# Software Requirements Specification

**for**

# <RS Remover>

**Version 1.0 approved**

**Prepared by <Raj Shendkar>**

**<organization>**

**<25-2-23>**

# Table of Contents



# Revision History

| Name | Date | Reason For Changes | Version |
|---|---|---|---|
| | | | |
| | | | |

# 1. Introduction

## 1.1 Purpose

*The purpose of this document is to provide a detailed specification of the Automated Duplicate File Remover System. This system is designed to automate the process of file duplication deletion and remove the manual work required for finding and deleting duplicate files. This SRS will describe the overall behavior of the system, as well as its performance and quality assurance requirements.*

## 1.2 Document Conventions

*This system will be developed according to the IEEE standard. It is intended to speed up manual processes that consume a large amount of time. The font for the text will be Times New Roman, size 12, and for the header, it will be LGM, size 14 and bold.*

## 1.3 Intended Audience and Reading Suggestions

*This SRS will be helpful for the developer by providing guidelines and specifications for software development. It will also ensure that the final product meets end-user requirements and expectations, and serve as a communication tool with potential investors or customers. Additionally, it can be used as documentation for future reference and collaboration.*

## 1.4 Product Scope

*The purpose of the system is to provide an automated solution for deleting duplicate files.*

- *Automated detection and removal of duplicate files*
- *User-friendly interface for easy navigation and usage*
- *24/7 availability for users to manage their duplicate files*
- *History of removed duplicate files*
- *Search function with the ability to search for deleted and unique files, including filters for file status and size*
- *Backup option for removed duplicate files*

## 1.5 References

*The existing forms of duplicate file removers will be used as references, and any other manuals on duplicate file deletion that provide a vision and scope will also be referred to.*

# 2. Overall Description

## 2.1 Product Perspective

*Duplicate file deletion is a completely manual process that requires time, effort, and is often tiring and time-consuming, with various flaws associated with it. This new automated system version 1.0 has been developed from scratch to overcome these issues.*

## 2.2 Product Functions

- *Automate the file duplicate deletion process*
- *Provide a search option to find deleted and unique files*
- *Allow users to select and delete duplicate files in bulk*
- *Display a confirmation message before permanently deleting files*
- *Generate reports of deleted files and their locations*
- *Provide a user-friendly interface for easy navigation and operation*
- *Ensure data security and privacy during the deletion process*
- *Support various file formats for deletion*
- *Maintain a log of deleted files and actions taken by users.*

## 2.3 User Classes and Characteristics

*User of the system is general Users, People who want to remove duplicate files from their personal computer or device and also IT Professionals who manage computers and networks in organizations, and need to regularly remove duplicate files to optimize system performance and save storage space. Users should have basic knowledge of operating a computer or device,know how files are stored and managed on their computer or device. Users should be aware of the potential consequences of deleting files, and understand how to recover them if necessary.*

## 2.4 Operating Environment

*The RS Remover software will operate in a Windows or Linux operating system environment, specifically on Windows 10 or later versions. It will be compatible with both 32-bit and 64-bit architectures. The software will require a minimum of 2GB of RAM and 10GB of free disk space. It will also require access to the file system to perform its functions. The software will not have any dependencies on other software components or applications.*

## 2.5 Design and Implementation Constraints

*The design and implementation constraints for RS Remover include the use of the Python programming language and specific libraries for file management and deletion. The system will be designed to run on Windows and Mac operating systems and will require a minimum of 2GB of RAM for optimal performance. The system will be constrained by the user's file system permissions and may require administrative privileges to delete certain files. Security considerations will be taken into account to ensure that only duplicate files are deleted and no critical files are removed. Finally, the system will be designed to handle large numbers of files and will include efficient algorithms for identifying duplicate files.*

## 2.6 User Documentation

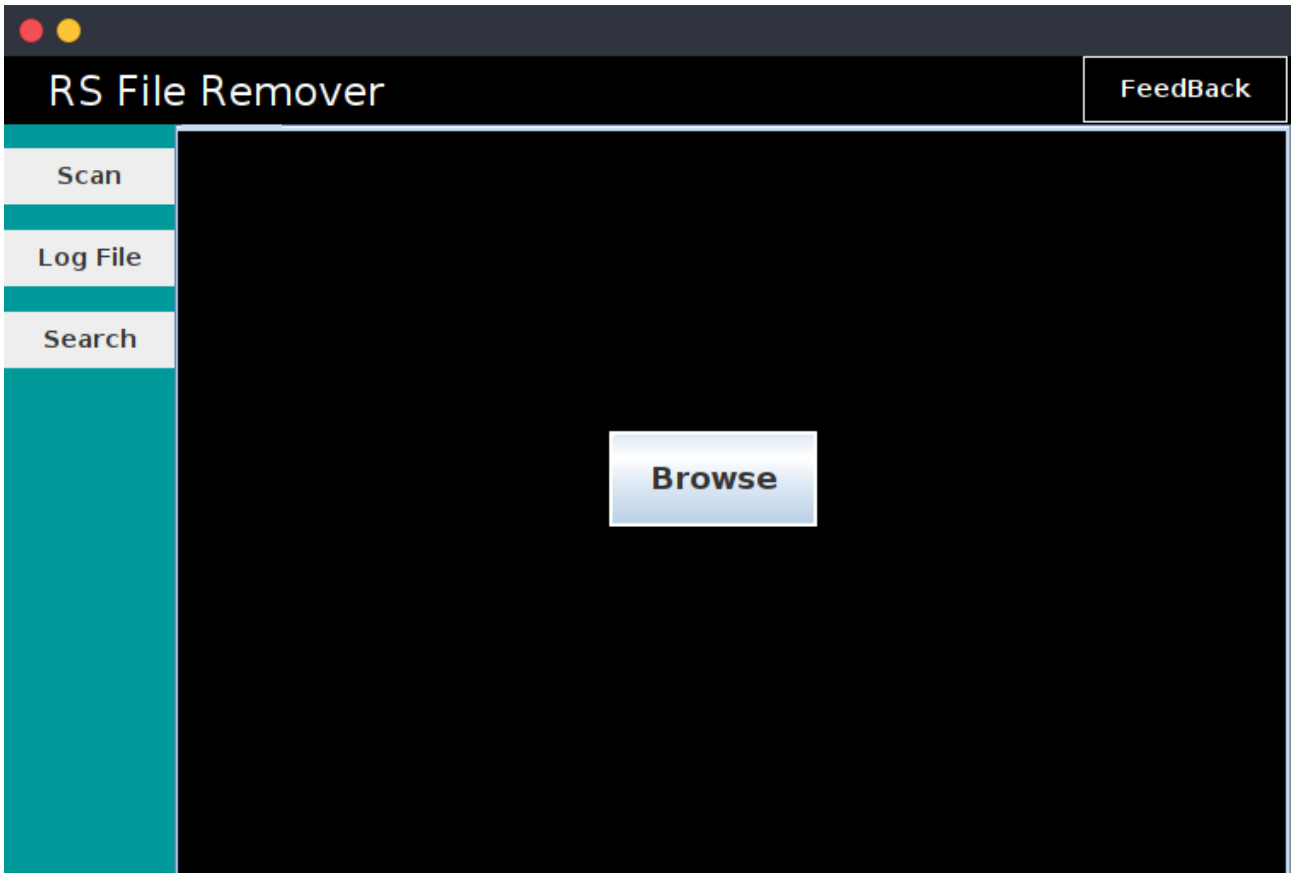
*The user manual and online help will be provided with detailed instructions, and an installation process will be given with an system.*

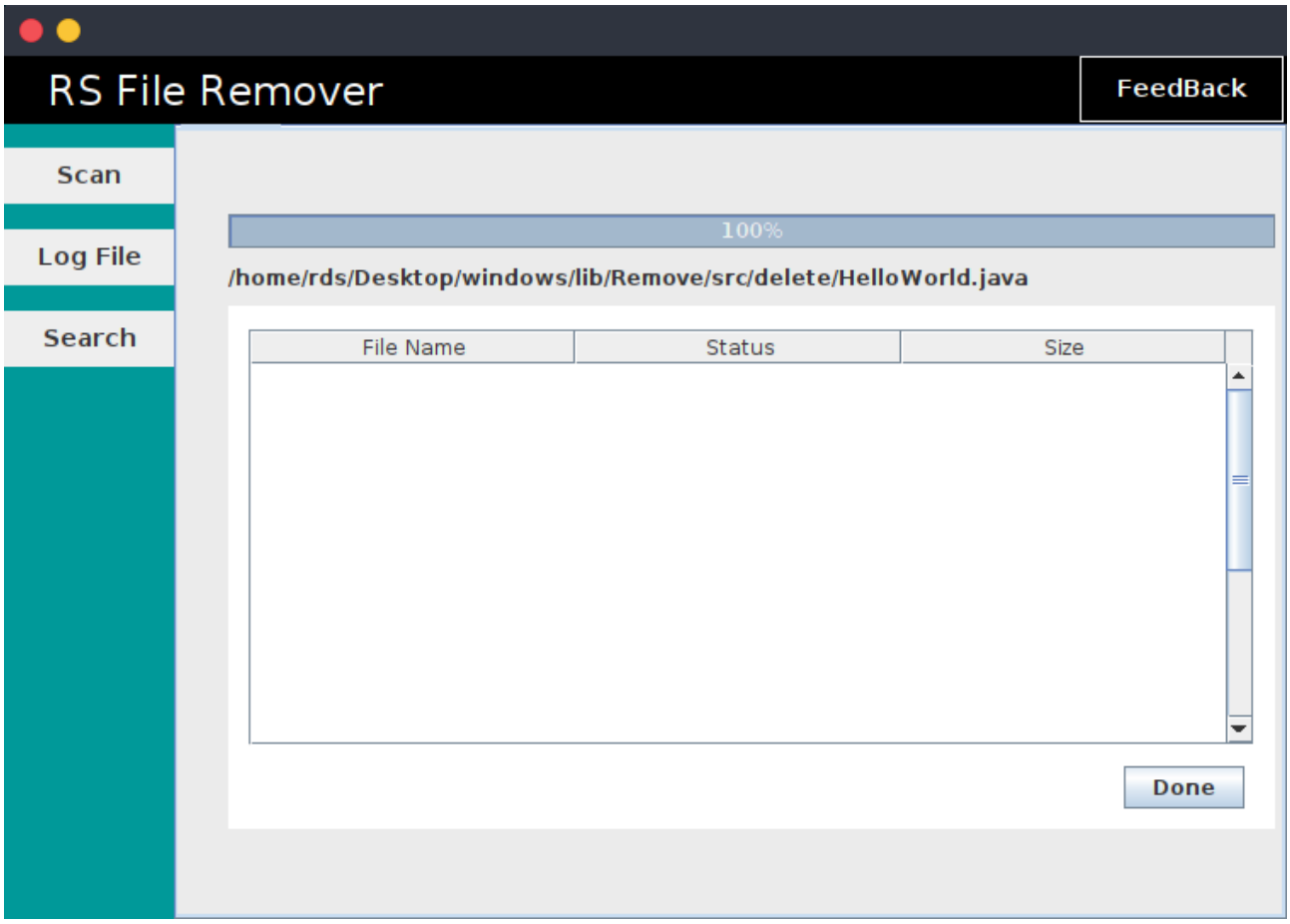
## 2.7 Assumptions and Dependencies

*The user has basic computer knowledge and knows how to operate the software. The software will be used on a system that meets the minimum hardware and software requirements. The user has access to the internet to download the software and updates. The software is dependent on the operating system and must be compatible with it. The software is dependent on the hardware, including CPU, memory, and storage, and must meet the minimum requirements.*
*The software is dependent on third-party libraries and APIs to perform certain functions.*
*The software is dependent on the user having the necessary permissions to perform certain operations, such as deleting files.*

# 3. External Interface Requirements

## 3.1 User Interfaces






## 3.2 Hardware Interfaces

*The software interfaces with the hardware components of the system through various ports and devices, such as USB and hard drives. The software supports different file types, such as text documents, images, videos, and audio files. The data and control interactions between the software and the hardware are managed through communication protocols such as USB, SATA, and Ethernet. The software also requires a minimum processor speed, RAM, and disk space to run smoothly on the hardware.*

## 3.3 Software Interfaces

*The RS Remover software will interface with the operating system and file system of the user's computer. It will also utilize standard programming interfaces and libraries for file management and search functionality. The software will be developed using Java programming language and will require Java Runtime Environment (JRE) version 8 or higher to be installed on the user's system. No other software interfaces or dependencies are expected.*

## 3.4 Communications Interfaces

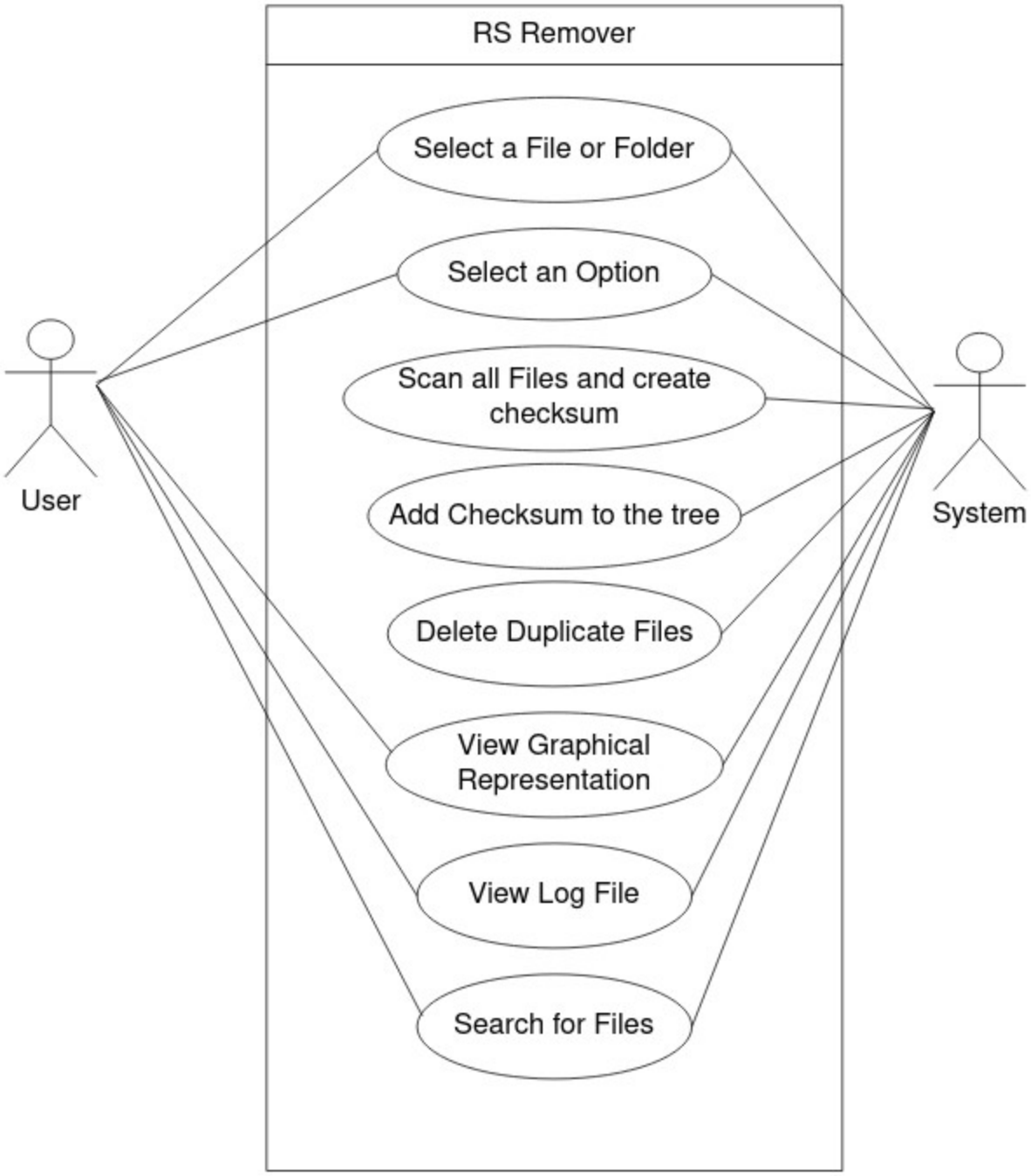
*As RS Remover is a standalone software application, it does not have any communications interfaces with external systems. However, the software may use standard communication protocols such as HTTP or FTP to access external resources.*

# 4. System Features

*<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>*

## 4.1 System Feature 1

- *Search*
- *Delete duplicate file*
- *Log file*
- *Scan*
- *User interface*
- *Support for multiple file formats*
- *Delete Empty File/Folder*
- *Ability to exclude specific folders or files from scanning*
- *Option to save scan results*
- *Fast and efficient scanning algorithm.*


RS Remover
Select a File or Folder
Select an Option
Scan all Files and create checksum
Add Checksum to the tree
Delete Duplicate Files
View Graphical Representation
View Log File
Search for Files
User
System

# 5. Other Nonfunctional Requirements

## 5.1 Performance Requirements

*The system should have a responsive user interface that displays results quickly, with minimal delays. The time taken to search for duplicate files should be optimized and should not take too long. The system should be able to handle large volumes of data and should not crash or hang during processing. The overall performance of the system should meet or exceed the expectations of the end-users.*

## 5.2 Safety Requirements

*As RS Remover is a software application designed for file management, there are no safety requirements associated with it. However, it is recommended that users take appropriate safety measures such as creating backups of their files before using the application to avoid any accidental loss of data.*

## 5.3 Security Requirements

*Security requirements for RS Remover include ensuring that the software does not compromise the security of the user's files or system. The system should be designed to prevent unauthorized access and should comply with standard security protocols. The software should not collect any personal information from the user and should not transmit any data over the internet unless explicitly authorized by the user. Additionally, the system should have an option for secure deletion of files to prevent any possibility of recovery by unauthorized parties.*

## 5.4 Software Quality Attributes

*The software should possess the qualities of usability, reliability, performance, maintainability, portability, and scalability.*

## 5.5 Business Rules

*All design and development rules for the software will be under the jurisdiction of RS Remover.*

# 6. Other Requirements

*Not Applicable*

# Appendix A: Glossary

- *SRS: Software Requirements Specification*
- *GUI: Graphical User Interface*
- *API: Application Programming Interface*
- *OS: Operating System*
- *RAM: Random Access Memory*
- *CPU: Central Processing Unit*
- *HDD: Hard Disk Drive*
- *UI: User Interface*
- *UX: User Experience*
- *HTTPS: Hypertext Transfer Protocol Secure*
- *FTP: File Transfer Protocol*
- *GUI: Graphical User Interface*
- *CLI: Command Line Interface*
- *IDE: Integrated Development Environment*
- *OOP: Object-Oriented Programming*

# Appendix B: Analysis Models

*<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>*

# Appendix C: To Be Determined List

*<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>*