Python and Javascript. Developers may contribute to the extension's improvement, create custom features, or integrate it into their own projects.
   - **Requirements:** Well-documented API and extension architecture, developer-friendly tools for extension enhancement, and an active developer community.

2.4 Operating Environment

The Movie SpoilerShield is designed to operate within the following environment:

- **Hardware Platform:** The extension is platform-agnostic and can run on any hardware that supports Google Chrome or compatible Chromium-based browsers. This includes desktops, laptops, and compatible mobile devices.

- **Operating System and Versions:** The extension is compatible with major operating systems, including but not limited to Windows (7 and above), macOS, and Linux distributions. It is specifically designed for seamless integration with Google Chrome on these platforms. The extension should be compatible with the latest versions of Google Chrome and updates should be provided to adapt to changes in the browser's architecture.

- **Browser Compatibility:** The primary operating environment is Google Chrome, and compatibility extends to other Chromium-based browsers. This includes popular browsers like Microsoft Edge, Opera, and Brave, ensuring a consistent experience across a variety of platforms.

- **Additional Software Components or Applications:** The Movie SpoilerShield operates as a standalone software solution and does not require additional software components for its core functionality. However, it may leverage external APIs or services for data updates and web scraping. Users are expected to have a stable internet connection for real-time updates.

- **Memory and Storage Requirements:** The extension is designed to be lightweight, with minimal impact on system resources. It should operate effectively with standard memory and storage configurations typical of modern computing devices.

2.5 Design and Implementation Constraints

The development of the Movie Spoiler Blocker Chrome Extension is subject to several constraints that shape the design and implementation process:

- **Quality of Data:** The effectiveness of the spoiler detection mechanism relies heavily on the quality and relevance of the underlying data. The extension's performance is directly impacted by the accuracy and comprehensiveness of the movie-related data used for analysis. Ensuring access to reliable and up-to-date data sources is critical to maintaining the extension's functionality.

- **Computational Capabilities:** The handling of large datasets, especially when utilizing advanced natural language processing techniques like BERT, imposes computational constraints. Developers need to possess the necessary computational resources to process and analyse data efficiently. Optimizations in algorithms and code are essential to manage computational complexity.

- **Web Scraping Limitations:** The extension utilizes web scraping to gather information from online sources. However, it must operate within ethical and legal boundaries, respecting the terms of service of websites and avoiding any adverse impact on server loads. Developers should implement scraping strategies that prioritize efficiency and compliance.

- **Security Considerations:** As the extension operates within the browser environment, security is of paramount importance. Developers must adhere to best practices in coding to prevent vulnerabilities and potential exploitation. Any interactions with external APIs or services should be conducted securely, and user data privacy must be a priority.

- **Design Conventions and Standards:** Adherence to coding standards, design conventions, and best practices is crucial for maintainability and collaboration. If the customer's organization will be responsible for maintaining the software, the development team should align with their established conventions to ensure smooth handover and ongoing support.

2.6 User Documentation

- **User Manuals:** Comprehensive written guides explaining how to use the software, navigate its features, and make the most of its capabilities.

- **Online Help/FAQs:** An online help system or Frequently Asked Questions section integrated into the software to provide immediate assistance to users when they encounter issues or have questions.
- **Tutorials:** Video tutorials, like YouTube tutorials, that demonstrate how to use various aspects of the software effectively. These can be hosted on the platform's website or accessible through the software itself.
- **Quick Start Guides:** Concise guides for users who want to get started quickly and need basic instructions to set up their profiles and start using key features.
- **Feedback and Support Channels:** Information on how users can provide feedback, report issues, and contact the support team for assistance. This may include email addresses, chat support, or phone numbers.
- **Release Notes:** Detailed information about software updates, bug fixes, and new features in each release.

2.1 Assumptions and Dependencies

The successful development and deployment of the Movie Spoiler Blocker Chrome Extension are based on certain assumptions and dependencies. These factors may impact the project if they prove to be incorrect, change, or are not shared among stakeholders:

Assumptions:

1. **Availability of Quality Data:** It is assumed that reliable and quality movie-related data will be available for analysis. The effectiveness of the spoiler detection algorithm depends on the richness and accuracy of the underlying data.

2. **Web Scraping Legality:** The project assumes that web scraping activities conducted by the extension comply with legal and ethical standards. It is expected that the development team adheres to the terms of service of websites and avoids any actions that may violate regulations or policies.

3. **Stability of External APIs:** If the extension relies on external APIs for certain functionalities, it is assumed that these APIs remain stable and accessible. Changes or disruptions in these APIs could impact the extension's performance.

Dependencies:

1. **Chrome Browser Compatibility:** The Movie Spoiler Blocker Chrome Extension is dependent on the compatibility and stability of the Chrome browser. Changes in the browser's architecture or updates may necessitate adjustments to ensure continued functionality.

2. **Data Source Availability:** The project is dependent on the availability and consistency of movie-related data sources. Changes in the structure or accessibility of these sources may require adaptation in the extension's data retrieval and analysis processes.

3. External Interface Requirements

3.1 User Interfaces

The Movie Spoiler Blocker Chrome Extension is designed to provide an intuitive and user-friendly interface within the Google Chrome browser environment. While the detailed design specifications are covered in a separate user interface specification, the following outlines the logical characteristics of each interface between the software product and the users:

- **Browser Integration:** The extension seamlessly integrates with the Google Chrome browser, presenting users with a minimalist and non-intrusive interface. This integration ensures that the user experience aligns with standard Chrome extension conventions, allowing users to access the extension effortlessly.
- **Spoiler Sensitivity Settings:** Users could customize spoiler sensitivity settings through a dedicated options page. This page allows adjustments to the level of sensitivity for detecting potential spoilers, providing a personalized experience based on user preferences.
- **Whitelisting Functionality:** The extension includes functionality for users to whitelist trusted websites where spoilers are permitted. This feature enables users to exempt specific websites from spoiler blocking, catering to individual preferences.
- **Real-time Updates:** Users are informed of the extension's status through non-intrusive notifications, ensuring they are aware of the latest updates and the extension's operational state. Notifications are designed to be informative and discreet to maintain a smooth browsing experience.
- **User Guidance and Help:** The extension incorporates a help section accessible from the options page, providing users with guidance on how to use various features and customize settings. This ensures users can easily navigate and make the most of the extension's capabilities.
- **Error Messages and Diagnostics:** In the event of an issue, the extension communicates errors to users through clear and concise error messages. These messages follow established conventions to aid users in understanding and resolving issues effective.

3.2 Software Interfaces

The Movie Spoiler Blocker Chrome Extension interfaces with various software components to achieve its functionality. The connections between this product and other specific software components are described below:

- **Chromium-based Browsers**: The extension is designed to operate seamlessly within the Google Chrome browser environment. It leverages the Chrome Extension API to interact with the browser, enabling functionality such as content blocking, notifications, and user interface integration. Compatibility with other Chromium-based browsers is ensured to provide a consistent experience.
- **Python and Javascript Libraries:** The development of the extension involves the use of Python and Javascript for natural language processing, web scraping, and other functionalities. Specific libraries and versions, such as the BERT library for natural language processing in Python, are utilized to enhance the extension's capabilities.

- **Web Scraping Tools:** The extension may utilize web scraping techniques to gather movie-related data from external websites. While the specific tools and libraries are not explicitly mentioned here, they are integrated into the extension's backend to facilitate data retrieval and analysis.

4. System Features

4.1 Spoiler Detection and Blocking

4.1.1 Description and Priority

The Spoiler Detection and Blocking feature aims to analyze webpage content in real-time for potential spoilers and dynamically block spoiler material based on user preferences. This is a high-priority feature as it forms the core functionality of the Movie Spoiler Blocker Chrome Extension.

4.1.2 Stimulus/Response Sequences

- **Stimulus:** User navigates to a webpage containing movie-related content.
  - **Response:** Extension analyses the content using BERT for potential spoilers.
- **Stimulus:** Spoiler is detected based on user sensitivity settings.
  - **Response:** Spoiler content is dynamically blocked, preventing user exposure.
- **Stimulus:** User adjusts spoiler sensitivity settings in the options page.
  - **Response:** Spoiler detection is recalibrated according to the updated settings.

4.1.3 Functional Requirements

1. **Real-time Content Analysis:** The extension must continuously analyse webpage content using BERT to identify potential movie spoilers.

2. **Dynamic Spoiler Blocking:** Spoiler content must be dynamically blocked from view based on user-defined sensitivity settings.

3. **User Sensitivity Settings:** Users should be able to customize the level of spoiler sensitivity through the options page.

4. **Notification of Spoiler Detection:** Users receive non-intrusive notifications when spoilers are detected and blocked, providing information on the action taken.

5. **Error Handling:** In the event of errors or failures in spoiler detection, the extension should provide clear error messages or notifications to users.

6. **Compatibility:** Ensure compatibility with a variety of webpage structures and layouts to accurately detect spoilers.

5. Nonfunctional Requirements

5.1 Performance Requirements

The performance of the Movie Spoiler Blocker Chrome Extension is crucial to providing a seamless and responsive user experience. The following performance requirements outline the expected behavior of the extension under various circumstances:

1. **Real-time Content Analysis:**

   - **Requirement:** The extension must analyse webpage content in real-time using BERT.

   - **Rationale:** Real-time analysis ensures that spoilers are promptly detected, maintaining the element of surprise for users.

2. **Dynamic Spoiler Blocking:**

   - **Requirement:** Spoiler content must be blocked dynamically within 1 second of detection.

   - **Rationale:** Swift and responsive blocking is essential to prevent users from accidentally viewing spoilers while browsing.

3. **User Sensitivity Settings Adjustment:**

   - **Requirement:** Changes to user sensitivity settings should take effect instantly.

   - **Rationale:** Immediate adjustment allows users to fine-tune the extension's behaviour based on their preferences.

4. **Whitelisting Functionality:**

   - **Requirement:** The whitelisting process should take less than 2 seconds to update.

   - **Rationale:** Efficient whitelisting ensures users can quickly customize their preferences for specific websites.

5. **Notification Delivery:**

   - Requirement: Spoiler notifications must be delivered within 3 seconds of detection.

   - Rationale: Timely notifications inform users about the extension's actions, contributing to a transparent user experience.

6. **Efficient Algorithm:**

   - **Requirement:** The spoiler detection algorithm should have a low impact on CPU and memory usage.

   - **Rationale:** Efficiency is crucial to prevent performance degradation during regular browsing activities.

5.2 Safety Requirements

The Movie Spoiler Blocker Chrome Extension is designed to enhance the browsing experience by preventing exposure to spoilers. While the primary focus is on user convenience and content filtering, safety considerations are also paramount to ensure the responsible use of the extension. The safety requirements include:

1. **Legal and Ethical Web Scraping:**
   - Requirement: Web scraping activities must adhere to legal and ethical standards.
   - Safeguard: Ensure that web scraping is performed responsibly, respecting the terms of service of websites and applicable regulations.

2. **Notification Clarity:**
   - Requirement: Spoiler detection notifications must be clear and non-alarming.
   - Safeguard: Design notifications to inform users without causing undue stress or panic.

3. **Compatibility Testing:**
   - Requirement: Ensure compatibility with the latest browser updates.
   - Safeguard: Regularly test the extension with new browser versions to prevent compatibility issues.

4. **Error Handling Transparency:**
   - Requirement: Error messages should be informative and transparent.
   - Safeguard: Design error messages to guide users on resolving issues and understanding potential risks.

5. **Performance Impact Mitigation:**
   - Requirement: The extension's impact on system performance should be minimal.
   - Safeguard: Optimize algorithms and code to ensure efficient operation without causing system slowdowns.

5.3 Software Quality Attributes

- Usability
- Adaptability
- Maintainability
- Reliability
- Robustness
- Testability
- Performance Efficiency
- Compatibility
- Privacy and Security
- Accessibility
- Scalability
- Interoperability

6. Project Plan

6.1 Team Members
1. Shanu Mathew – EN20CS303045
2. Vaibhav Chaubey – EN20CS303053
3. Prince Soni – EN20CS303038

6.2 Division of Work
1. Shanu Mathew – Model Development
2. Vaibhav Chaubey – Web Scraping
3. Prince Soni – Backend

6.3 Time Schedule
1. **Project Initiation and Requirements (Week 1-2):**
   - Define project scope and objectives.
   - Set up the development environment.
   - Identify and document basic functional requirements.
2. **Implementation (Week 3-6):**
   - Begin coding based on basic requirements.
   - Implement core functionality (e.g., spoiler detection and blocking).
   - Develop a simple user interface.
3. **Testing and Refinement (Week 7-8):**
   - Conduct basic testing to ensure functionality.
   - Gather feedback from potential users.
   - Make necessary adjustments and refinements.
4. **Documentation and Presentation (Week 9-10):**
   - Document the codebase and basic system architecture.
   - Prepare a concise user guide.
   - Create a presentation summarizing the project.
5. **Final Review and Submission (Week 11):**
   - Conduct a final review of the project.
   - Make any last-minute refinements.

Appendix A: Glossary

1. **Movie SpoilerShield:** The innovative software solution designed to enhance the user experience by preventing exposure to movie spoilers while browsing the internet.

2. **BERT (Bidirectional Encoder Representations from Transformers):** A natural language processing technique used in the Movie Spoiler Blocker Chrome Extension for analyzing textual content to identify potential movie spoilers.

3. **Web Scraping:** The process of extracting information from websites, utilized by the extension to gather pertinent data for spoiler detection.

4. **Chromium-based Browsers:** Web browsers built on the open-source Chromium project, including Google Chrome, Microsoft Edge, Opera, and Brave.

5. **Spoiler Sensitivity Settings:** User-configurable settings that allow customization of the level of sensitivity for detecting potential movie spoilers.

6. **Whitelisting:** A functionality within the extension that enables users to switch off the extension for some websites.

7. **Real-time Content Analysis:** The continuous analysis of webpage content using BERT to identify potential movie spoilers as users navigate through various web pages.

8. **Dynamic Spoiler Blocking:** The real-time blocking of spoiler content based on user-defined sensitivity settings to prevent accidental exposure.

9. **Error Handling:** The process of managing and communicating errors or failures in spoiler detection, ensuring users receive clear and informative messages.

10. **Compatibility Testing:** Testing conducted to ensure the extension remains compatible with the latest updates of the Google Chrome browser.

11. **Web Scraping Legality:** The adherence to legal and ethical standards when conducting web scraping activities to gather information from online sources.

12. **Performance Impact Mitigation:** Measures taken to minimize the impact of the extension on system performance, ensuring efficient operation without causing slowdowns.