



SPAM MESSAGE DETECTOR APP

AN APPLICATION TO IDENTIFY SPAM MESSAGES
USING PRE-TRAINED ML MODELS



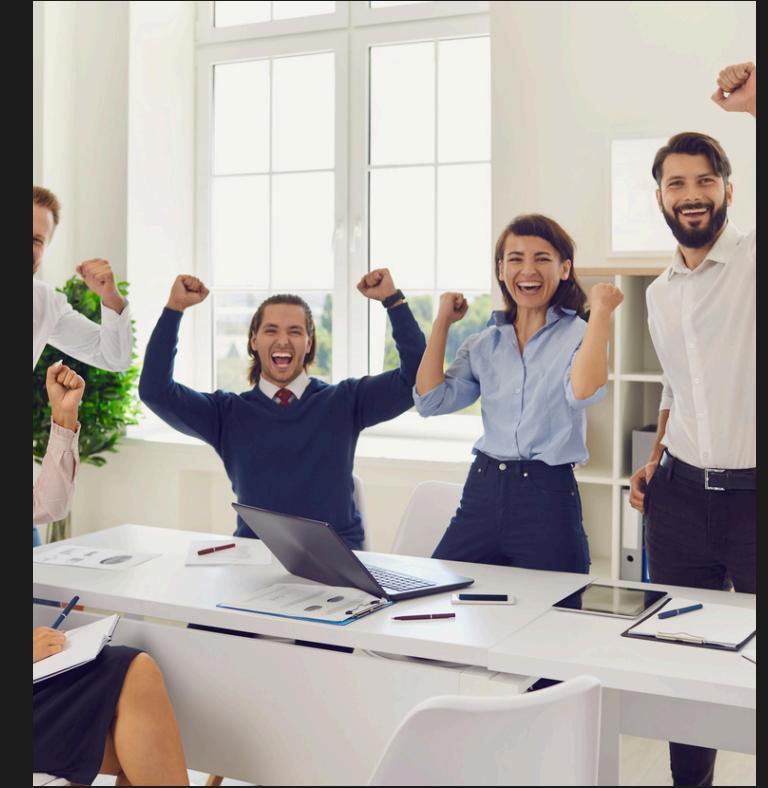
PRESENTED BY **SHIVAM PRATAPWAR**

FEATURES OF THE APP



CORE FEATURES:

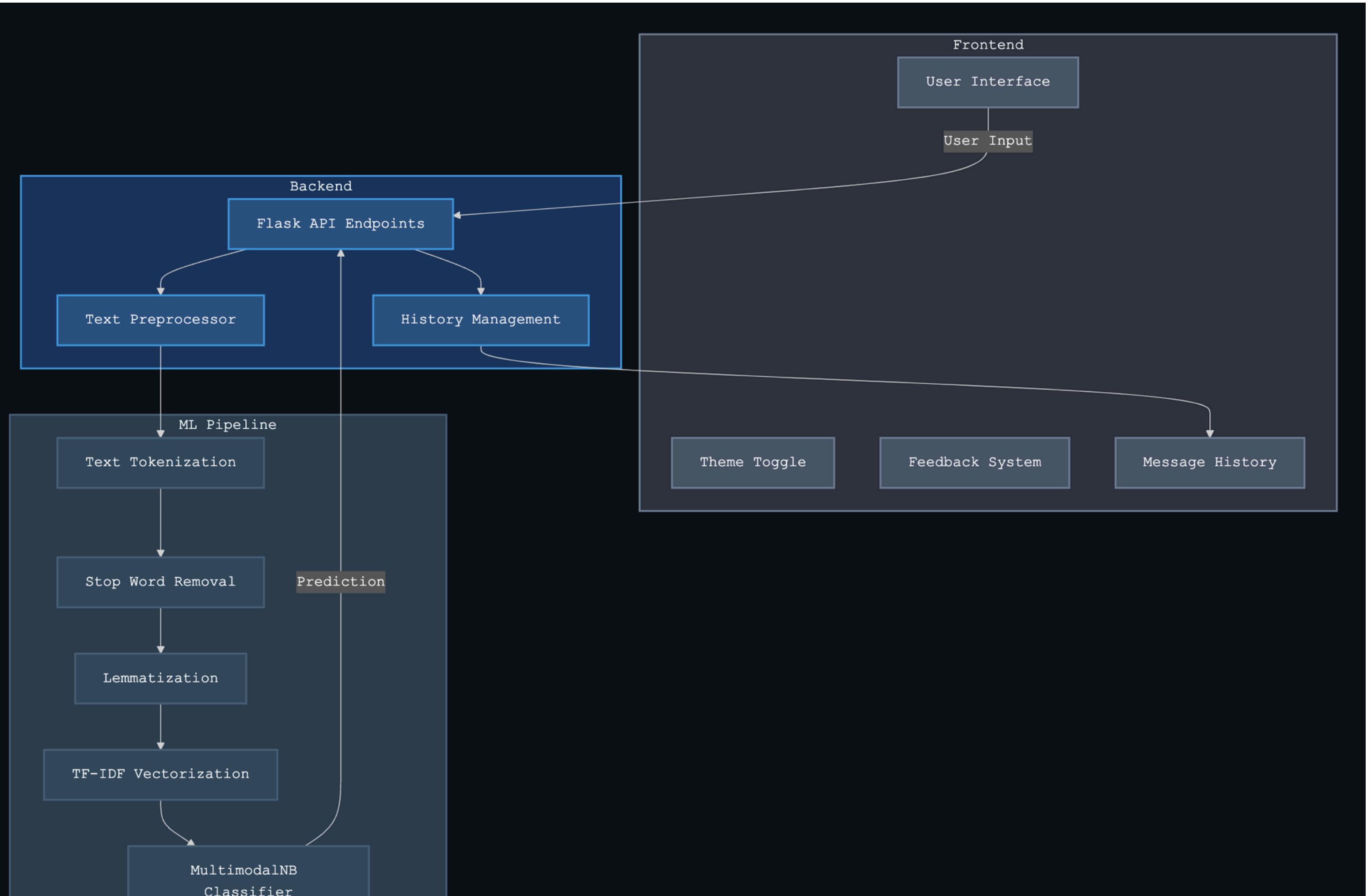
- REAL-TIME TEXT CLASSIFICATION
- HIGH-ACCURACY SPAM DETECTION
- MESSAGE HISTORY TRACKING
- CONFIDENCE SCORE DISPLAY
- TEXT PREPROCESSING VISUALIZATION
- DARK/LIGHT THEME TOGGLE
- RESPONSIVE WEB INTERFACE



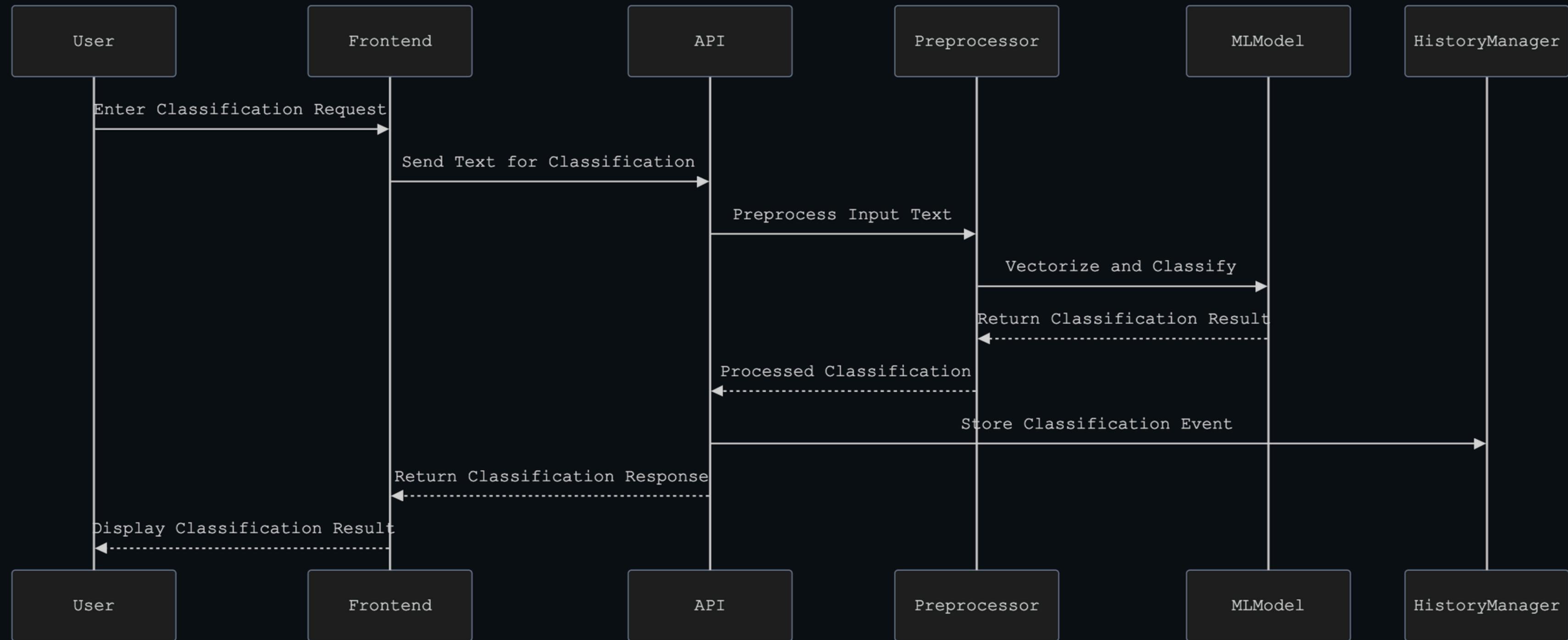
ADVANCED FEATURES:

- MESSAGE HISTORY MANAGEMENT (LAST 50 MESSAGES)
- DETAILED CLASSIFICATION METRICS
- PROCESSED TEXT VISUALIZATION
- INTERACTIVE USER FEEDBACK SYSTEM





* WORKING OF THE APP



* WORKING OF THE APP



Tools and Frameworks

Frontend

- HTML5/CSS3
- JavaScript
- Bootstrap 5
- Interactive UI components

Backend

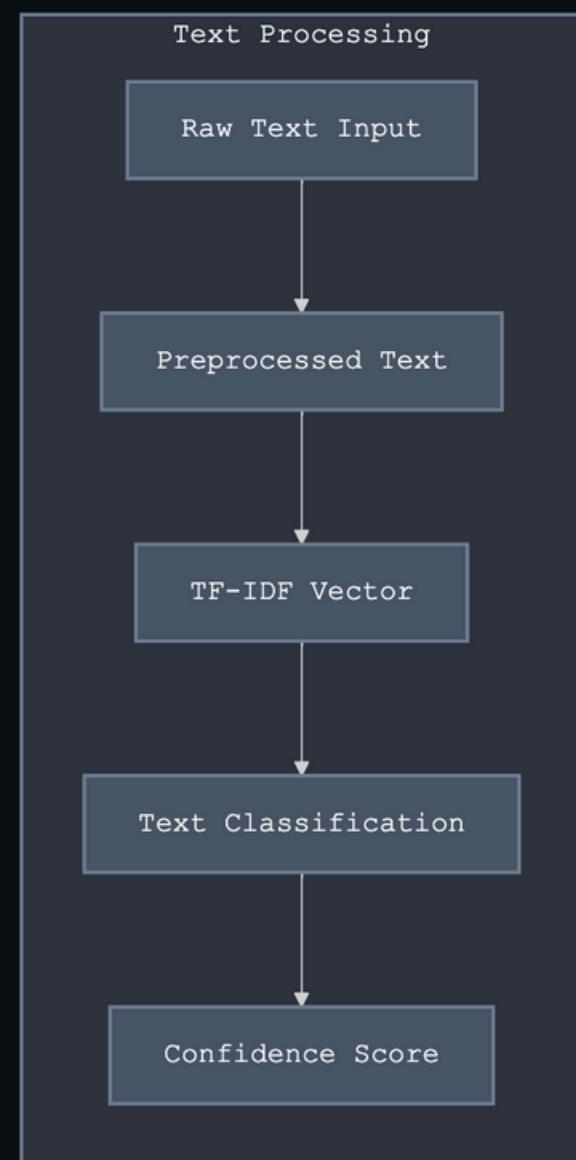
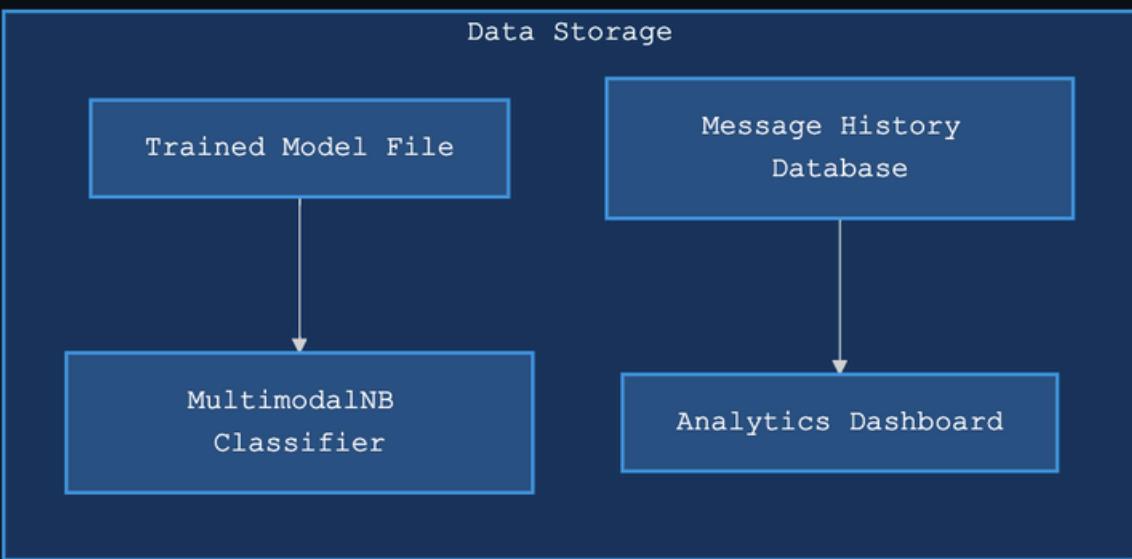
- Python 3.x
- Flask framework
- RESTful API architecture

Machine Learning

- NLTK for NLP
- scikit-learn
- TF-IDF Vectorization
- MultinomialNB Classifier

Data Management

- In-memory storage (current)
- Pickle serialization
- JSON data exchange



* INTENDED USE CASES



1 PROFESSIONAL DOMAINS

Telecom Companies

- Automated spam filtering
- Network quality improvement
- Customer protection

2 PERSONAL SAFETY

Individual Users

- Block unwanted messages
- Protect from potential scams
- Reduce communication noise

3 ORGANIZATIONAL APPLICATIONS

Customer Support

- Filter support channel messages
- Prioritize critical communications
- Reduce manual screening efforts

4 CYBERSECURITY

Fraud Prevention

- Early detection of phishing attempts
- Protect sensitive information
- Provide real-time threat assessment

* GAMIFICATION POSSIBILITIES

ENGAGEMENT STRATEGIES

Classification Mastery

- LevelsEarn "Spam Hunter" badges
- Progress through detection accuracy
- Unlock special app features



USER MOTIVATION MECHANISMS

Classification Mastery

- LevelsEarn "Spam Hunter" badges
- Progress through detection accuracy
- Unlock special app features

Community Contribution

- Leaderboards for most accurate users
- Collaborative model improvement
- Reputation points for feedback



Detection Streak

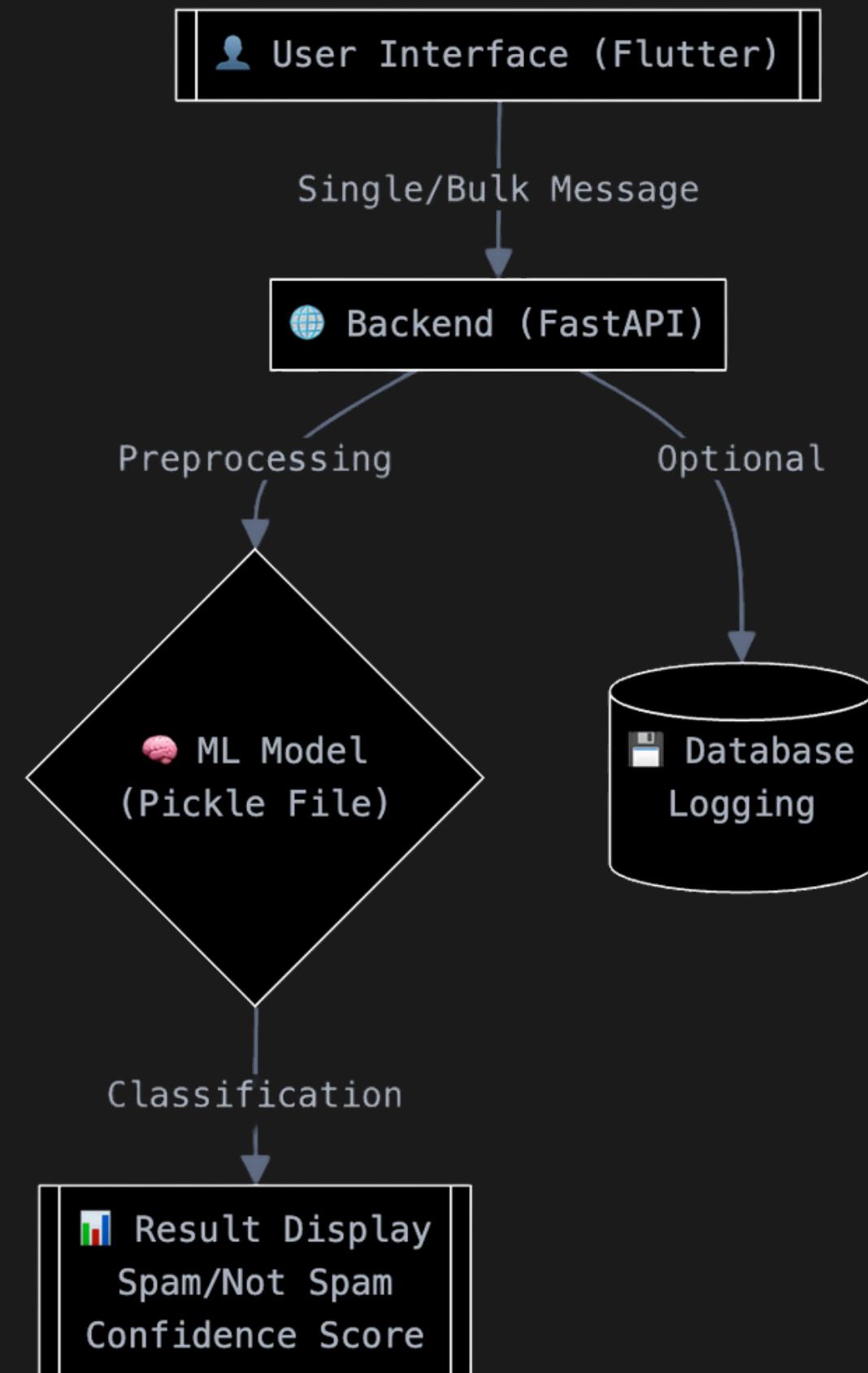
SystemDaily spam detection challenges

- Consecutive correct classification rewards
- Personalized achievement tracking

* HIGH-LEVEL DESIGN

Tools and Frameworks

- Frontend: Flutter (cross-platform for mobile and web UI).
- Backend: FastAPI (lightweight, fast web framework for APIs).
- Model: Pre-trained spam classification model, saved as .pkl.



CONCEPT AND OBJECTIVE



PROBLEM STATEMENT:

- SPAM MESSAGES CLUTTER INBOXES, POSING RISKS LIKE PHISHING AND FRAUD.



OBJECTIVE:

- BUILD A LIGHTWEIGHT APP THAT ACCURATELY DETECTS SPAM MESSAGES USING A PRE-TRAINED ML MODEL.



FEATURES:

- Detect if a single message is spam.
- Bulk processing for multiple messages from a text file.
- Spam probability score for each input.



* EXECUTION PLAN



WEEK 1: DATA AND MODEL DEVELOPMENT

- Download SMS Spam Collection dataset and explore class distribution.
- Preprocess text (remove stopwords, tokenize, vectorize). Train a simple ML model (Naive Bayes or Logistic Regression).
- Fine-tune the model, evaluate performance, and save it as a .pkl file.

WEEK 2: API AND BACKEND DEVELOPMENT

- Set up FastAPI backend with endpoints for single text input and bulk file upload.
- Integrate the .pkl model into the API and implement preprocessing pipelines.
- Test endpoints and debug for edge cases.

WEEK 3: TESTING, DEPLOYMENT, AND PRESENTATION

- Test end-to-end functionality with varied input types (text, files).
- Deploy the API on a cloud or local server and test API response times.
- Prepare the final presentation and demo, rehearse for viva.

*



*

THANK YOU FOR
YOUR
ATTENTION