### **Email Scanning System**

### 

### **Palthi Malleswari**

**RGUKT-Nuzvid,CSE**

[**n190878@gmail.com**](mailto:n190878@gmail.com)

**N190878,CSE-01**

**Guide: KKSing Sir**

**Edubot**

### **Table of Contents**

1.Introduction

Project Objective

2.Functional Requirements

3.Components and Explanation of project

4.Outcomes

5.Application Outputs

6.Conclusion

### 

### **Introduction:**

### **Project Objective**

The objective of this project is to develop an Email Scanning System that can efficiently process and scan emails for potential threats using keywords. The system is equipped with functionalities to manage keywords, scan emails, provide a user-friendly GUI, and generate reports. Additionally, it incorporates security measures and thorough testing to ensure reliability and safety.

### **2. Functional Requirements**

1. **Keyword Management:**
   * Add keywords with associated weights.
   * Update existing keywords and weights.
   * Remove keywords.
   * Display the list of current keywords and their weights.
2. **Email Scanning:**
   * Scan emails to calculate a suspicion percentage based on the presence of keywords.
   * Display scan results.
3. **Database Interaction:**
   * Store and retrieve email records, scan results, and keywords using MySQL.
4. **Graphical User Interface:**
   * A Swing-based desktop application for managing keywords and scanning emails.
   * Forms and buttons for keyword management and email scanning.
5. **Reporting and Analytics:**
   * Generate detailed reports on email scan results.
   * Visualise data using charts and graphs.
   * Export reports in PDF format.
6. **Web Application:**
   * Deployed on TomcatServer.
   * Secure session management.
   * Exception Handling
7. **Testing and Debugging:**
   * Unit tests to ensure the correctness of individual components.

### **3. Explanation of the Project**

#### **Components**

1. **Database:**
   * Tables for storing emails, scan results, and keywords.
   * JDBC for database connectivity and CRUD operations.
2. **Backend Logic:**
   * KeywordDAO: Model Manages database operations for keywords.
   * userDAO: Model Manage database operations for user activities
   * Keyword and User : Utility Classes
   * AddKeywordServlet: to handle requests of new keyword addition to db
   * updateKeyWordServlet: to handle requests to update a given keywords’s weight
   * deleteKeywordServlet: to handle deletion of keywords
   * viewKeyWordSErvlet: to view all the current keywords in db
   * Home.html : Home Page for EmailScanningSystem
   * addKeyword.html : WebPage to add keywords
   * updateKeyword.html: WebPage to update the keywords
   * deleteKeyword.html: to delete keywords
   * viewKeyWord.html: to display current Keywords
   * ExceptionHandling: to handle all the DB,Runtime exceptions.
   * KeyWordsWeights: to store predefined keywords of from db
   * EmailScanner: Main Class for Console operations for Email Scanning and to handle multiple email requests for scanning
   * DatabaseUtil: A utility class to connect to the MySQL
   * EmailScannerTests: to Run all the Test Cases
   * EmailReports: to generate email scanning reports using JFree data visualisation
   * MainFrame : Desktop Application Home Page
   * ManageKeywordsFrame: to manage all buttons for key management
   * SuspicionCalculator: Calculates the suspicion percentage for email content.
   * ScanEmail: Integrates keyword management and email scanning functionalities.
3. **Graphical User Interface:**
   * Main application window for keyword management and email scanning.
   * Forms and buttons for user interactions.
4. **Reporting:**
   * EmailReport: Generates and visualises reports.
   * JFreeChart for data visualisation.
5. **Testing:**
   * JUnit test cases to ensure functionality and reliability.

### **4. Outcomes**

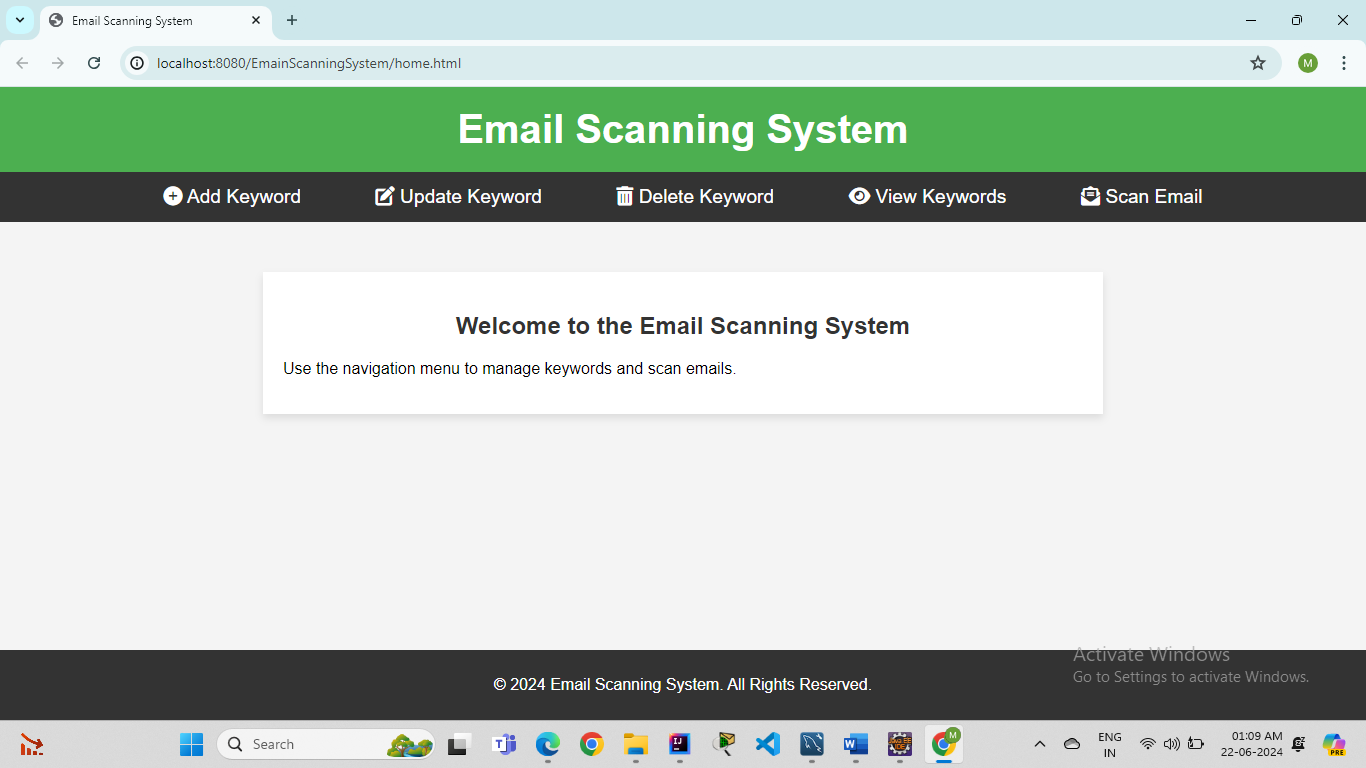
1. **Effective Email Scanning:**
   * Successfully implemented email scanning based on keyword weights.
   * Accurate calculation of suspicion percentage.
2. **User-Friendly Interface:**
   * Intuitive GUI for managing keywords and scanning emails.
3. **Detailed Reporting:**
   * Comprehensive reports with visual data representation.
4. **Security:**
   * User authentication and data encryption implemented.
   * Secure session management to prevent vulnerabilities.
5. **Robust Testing:**
   * Developed unit tests to ensure component functionality.

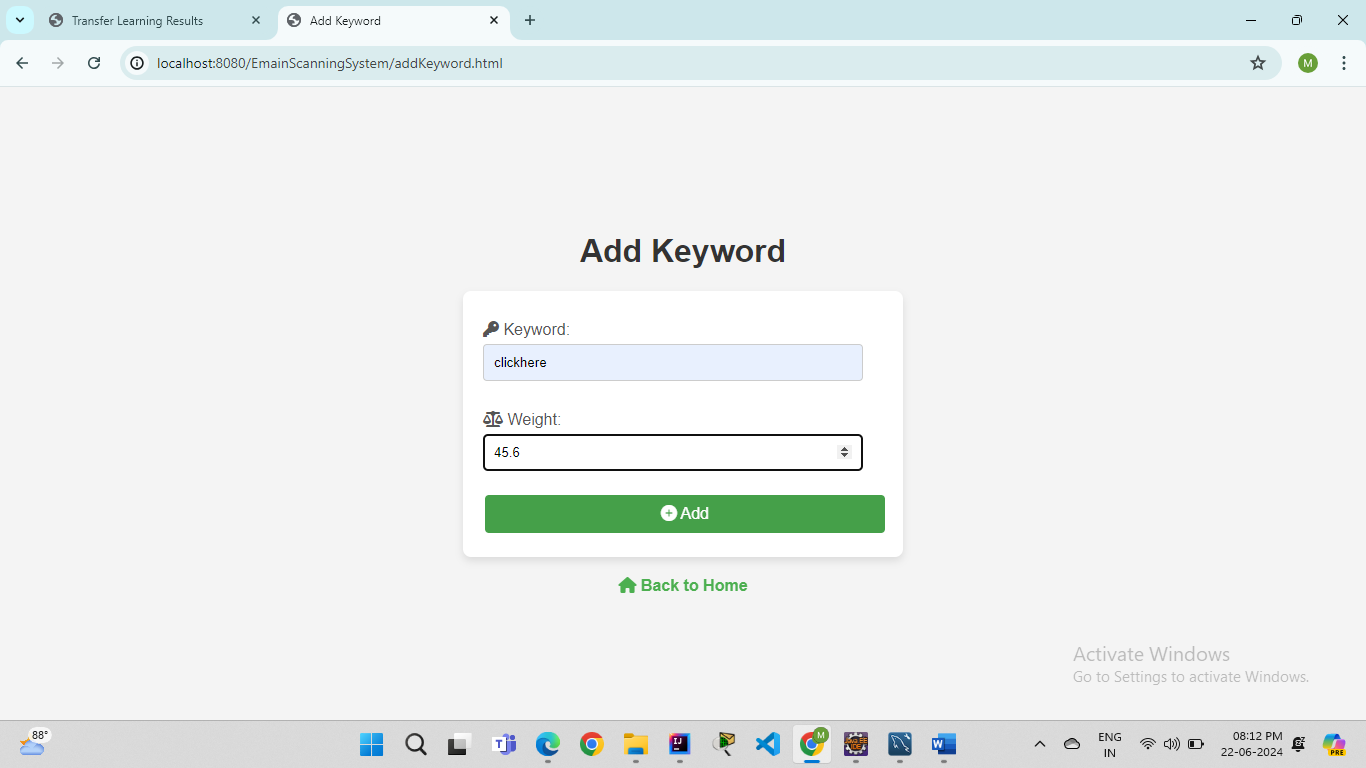
**5.Application Output :**

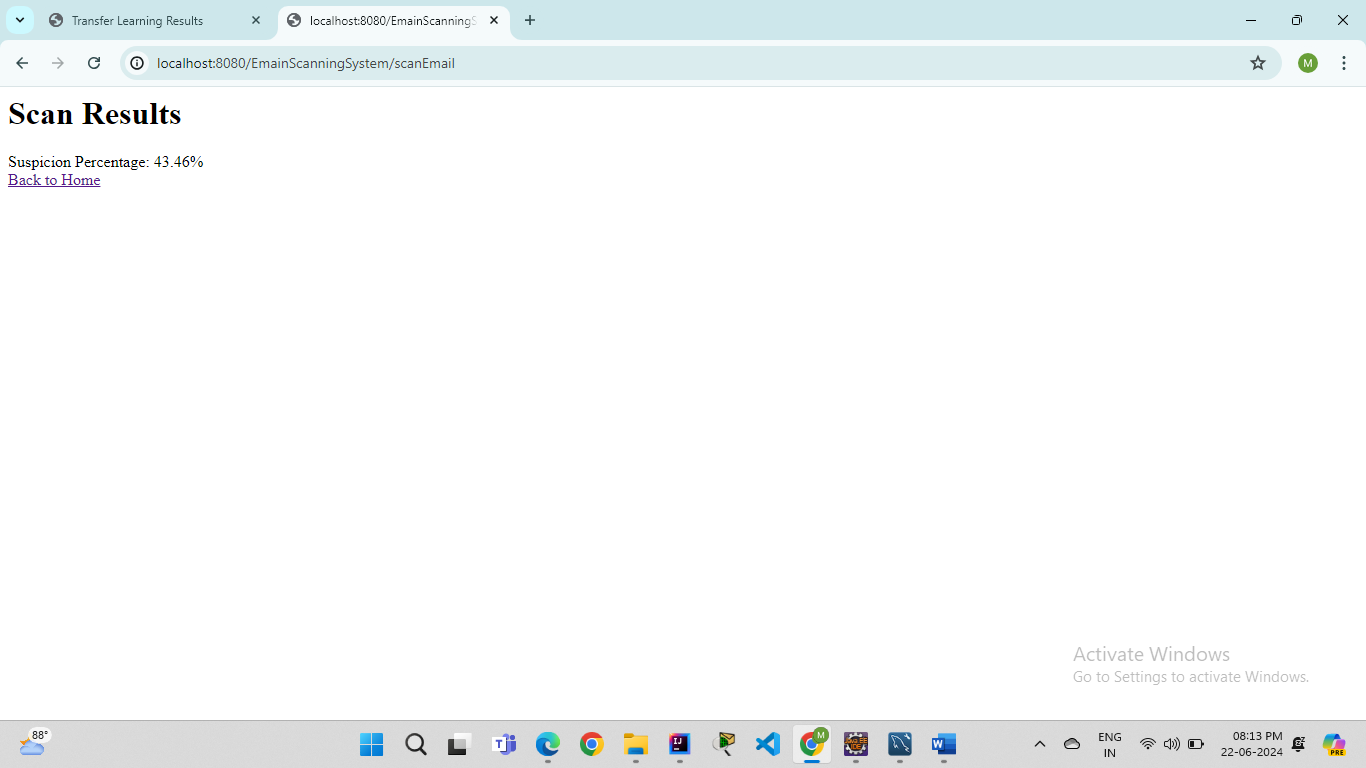
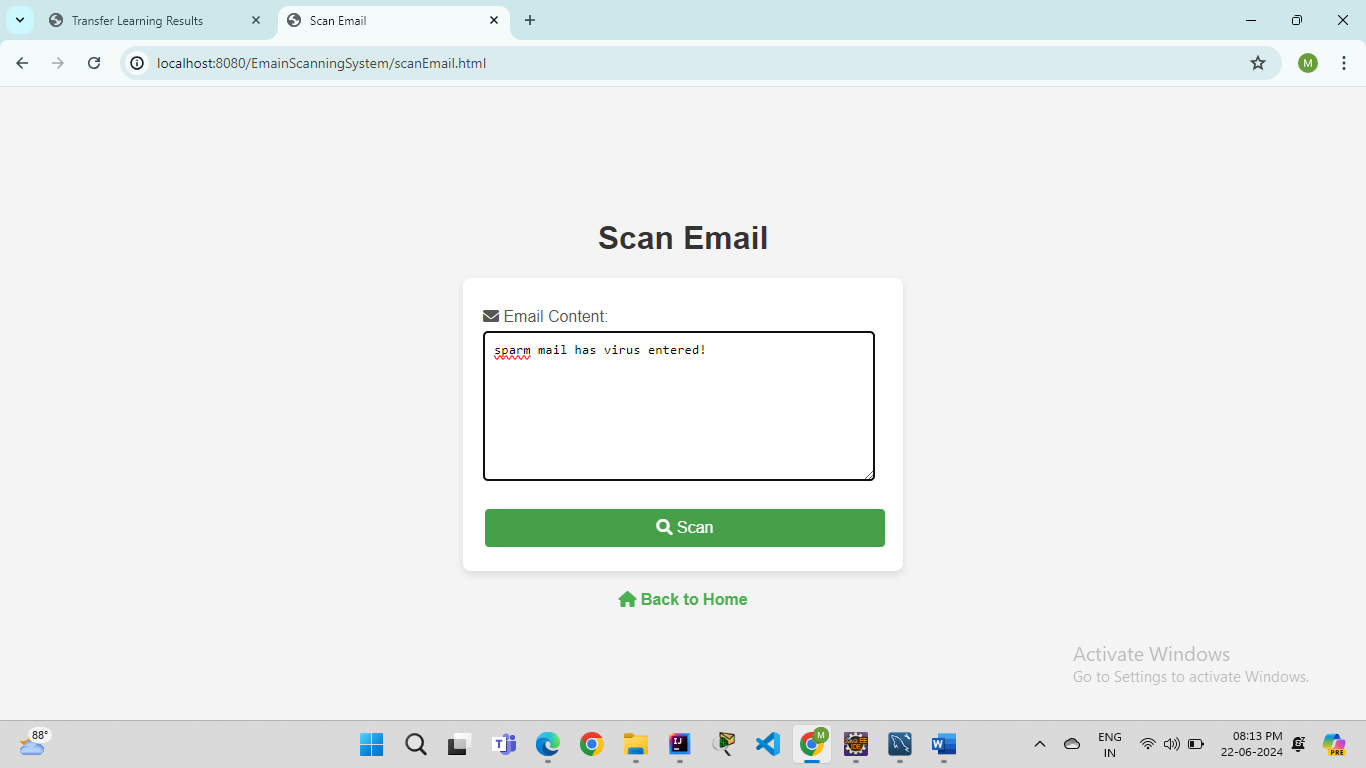
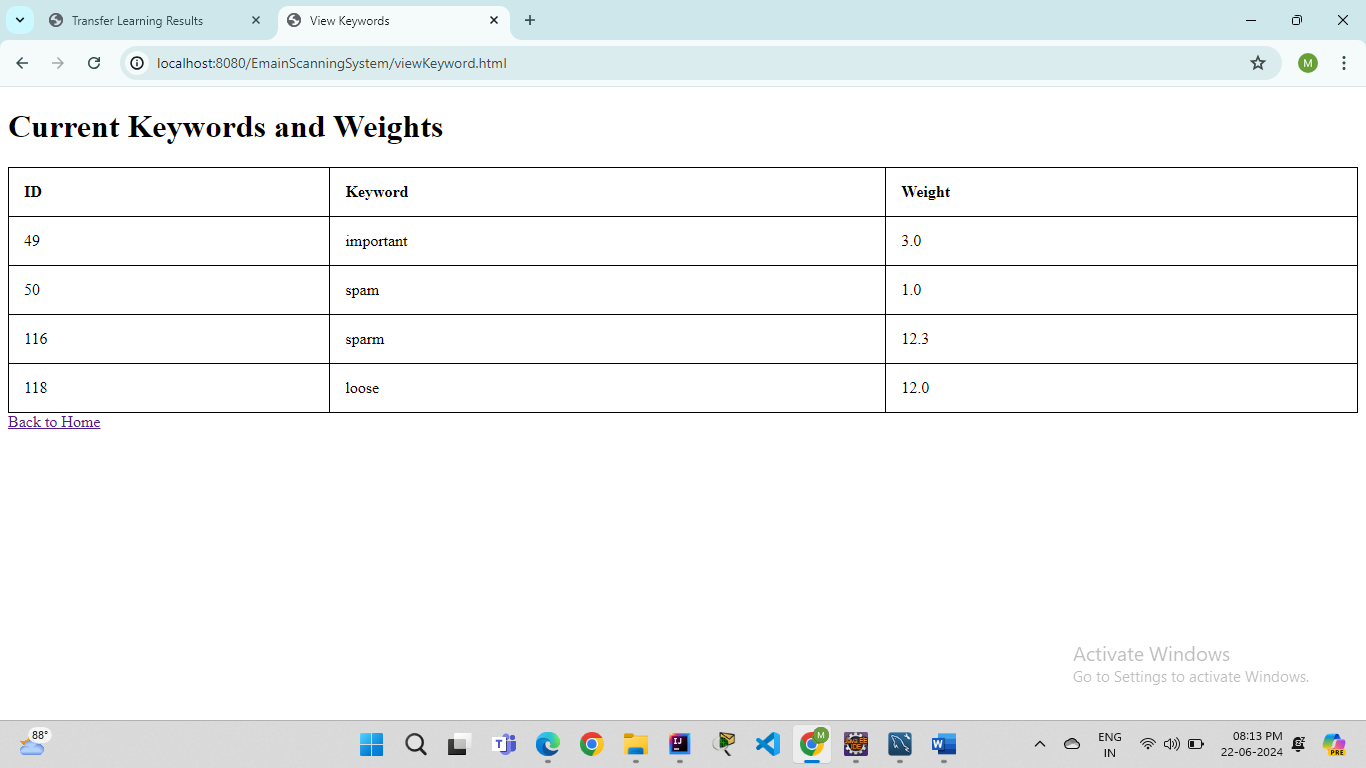
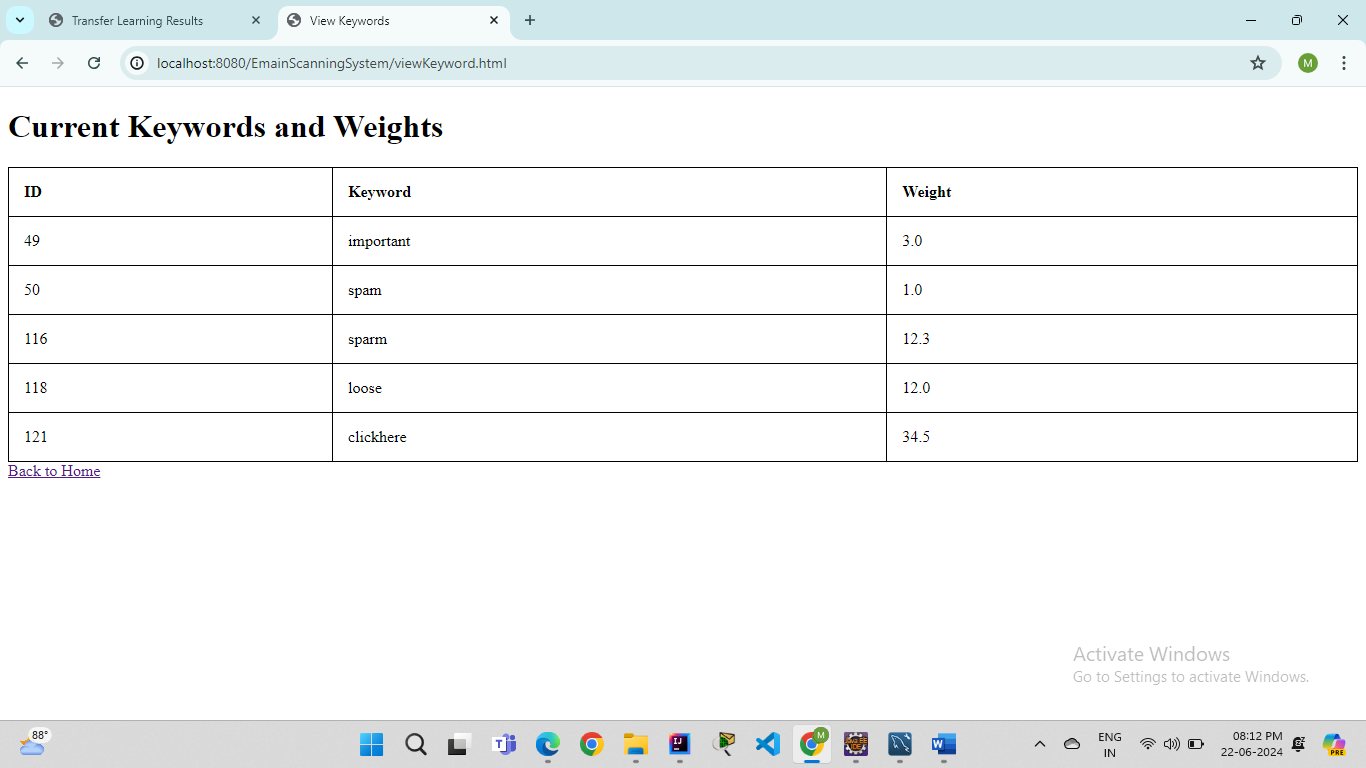
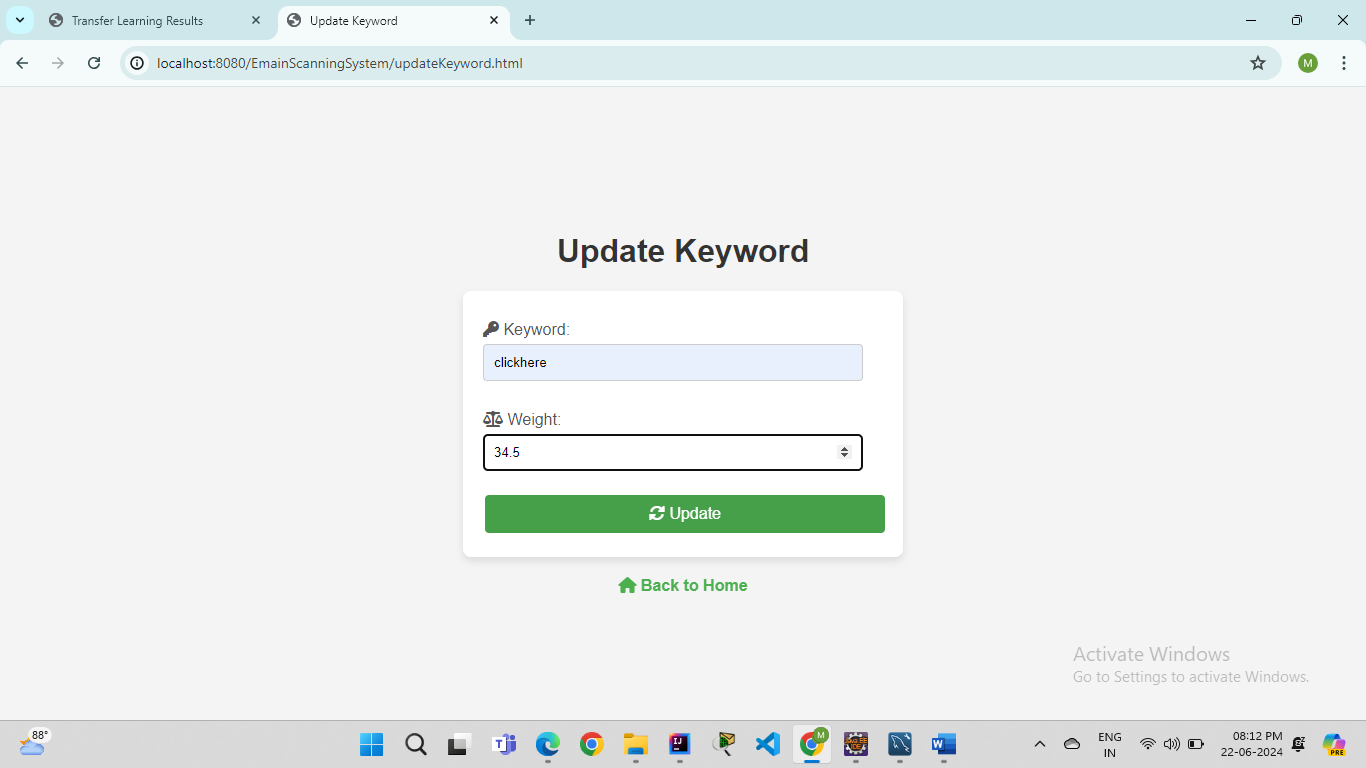
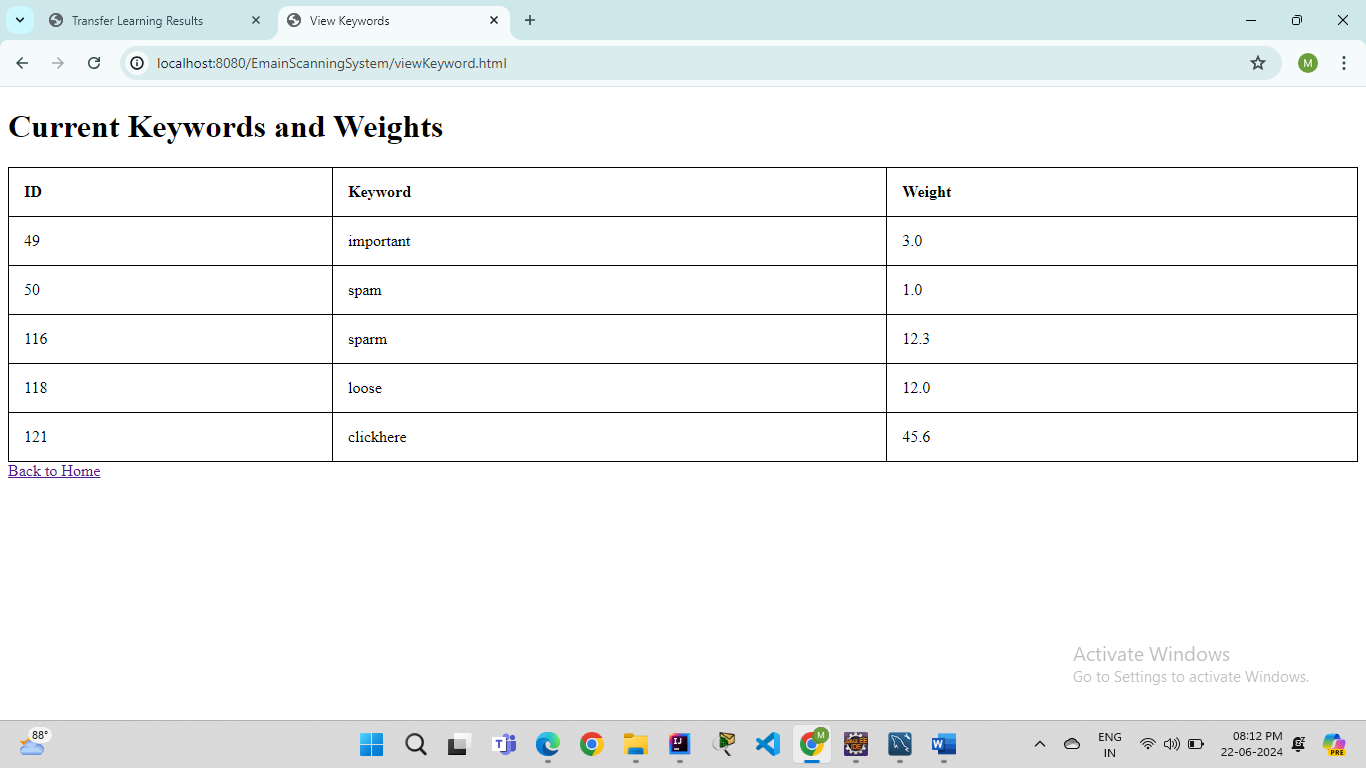
**Analysis of Email & Reports**

****

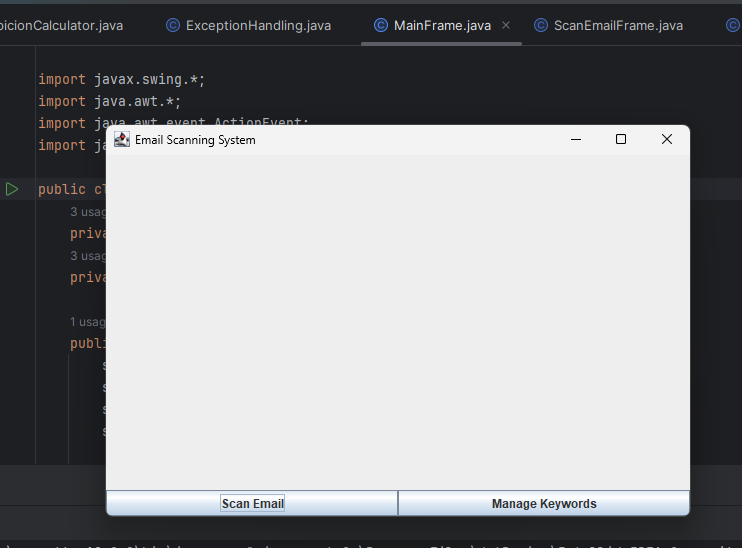
**Web Application Home Screen :**

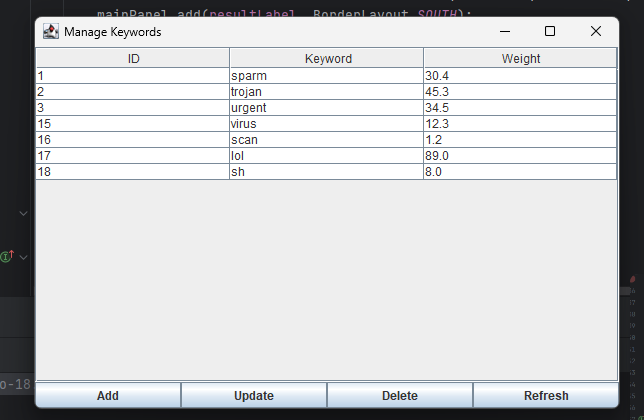
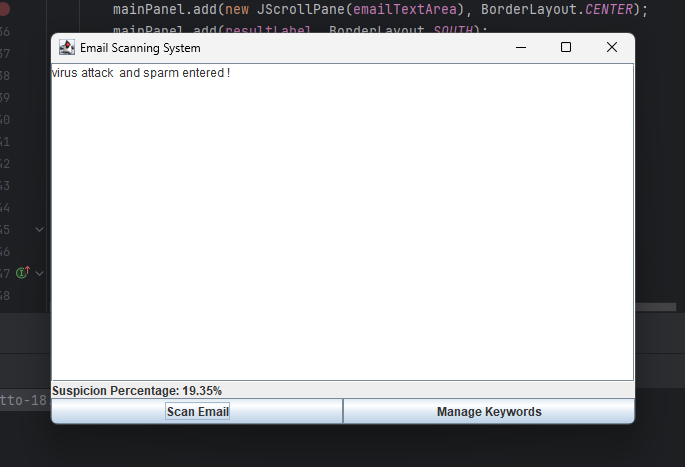
****

****

****

**Desktop Application :**

****

****

### **6. Conclusion**

The Email Scanning System project successfully meets its objectives by providing a comprehensive solution for managing and scanning emails for threats. The system is user-friendly, secure, and thoroughly tested, making it a reliable tool for email management and threat detection.