Piotr Copek
Hyunseok Cho
Mateusz Znaleźniak
Szymon Molicki
Zuzanna Micorek

13.11.2026

Software Engineering

System Vision

Index

System Vision

1. System Domain Description

1.1 Problem Domain

Electronic mail is one of the most used communication channels for both private and professional use. However, despite the introduction of several security measures, spam mail still remains one of the major issues faced by internet users. Spam mail may contain unwanted advertisements, phishing, malware, etc.

Currently, spam detection technologies use machine learning techniques that help these systems recognize and classify spam mail based on their ability to understand complex patterns between legitimate mail and spam mail. However, most spam detection technologies currently used are black boxes, meaning they do not reveal how they make their determinations. In addition, most spam detection technologies require users to be logged in, which may compromise user privacy.

In this project, the domain is automated email content analysis with a high emphasis placed on privacy, transparency, and simplicity. This system will be used for text-based spam mail detection without requiring users to be logged in, nor will there be any need for persistent user profiles. This system will be for an Internet-based web application, which may be accessed by any web browser.

1.2 Business and User Context

The system is meant for users who are interested in quickly checking if a given email message is spam or not. The users may include students, office going individuals, or people who are concerned about phishing and fraud. Users are anonymous and do not need to create any accounts.

From a business standpoint, the system can be considered as a standalone service or a prototype that can be incorporated into a larger email processing pipeline. The system is designed to be lightweight and easy to extend with additional classification models in the future.

1.3 System Scope

The Email Spam Detection System is a stateless web application that:

- Accepts email text as input
- Performs spam classification using machine learning models
- Optionally provides explanations for classification decisions
- Returns results immediately to the user
- Does not store user data or email content

Out of scope are:

- Email inbox integration
- User authentication and authorization
- Persistent storage of emails or user activity
- Administrative dashboards and analytics

2. Description of Main System Processes

2.1 Submitting an Email for Analysis

The first process starts when the user enters the content of the email into a text form available on the system's web interface. The user has the option to request an explanation of the classification result. When the form is submitted, the system validates the input and sends it to the core spam service.

2.2 Spam Classification Process

The service for detecting spam acts as a facade, which coordinates the classification process. The service loads a pre-trained language model in memory (if it has not been loaded before) and performs an inference on the text. The process determines the probability of an email being spam and returns a classification result as being either spam or not spam.

If explanation mode is enabled, it will also calculate the most influential keywords contributing to the classification as being spam.

2.3 Alternative Risk Analysis (Sliding Window)

As an alternative method, the system also implements a sliding window classifier. In this method, the lengthy content of an email is broken down into overlapping text windows. The text windows are then processed independently, and the results are combined to determine the final probability of spam. This method is very effective in identifying spam patterns spread over large texts.

2.4 Presenting Results to the User

After the analysis is complete, system generates a response containing:

- The spam or not spam verdict
- Highlighted keywords (if explanation was requested)

The results are rendered on a dedicated results page. Original email text is displayed only during active session and is not stored after the response is returned.

2.5 Error Handling and Logging

In case of errors (e.g. model loading failure, invalid input, internal exception), the system displays a user friendly error page. Technical details are recorded in log for debugging and maintenance. Logging is file based and does not include user content.

Project Dictionary

1. Subjects

Term	Description
SpamService	Central service coordinating spam classification requests and acting as a facade to underlying models
SpamChecker	Machine learning classifier responsible for predicting whether text is spam

Term	Description
GeneralRiskOverseer	Alternative classifier using a sliding window strategy for long texts
TrainingModule	Component responsible for training and evaluating machine learning models
Logger	Logging subsystem used to record application events and errors
Flask Application	Web application handling HTTP requests and responses

2. Conceptual Entities

Term	Description
SpamPrediction	Immutable data object representing the result of spam classification
GroPrediction	Result object produced by the sliding window classifier
GroWindowConfig	Configuration defining window size and stride for text segmentation
GroTrainConfig	Configuration parameters used during model training
Configuration	Centralized application settings such as model paths and logging options

3. Functions and Activities

Term	Description
Classify Text	Process of determining whether an email is spam
Explain Prediction	Activity of extracting keywords contributing to classification
Load Model	Initialization of a pre-trained machine learning model
Train Model	Process of training a spam detection model on a dataset
Log Event	Recording informational or error messages to log files

4. Persons

Term	Description
User	Anonymous person submitting email text for spam analysis
Administrator	Person responsible for maintaining the system (out of scope for implementation)

5. Technical Terms

Term	Description
Spam	Unwanted or malicious email content
Spam Probability	Numerical value representing likelihood that text is spam
Confidence	Measure of certainty in the classification result
Sliding Window	Technique of processing long text in overlapping segments
Stateless Service	System that does not store user data between requests

6. Functional assumptions

- User can paste text into an input field
- System classifies the text as spam or not spam, nothing in between
- System returns a message with the classification
- System generates feature-based explanation (keywords)
- Frontend communicates with backend via an API

7. Non-Functional assumptions

- **Performance** - Response time under 3 seconds for classification
- **Scalability** - Ability to handle many users simultaneously
- **Security** - Emails sent by users are not stored permanently; secure transmission (HTTPS)
- **Usability** - Clear, intuitive interface similar to chat

- **Maintainability** - Modular backend structure allowing easy updates of the spam model
- **Reliability** - System should maintain high uptime
- **Explainability** - Explanations should be understandable for non-technical users
- **Availability** - Hosted on a standard server environment accessible 24/7

Milestones

- Project setup and repository creation
- UI mockups
- Basic classifier prototype
- Frontend - Backend integration
- Explanation engine
- Testing
- Deployment
- Final presentation

Summary

The System Vision document provides an outline of its scope, purpose, and main processes for the Email Spam Detection System. The Project Dictionary provides a common vocabulary. These documents lay the foundation for further analysis, design, and implementation. They ensure a common understanding between the development team and customer.