Tribhuvan University
Faculty of Humanities and Social Science

File Compression & Malware Detection System

A PROJECT PROPOSAL

Submitted to
Department of Computer Application
Mechi Multiple Campus

In partial fulfilment of the requirements for the Bachelor in Computer Application

Submitted by

Santosh Bhandari - 58

July 2023

Under the Supervision of
Raju Poudel

Table of Contents

1. Introduction

In the digital age, the efficient storage and secure handling of data are crucial. With the increasing volume of data generated daily, file compression has become a necessary tool for conserving storage space and optimizing data transmission. However, alongside the benefits of digital file-sharing comes the risk of malware infections. Malware, or malicious software, can cause significant damage to systems and compromise sensitive information. Therefore, integrating a robust malware detection mechanism with file compression processes can provide a dual benefit of storage efficiency and enhanced security.

The "File Compression and Malware Detection System" project aims to address these needs by developing a web-based application that enables users to upload files for compression while ensuring they are free from malware. This system will scan the uploaded files against a comprehensive database of known malware signatures. If any malicious content is detected, the system will issue a warning and halt the compression process, thereby safeguarding the user's data and system integrity. Conversely, if no malware is found, the file will be compressed and made available for download, providing a seamless and secure file-handling experience.

2. Problem Statement

In today's digital era, individuals and organizations generate and handle vast amounts of data daily. Efficient data management practices, such as file compression, are essential for conserving storage space and enhancing data transmission speeds. [1]

The Problem of the Statement are as follows:

- Efficient data management, such as file compression, is essential for conserving storage space and optimizing data transmission.
- People don’t have strong knowledge to identify Malware.
- Difficulty in maintaining accurate and up-to-date malware records.

3. Objectives

The Objectives are as follows:

- To Develop a system to compress files efficiently,

- To Develop a robust malware detection system that scans uploaded files for known malware signatures before compression.

4. Methodology

Here we are going to use the Iterative waterfall methodology while building this project. Since this model has specific features and technical clearance. We are implementing this method to make our project successful.[2]

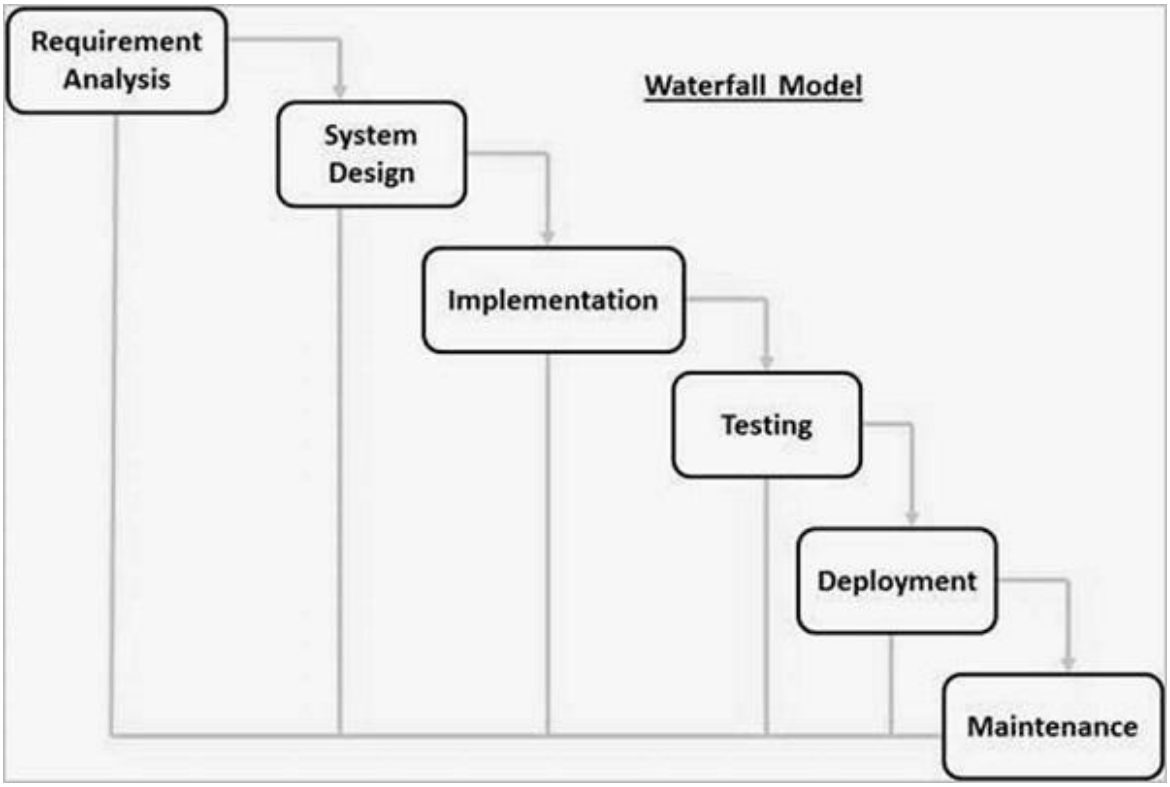

Fig: IterativeWaterfall Model

Why IterativeWaterfall Model?

The Iterative Waterfall Model was chosen for the "File Compression and Malware Detection System" project due to its structured approach and flexibility. This model allows for clear, sequential phases with the opportunity to revisit and refine each phase based on feedback and testing results, ensuring continuous improvement and thorough risk management. The iterative nature of the model also facilitates regular stakeholder involvement, allowing for feedback to be incorporated throughout the development process.

The Iterative Waterfall Model also promotes detailed documentation and systematic progress tracking, which are essential for managing a project as complex as the "File

Compression and Malware Detection System." This meticulous approach ensures that every aspect of the system, from file compression mechanisms to malware detection algorithms, is thoroughly planned, developed, and tested. By incorporating regular feedback loops and iterations, the model ensures that any issues are promptly addressed, leading to a more polished and reliable final product.

a. Requirement Identification

Requirement identification is a crucial step in developing the File Compression & Malware Detection System, as it lays the foundation for the design and development of the software solution. The process involves studying the existing system, if any, and collecting the requirements to address the malware-detecting needs effectively.

I. Study of Existing System

The first step in requirement identification for the proposed system is to conduct a thorough study of the existing system. Here, I have considered two existing Systems i.e. Virus Total and Hybrid Analysis.

Virus Total:

VirusTotal is a widely used online service that analyzes files and URLs for viruses, worms, trojans, and other types of malicious content. It aggregates the results from various antivirus engines, website scanners, and other tools to provide a comprehensive analysis of the submitted content.

Pros:

- The System provides aggregate results from multiple antivirus engines and tools.
- It continuously updates its databases with new malware signatures.
- It also provides API for integration into other applications.

Cons:

- Files submitted for analysis are shared with various antivirus vendors.
- The free version provides only a limited number of scans.

Hybrid Analysis:

Hybrid Analysis is a malware analysis service that offers both static and dynamic analysis of files. It provides detailed reports on the behaviour and characteristics of

suspicious files, helping users to identify potential threats and understand their impact.

Pros:

- It provides comprehensive reports of programs and files.
- It allows users to apply custom YARA rules.

Cons:

- The free version has restrictions on file size.
- The detailed analysis may sometimes result in false positives.

II. Requirement Collection

After studying the existing system, the next step is to collect the requirements for the System. This involves gathering information about the desired features, functionalities, and objectives of the new system. Requirement collection can be done through various techniques, including interviews, surveys, and meetings. [3]

During the requirement collection phase, the following aspects are considered:

- File Upload: Determining what types and sizes of files users can be allowed to upload.
- Malware Detection: The system should accurately detect and identify malware in the uploaded files and establish a comprehensive database of malware signatures.
- File Compression: Compress files only if they are free of malware.

During the requirement collection phase, it is important to involve all relevant people to ensure that their needs and expectations are captured accurately. The requirements will serve as the foundation for the design and development of the final System, ensuring that it meets the specific needs and improves upon the limitations of the existing system.

b. Feasibility Study

Before proceeding with the development of the System, a feasibility study is conducted to assess the viability and potential success of the project. This study evaluates various aspects, including technical feasibility, economic feasibility, and operational feasibility, to determine if the project is worth pursuing. [4]

I. Technical

Technical feasibility assesses whether the development and implementation of the proposed system are achievable using the available technology and resources. It involves evaluating the compatibility of the software solution with the existing infrastructure, hardware, and software systems.

Parameters	Remarks
Hardware Resources	Available
Technical Expertise	Yes
Programming Language	Free
Software Tools	Open Source / Free

II. Operational

Operational feasibility assesses whether the proposed System can be smoothly integrated into the existing operations and processes. It focuses on evaluating the system's compatibility with workflow and the acceptance of the system by users.

Parameters	Remarks
User Acceptance	Yes
Ease of Use	Yes
Training Requirement	Yes
Operational Cost	Affordable

III. Economic

Economic feasibility assesses the financial viability and benefits of developing and implementing the proposed System. It involves analyzing the costs associated with the project and comparing them with the potential benefits and returns on investment.

Parameters	Remarks
Development Tools	Open Source / Free
Servers	Free
IDE	Open Source / Free
Programming Language	Free

Based on the findings of the feasibility study, it can be concluded that the proposed System can be developed with a high level of confidence.[5] The operational feasibility assessment suggests that the proposed system can be seamlessly integrated into the existing operations and can be easily operated. Furthermore, the economic feasibility study demonstrates that the proposed system can be developed cost-effectively. Considering these positive outcomes, it is evident that the proposed System has the potential to address the billing needs effectively and improve efficiency, accuracy, and financial management.

c. High Level Design of System

For a better understanding of the system components and functionalities, The System flow chart is presented below:

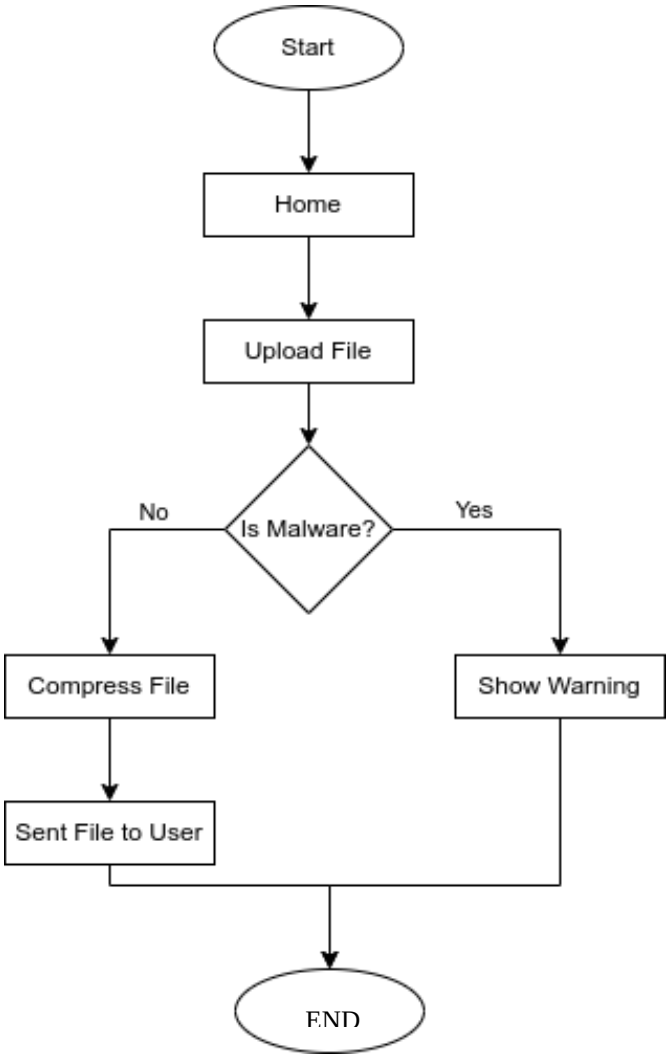

Fig: System Flowchart

5. Gantt Chart

The Gantt chart showing the work schedule is given below:

ID	Name	Start	Finish	Duration
1	Requirement Collections	6/27/2024	7/2/2024	4 day
2	Analysis	7/3/2024	7/10/2024	6 day
3	System Design	7/11/2024	7/24/2024	10 day
4	Coding	7/25/2024	9/11/2024	35 day
5	Testing	8/22/2024	9/12/2024	16 day
6	Deployment & Maintenance	9/13/2024	9/19/2024	5 day
7	Documentation	7/15/2024	9/13/2024	45 day

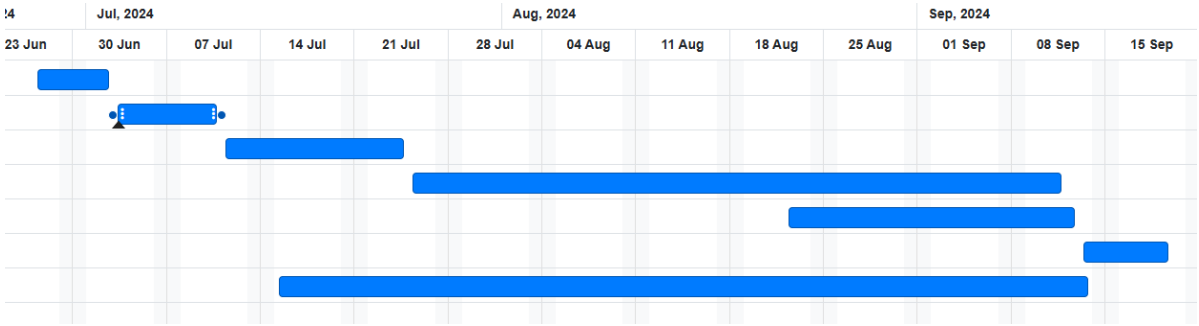

Fig: Gantt Chart

6. Expected Outcome

The "File Compression and Malware Detection System" is expected to deliver a comprehensive solution that integrates efficient file compression with advanced malware detection. Users will benefit from a streamlined process where they can easily upload files, receive immediate feedback on the presence of malware, and compress files securely. By automating the malware scanning before compression, the system will prevent the risk of malware infections and ensure that only clean files are compressed and stored. This integration enhances both security and efficiency, reducing manual intervention and minimizing the risk of data breaches.

Additionally, the system will provide a user-friendly interface that simplifies the file upload and compression process, making it accessible even to those with limited technical expertise. The project aims to deliver a reliable and scalable solution, capable of handling a large number of users and file uploads while maintaining high performance and security standards. Comprehensive documentation will support users

and developers, ensuring that the system is easy to use, maintain, and extend. Overall, this project will contribute to improved data management practices and enhanced security in digital file handling.

7. References

[1] Samir, "The Top Security Risks of File Sharing and How to Avoid Them," Time Tracko, [Online]. Available: https://timetracko.com/blog/the-top-security-risks-of-file-sharing-and-how-to-avoid-them/.

[2] "SDLC - Waterfall Model," Tutorials Point, [Online]. Available: https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm.

[3] T. Asana, "A 6-step guide to requirements gathering for project success," Asana, 31 January 2024. [Online]. Available: https://asana.com/resources/requirements-gathering.

[4] J. Bridges, "What Is a Feasibility Study? How to Conduct One for Your Project," Project Manager, 19 April 2023. [Online]. Available: https://www.projectmanager.com/training/how-to-conduct-a-feasibility-study.

[5] "Technical Feasibility," Launch Notes, [Online]. Available: https://www.launchnotes.com/glossary/technical-feasibility-in-product-management-and-operations.