# Software Requirements Specification (SRS)

# ## Project Name: RSA Password Manager

\*\*Version:\*\* 1.0

\*\*Date:\*\* August 12, 2024

\*\*Prepared by:\*\* [Your Name]

---

# ## 1. Introduction

## ### 1.1 Purpose

This document specifies the requirements for the RSA Password Manager. The software will securely manage user passwords using RSA encryption, providing a way to store and retrieve passwords safely.

## ### 1.2 Scope

The RSA Password Manager will allow users to add, retrieve, and delete passwords for various accounts, using RSA encryption to ensure that passwords are stored securely. The application will be a desktop application with a basic graphical user interface (GUI).

## ### 1.3 Definitions, Acronyms, and Abbreviations

- \*\*RSA:\*\* Rivest-Shamir-Adleman, a widely used public key cryptosystem for secure data transmission.

- \*\*GUI:\*\* Graphical User Interface

- \*\*Encryption:\*\* The process of converting information into a secure format that can only be read by someone with the decryption key.

## ### 1.4 References

- [RSA Algorithm Reference]

- [Discrete Mathematics Textbook, Chapter on Cryptography]

---

# ## 2. Overall Description

## ### 2.1 Product Perspective

The RSA Password Manager will be a standalone application running on Windows and macOS. It will use RSA encryption for securing passwords and a local database for storing encrypted data.

## ### 2.2 Product Functions

- \*\*User Authentication:\*\* Secure login to the password manager.

- \*\*Password Storage:\*\* Encrypt and store passwords.

- \*\*Password Retrieval:\*\* Decrypt and display stored passwords.

- \*\*Password Deletion:\*\* Remove passwords from the database.

- \*\*RSA Key Management:\*\* Generate and manage RSA key pairs.

## ### 2.3 User Classes and Characteristics

- \*\*End Users:\*\* Individuals who need to securely store and manage their passwords.

- \*\*Admin Users:\*\* (If applicable) Users who have additional privileges for managing other users' data.

## ### 2.4 Operating Environment

- \*\*Desktop Operating Systems:\*\* Windows 10 or later, macOS Mojave or later

- \*\*Programming Language:\*\* Python or Java

- \*\*Database:\*\* SQLite or local file-based storage

## ### 2.5 Design and Implementation Constraints

- RSA key sizes must be at least 2048 bits for adequate security.

- The application must comply with standard cryptographic practices.

---

# ## 3. Specific Requirements

## ### 3.1 Functional Requirements

\*\*3.1.1 User Authentication\*\*

- Users must authenticate using a master password.

- The master password should not be stored but used to derive encryption keys.

\*\*3.1.2 Password Storage\*\*

- Users must be able to add passwords associated with different account names and descriptions.

- Passwords must be encrypted using RSA before being stored in the database.

\*\*3.1.3 Password Retrieval\*\*

- Users must be able to search and retrieve passwords by account name.

- Encrypted passwords should be decrypted using RSA when retrieved.

\*\*3.1.4 Password Deletion\*\*

- Users must be able to delete stored passwords.

- Deleted passwords should be securely removed from the database.

\*\*3.1.5 RSA Key Management\*\*

- Users must be able to generate and manage RSA key pairs (public and private keys).

- The private key must be securely stored and protected.

## ### 3.2 Non-Functional Requirements

\*\*3.2.1 Performance Requirements\*\*

- The application should encrypt and decrypt passwords in a reasonable time frame, with delays under 1 second for typical operations.

\*\*3.2.2 Security Requirements\*\*

- The system must securely store RSA private keys.

- Encryption and decryption processes must be protected against common attacks (e.g., side-channel attacks).

\*\*3.2.3 Usability Requirements\*\*

- The GUI must be intuitive and easy to navigate.

- The application should provide clear error messages and user guidance.

---

# ## 4. External Interface Requirements

## ### 4.1 User Interfaces

- The application will have a GUI allowing users to interact with the password manager, including forms for adding and retrieving passwords, and dialogs for RSA key management.

## ### 4.2 Hardware Interfaces

- No specific hardware requirements beyond a standard desktop computer.

## ### 4.3 Software Interfaces

- The application will interact with a local database or file system for storing encrypted passwords.

- RSA encryption will be implemented using a standard cryptographic library.

## ### 4.4 Communication Interfaces

- No external network communication required; all data handling will be local.

---

# ## 5. Other Requirements

## ### 5.1 Data Security

- All data stored and transmitted by the application must be encrypted using RSA.

## ### 5.2 Backup and Recovery

- The application should allow users to back up and restore their encrypted data.

## ### 5.3 Documentation

- User documentation should be provided, explaining how to use the password manager and manage RSA keys.