# Tài liệu yêu cầu phần mềm (SRS)

**Tên dự án:** PhishingDetect  
**Ngày:** 23/09/2025  
**Người viết:** Team tụi mình

## 1. Giới thiệu

### 1.1 Mục đích

Bọn mình muốn làm một web app (dùng Django) để giúp người dùng kiểm tra xem email Gmail của họ có bị lừa đảo (phishing) không. Ngoài ra cũng có thể nhập một đường link (URL) hoặc tải lên file đính kèm để kiểm tra.

### 1.2 Phạm vi

App sẽ:  
- Quét hộp thư Gmail (sau khi người dùng cho phép).  
- Cho người dùng nhập một URL để check.  
- Cho người dùng upload hoặc kéo thả file để check.  
- Có phân tích file/link bằng nhiều cách: phân tích nhanh, chạy trong sandbox (kiểu môi trường giả lập).  
- Dùng AI/ML để phát hiện và giải thích vì sao kết quả là phishing hay không.  
- Có màn hình quản lý cho người dùng và admin.

### 1.3 Giải thích vài từ

* **Sandbox:** môi trường giả lập để chạy file/link nguy hiểm cho an toàn.
* **ExplainAI:** AI tạo lời giải thích bằng tiếng Việt hoặc tiếng Anh để người dùng hiểu.

## 2. Mô tả chung

* Đây là web app chạy bằng Django, có API REST.
* Có tích hợp Gmail API để đọc email.
* Người dùng:
  + **User bình thường:** kết nối Gmail, nhập link, tải file, xem kết quả.
  + **Admin:** quản lý hệ thống, sandbox, mô hình AI.
* Chạy được trên Python + PostgreSQL.
* Giao diện có thể làm bằng Django template.

## 3. Các chức năng chính

1. **Kết nối Gmail (OAuth2)**
   * Người dùng cho phép app truy cập email (chỉ đọc).
2. **Quét email tự động**
   * App sẽ quét email mới và phân tích.
3. **Kiểm tra URL thủ công**
   * Nhập 1 link để app kiểm tra xem có phải phishing không.
   * Nếu cần, link đó sẽ được chạy thử trong sandbox.
4. **Tải file để kiểm tra**
   * Người dùng kéo thả hoặc upload file.
   * App phân tích nhanh (hash, chữ ký virus, metadata).
   * Nếu nghi ngờ thì chạy trong sandbox.
5. **Phân tích static + dynamic**
   * Static: đọc thông tin cơ bản từ URL, file.
   * Dynamic: chạy trong sandbox để xem hành vi.
6. **AI phát hiện và giải thích**
   * App sẽ dùng ML/Deep Learning để phân loại phishing.
   * Có phần giải thích (ExplainAI) cho dễ hiểu.
7. **Dashboard**
   * User: xem kết quả quét email/link/file.
   * Admin: xem tình trạng sandbox, mô hình AI, logs.

## 4. Giao diện và API

* Có ô nhập URL, nút submit.
* Có chỗ kéo thả file.
* Có màn hình chờ khi sandbox đang chạy.
* Có kết quả hiển thị: điểm nguy hiểm, lý do, và gợi ý nên làm gì.
* API REST: /api/scan/url, /api/scan/file, /api/scan/email, /api/results/{id}.

## 5. Yêu cầu phi chức năng

* **Bảo mật:** tất cả dữ liệu được mã hóa, có thể xóa theo yêu cầu.
* **Hiệu năng:**
  + Check URL nhanh < 3 giây nếu có sẵn dữ liệu.
  + Chạy sandbox thì chậm hơn, có thể mất vài phút.
* **Khả năng mở rộng:** có thể thêm sandbox, thêm server khi nhiều người dùng.

## 6. Dữ liệu và mô hình

* Dùng dataset công khai (Enron, PhishingTank, v.v.) + dữ liệu người dùng feedback.
* Mô hình AI:
  + URL: XGBoost hoặc CNN/Transformer.
  + Nội dung email: mT5, XLM-R.
  + File: kết hợp static + sandbox.

## 7. Điều kiện chấp nhận

* Người dùng kết nối Gmail và quét email thành công.
* Nhập URL → có kết quả.
* Upload file → có kết quả.
* Mỗi kết quả đều có giải thích bằng tiếng Việt/Anh.
* Người dùng có thể yêu cầu xóa toàn bộ dữ liệu.

## 8. Rủi ro

* Đọc Gmail phải được người dùng cho phép, phải có chính sách bảo mật rõ ràng.
* Sandbox phải an toàn để không bị lây lan virus.
* License cho AI phải hợp pháp.