프로젝트 식별 기술 정리서

팀 Genesis (고재현, 심수민, 전영우)

1) 기술 스택 **(Tech Stack)**

1. 프론트 엔드
   - React 대시보드/차트(인증/연동: OAuth 시작/콜백, 연결 상태 표시)
2. 백엔드
   - Spring Boot : 백엔드 (API, 계정 연동, 정책 관리, DB 연동)
   - 데이터베이스
     MySQL : 사용자 계정 정보, 파일 해시값, 이메일 데이터

2) 기술적 요구사항

**** 기능적 요구사항 ****
F1. 악성 코드 감염 가능성 검증 기능
FR-001. 브라우저에서 이메일 파일 또는 첨부파일을 업로드한다.
FR-002. 이메일 제목과 본문을 분석한다
FR-003. 피싱 또는 스미싱 가능성에 따라 안전, 의심, 경고 표시를 한다.
FR-004. 첨부파일을 정적으로 분석, 조회(VirusTotal API)등을 활용하여 악성여부를 판별한다.
FR-005. 받은편지함 UI에서 각 메일 항목에 안전/의심/위험 배지를 표시해야 한다.

*** 품질 요구사항 ****

1. 운영 환경
NF-001 시스템은 WINDOWS 및 MAC OS 환경에서 동작할 수 있어야 함.
NF-002 다양한 종류의 이메일 서비스(네이버 메일, gmail, 다음 메일 등)에서 적용이 가능해야 한다

2. 성능
NF-003 시스템 응답시간은
NF-004 시스템은 24시간동안 작동할 수 있어야 한다.
NF-005 오류는 1초 이내에 제시되어야 한다.
NF-006 동시 사용자수 1000명 이상 지원 및 성능이 저하되지 않아야 한다.

3. 보안 요구사항
NF-007 사용자 역할에 따른 자료 접근이 차별화 되어야 한다.
NF-008 사용자 로그인 기능이 제공되어야 한다.

4. 문화 및 정책적 요구사항
NF-009 개인정보가 공개되지 않도록 각별히 주의한다.

NF-010 시스템은 한글 및 영어를 지원하여야 한다.
NF-011 정보통신망 이용촉진 및 정보보호 등에 관한 법률을 준수한다.
NF-012 개인정보 보호법을 준수해야 한다.

** 인터페이스 요구사항 **
1. 사용자 인터페이스 요구사항
IR-001 시스템은 윈도우 MAC OS 기반의 운영 환경을 제공해야 한다.
IR-002 시스템은 메일 검사 결과(정상/의심/악성)를 직관적인 색상(녹색/주황/적색)으로 표시하여야 한다.

IR-003 시스템은 악성으로 판별된 메일의 상세 분석 근거(첨부파일 해시, 탐지 엔진 결과)를 조회할 수 있어야 한다.
IR-004 시스템은 관리자에게 의심 메일 격리/복원 기능을 제공해야 한다.

2. 외부 시스템 인터페이스 요구사항

IR-005 시스템은 Virus Total API과 연동하여 이메일 내용 및 첨부파일을 조회해야 한다.
IR-006 시스템은 Gmail API를 통해 메일 원문 및 첨부파일을 수집해야 한다.
IR-007 시스템은 네이버 메일 IMAP 프로토콜을 통해 메일 데이터를 수집해야 한다.

3) 기술 검토

1. 타당성, 관련성 (사업/제품 목표와 얼마나 부합하는가)

이메일 악성코드 감지 프로그램은 사용자가 이용하는 메일 서비스의 수신 메일을 대상으로 분석 자동화 및 차단/알림 기능을 제공하여 악성코드가 사용자의 PC에서 실행되는 것을 방지할 수 있다.

2. 실현 가능성 (필요 인력, 장비, 예산)

이메일 악성코드 감지 프로그램의 개발은 JAVA (Spring Boot 기반 백엔드 개발 (API, 계정 연동, 정책 관리, DB 연동), MySQL 등의 기술을 요구한다.
조원들의 개발 경력을 고려하면, C, C++, python, java, kotlin, html, 스프링부트, 리액트 등의 풍부한 경험이 있는 것으로 보아 충분히 개발할 수 있을 것으로 예상된다.

3. 위험도 분석 및 관리

- 기술적 위험
  기술의 실현 가능성을 저해하는 주요 장애물은 실시간 메일 제공 유무이다.
  따라서 각 이메일 서비스에서 실시간으로 수신되는 메일을 제공받을 수 있는지 확인이 필요하다.

VirusTotal API의 버전 업데이트에 의해 서비스 제공에 문제가 발생할 수 있다. 따라서 VirusTotal API 버전에 따라 유지보수가 가능해야한다.

- 위험 관리 방안(기술적 위험)
  기술적 측면에서는 메일 동기화 복잡성을 줄이기 위해 초기에는 Gmail, Outlook 등 주요 서비스 위주로 구현하고, 오탐·미탐 문제는 룰 기반 탐지와 ML, 그리고 VirusTotal 교차검증을 병행한다.

  외부의존성(유지보수 부담)
  오픈소스 룰셋 자동 업데이트 스케줄러 적용하고고, 정기 점검 일정을 수립한다.