RE-DACT Requirements Document

Software Requirements Specification (SRS)

Project Name: RE-DACT - Secure Redaction Tool

Version: 1.0

Date: September 2024

Author: Big Hero 110

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) defines the functional and non-functional requirements for **RE-DACT**, a user-friendly and secure redaction tool. The tool allows redaction, masking, and anonymization of various input formats on a gradational scale defined by the user, with an additional capability to generate synthetic data for future use. This document will serve as a guide for developers, testers, and stakeholders to ensure the system is built to meet user and security expectations.

1.2 Scope

The RE-DACT tool will:

- Allow users to redact or obfuscate data in text, image, video, and document formats while retaining the structural integrity of the original content.
- Offer a gradational redaction scale that lets users control the degree of redaction/anonymization.
- Generate synthetic datasets based on anonymized data.
- Be available as a web-based tool and an offline desktop version.

- Ensure data security and user privacy by preventing data retention or access by third parties.

1.3 Definitions, Acronyms, and Abbreviations

- **Redaction**: The process of masking sensitive information within a dataset.
- **Anonymization**: Removing or obfuscating information that can identify individuals or sensitive details.
- **Synthetic Data**: Artificially generated data that maintains statistical similarities to real data but contains no actual user information.
- **PoC**: Proof of Concept.
- **COTS**: Commercial Off-The-Shelf.

2. System Overview

RE-DACT is designed to help users handle sensitive information securely by redacting or anonymizing data. The system will cater to various industries like legal, healthcare, and research where the protection of identifiable information is crucial. It will offer both manual and machine-learning-based redaction capabilities.

3. Functional Requirements

3.1 Input and Output Handling

- **Input Support**:
  - Text files (e.g., .txt, .docx)
  - PDF documents
  - Images (e.g., .jpg, .png)
  - Videos (Stage 2)
- **Output Support**:
  - Redacted versions of files (with customizable levels of redaction).

    - Logs of the redaction process.

3.2 Gradational Redaction

- **Redaction Scale**: Users can choose the degree of redaction, from simple anonymization (e.g., removing names) to complete obfuscation where even the correlational logic is masked.
- **Annotation**: Display a preview of the redacted data with the selected level of redaction.

3.3 Synthetic Data Generation

- The system will provide options for generating synthetic datasets that mirror the original structure but contain no identifying information.

3.4 User Interface (UI)

- The system will have an intuitive graphical user interface (GUI) for both novice and advanced users, available in:
    - **Web-based interface**: Accessible from modern browsers.
    - **Offline desktop version**: Available for secure environments without internet access.

3.5 Security Features

- **Minimum Data Retention**: No input data will be stored or retrievable by any third-party service.
- **User Control**: Users will maintain full control of their data throughout the redaction process.
- **Secure Coding**: Secure coding practices will be used to prevent unauthorized access and ensure the integrity of redacted files.

3.6 Machine Learning (ML) Capabilities

- In future versions, the tool will learn from user interactions to improve the efficiency and accuracy of redaction.

- Training will be conducted on publicly available datasets to refine the tool's redaction capabilities.

4. Non-Functional Requirements

4.1 Security

- **Data Encryption**: All data will be encrypted during the redaction process, whether in transit or at rest.
- **Minimal API Dependencies**: Ensure minimal external API dependencies to reduce potential security vulnerabilities.

4.2 Performance

- **Speed**: The system should provide redacted outputs within a reasonable timeframe, optimizing for large datasets and multimedia files.
- **Scalability**: The tool must handle increasing amounts of data and diverse input formats without degradation in performance.
- **Benchmarking**: Performance should be benchmarked against existing COTS solutions.

4.3 Usability

- **Ease of Use**: The tool should provide a clean, intuitive interface for users to easily adjust the redaction scale and process files.
- **Accessibility**: The UI should be accessible to users with different technical abilities.

4.4 Offline and Online Availability

- The tool will function in both online and offline environments, with the same core features available in each mode.

4.5 Compliance and Legal

- The tool must comply with data protection regulations (e.g., GDPR, HIPAA) to ensure that the redacted data cannot be reverse-engineered.

5. System Architecture

5.1 Overview

The system architecture will consist of:

- A frontend for user interactions.
- A backend handling file processing and redaction logic.
- A machine learning component responsible for the gradual improvement of synthetic data generation and redaction accuracy.

6. Constraints

- The system must operate in secure environments with minimal external dependencies.
- Offline versions must support full functionality without internet access.
- Synthetic data generation must be efficient while ensuring no identifiers remain.

7. Assumptions and Dependencies

- The tool assumes a diverse set of input formats will be supported over time, starting with text and document files, expanding to images and videos.
- It depends on publicly available datasets for training and testing the machine learning models.