Detailed Project Report

*Spam Detection System*

|  |  |
| --- | --- |
| Written by | Sarthak Bathla, Srishti Gupta |
| Last revised date | 28-2-24 |

**Contents**

1. Objective…….......................................................................................................................................3

2. Benefits……..........................................................................................................................................3

3. Architecture Description.....................................................................................................................5

3.1 Design……………………………………………………………………………………………………………………………………5

3.2. Data Description........................................................................................................................6

3.3. Working with CQL......................................................................................................................6

3.4. Data Preprocessing....................................................................................................................6

3.5. Model Building..........................................................................................................................6

3.6. Data from user...........................................................................................................................6

3.7. Data Validation .........................................................................................................................6

3.8. Model call .................................................................................................................................6

3.9 . Deployment .............................................................................................................................6

4. Q & A……………. ...................................................................................................................................7

**Objective:**

The objective of developing a low-level document for the spam detection app for emails is to provide a comprehensive guide that outlines the technical specifications, functionalities, and implementation details of the spam detection system. This document serves as a blueprint for developers, system administrators, and stakeholders involved in the design, development, deployment, and maintenance of the spam detection app. By clearly defining the objectives, requirements, algorithms, and architecture of the app, the document aims to ensure a smooth and efficient development process while achieving the desired outcomes of accurate spam detection, user satisfaction, and system performance.

**Benefits:**

1. Clarity and Understanding: By documenting the low-level details of the spam detection app, stakeholders gain a clear understanding of the system's functionalities, components, and workflows. This clarity helps developers and system administrators make informed decisions during the development, deployment, and maintenance phases.

2. Alignment with Requirements: The document ensures that the spam detection app aligns with the specified requirements and objectives. It serves as a reference point for validating whether the implemented functionalities meet the intended goals of accurate spam detection, minimal false positives, and efficient resource utilization.

3. Facilitates Collaboration: A well-documented low-level document fosters collaboration among team members, including developers, testers, and system administrators. It provides a common reference point for discussing technical details, troubleshooting issues, and coordinating efforts across different stages of the development lifecycle.

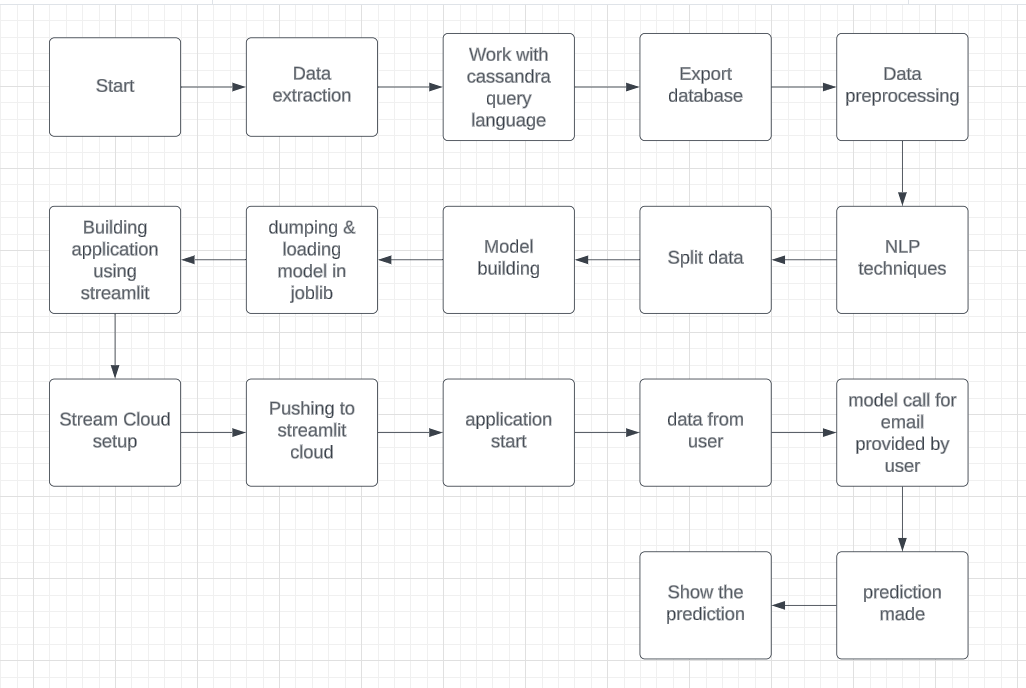
4. Enables Scalability and Extensibility: By detailing the architecture, algorithms, and data structures used in the spam detection app, the document lays the foundation for scalability and extensibility. It enables future enhancements, modifications, and integrations to accommodate evolving requirements and technological advancements without disrupting the core functionality of the system.

5. Enhances Maintenance and Support: The document serves as a valuable resource for maintaining and supporting the spam detection app over its lifecycle. It provides insights into the system's inner workings, making it easier to diagnose issues, perform updates, and address security vulnerabilities effectively.

6. Risk Mitigation: By documenting potential risks, limitations, and dependencies associated with the spam detection app, the document helps stakeholders anticipate challenges and implement appropriate mitigation strategies. This proactive approach reduces the likelihood of unexpected setbacks and ensures the long-term reliability and effectiveness of the system.

In summary, the low-level document for the spam detection app plays a crucial role in guiding the development process, fostering collaboration, ensuring alignment with requirements, and enabling scalability and maintainability. By providing a detailed roadmap for implementation, the document helps stakeholders realize the full potential of the spam detection system while delivering value to end users.

Architecture



**3. Architecture Description**

3.1 Data description

Spam detection dataset is the biggest publicly available spam mails dataset. Each mail has been categorized into spam and not spam. There are total 5573 rows which contains all possible type of spam mails.

3.2 Working with Cassandra Query Language

Cassandra Query Language," which is a query language for the Apache Cassandra database. Apache Cassandra is a distributed NoSQL database management system designed to handle large amounts of data across many commodity servers, providing high availability with no single point of failure. Here we uploaded and manipulated the data using CQL.

3.3 Data preprocessing

Data preprocessing techniques have been used. Data is being vectorized i.e converted into an array where data is being represented in binary format.

3.4 Model Building

We will choose the best model for training and prediction. Best parameters will be passed . Model with the highest accuracy will be chosen.

3.5 Data from user

We will collect mail body in text format which is supposed to be categorized

3.5 Data validation

Here, data validation will be done.

3.6 Model call

Model will be called and the data provided by user will be loaded. Categorization done by model is represented.

3.7 Deployment

The deployment is done on streamlit cloud.

Q & A

1. What are the primary objectives of the spam detection app low-level document?

- Answer: The primary objectives of the low-level document are to define technical specifications, outline system functionalities, describe implementation details, and guide stakeholders through the development, deployment, and maintenance processes of the spam detection app.

2. How does the low-level document benefit stakeholders involved in the development process?

- Answer: The document provides clarity and understanding of the system, aligns development efforts with requirements, facilitates collaboration among team members, enables scalability and extensibility, enhances maintenance and support activities, and helps mitigate potential risks associated with the spam detection app.

3. What key information is included in the low-level document for the spam detection app?

- Answer: The document typically includes details about the system architecture, algorithms used for spam detection, data structures, implementation workflows, integration points with other systems, performance metrics, security considerations, error handling mechanisms, testing strategies, and maintenance procedures.

4. How does the low-level document contribute to ensuring accurate spam detection?

- Answer: By detailing the algorithms, data processing techniques, and feature extraction methods used in the spam detection app, the document provides insights into how the system identifies and filters out spam emails while minimizing false positives and maintaining high accuracy levels.

5. What role does the low-level document play in facilitating collaboration among development team members?

- Answer: The document serves as a common reference point for discussing technical details, troubleshooting issues, coordinating development efforts, and sharing knowledge among developers, testers, system administrators, and other stakeholders involved in the project.

6. How does the low-level document address the scalability and extensibility of the spam detection app?

- Answer: By documenting the system architecture, design patterns, and integration points, the document lays the foundation for scaling the app to handle increasing volumes of email traffic and accommodating future enhancements, modifications, and integrations with external systems or technologies.

7. What are some potential risks and challenges addressed in the low-level document, and how are they mitigated?

- Answer: The document identifies potential risks such as data privacy concerns, algorithmic biases, performance bottlenecks, security vulnerabilities, and dependencies on third-party services. It outlines mitigation strategies, best practices, and contingency plans to address these risks and ensure the robustness and reliability of the spam detection app.

8. How often is the low-level document updated, and what triggers the updates?

- Answer: The document is typically updated throughout the development lifecycle in response to changes in requirements, technology advancements, system updates, security patches, performance optimizations, feedback from users, and lessons learned from previous deployments or incidents. Regular reviews and audits also prompt updates to ensure the document remains accurate and relevant.