DEPARTAMENTO DE COMPUTACIÓN, ELECTRÓNICA Y MECATRÓNICA

Desarrollo de aplicaciones móviles (LIS4012-1)

Profesor: Julio Noé Hernández Torres

LAB 04 - 06.02.2025 - Requirements

Cecilia Soriano Tochimani 173633

Melisa Sampieri Espinoza 173948

Luis Felipe Jarquin Romero 186812

Primavera 2025

13 de enero de 2025

Title & Team Members:

App title: VibeCheck

- Cecilia Soriano Tochimani, 173633
- Melisa Sampieri Espinoza, 173948
- Luis Felipe Jarquin Romero, 186812

Problem Statement:

Social media often becomes a platform for impulsive or emotionally charged communication, leading to misunderstandings, negativity, or unintended consequences. There is a need for a tool that helps users reflect on their tone and language before posting.

Target Audience:

People of all ages using social media, with a particular focus on teenagers and young adults who may post impulsively, on public figures who are under the spotlight and who care about their reputation, and non-native speakers who struggle with tone in a different language.

Key Features:

- Emotion recognition in text
- Suggest edits & improvements
- Multilingual.

Technical Approach:

- Development platform: iOS
- Key technologies: SwiftUI, generative language model

User Interface Sketches

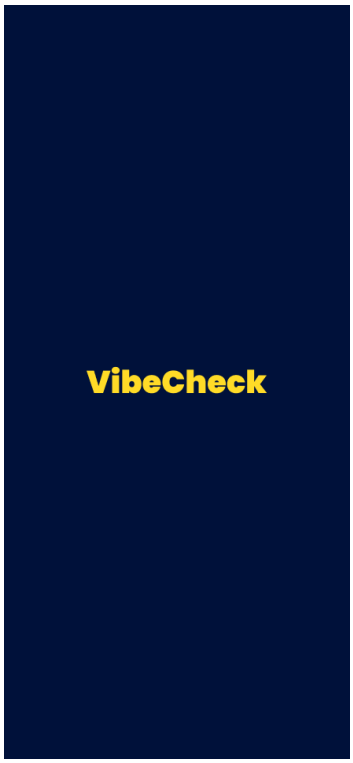

Figure 1. Splash screen

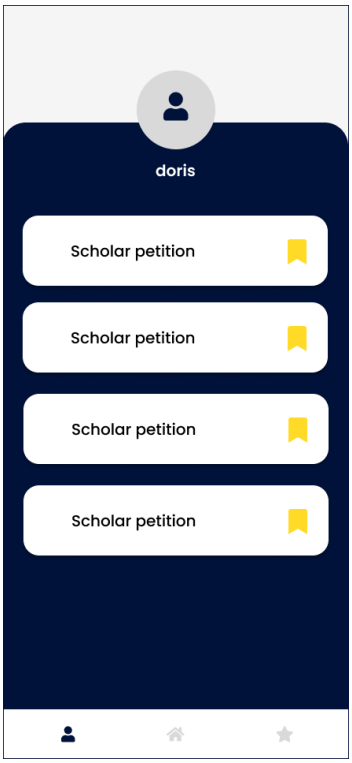

Figure 2. Profile

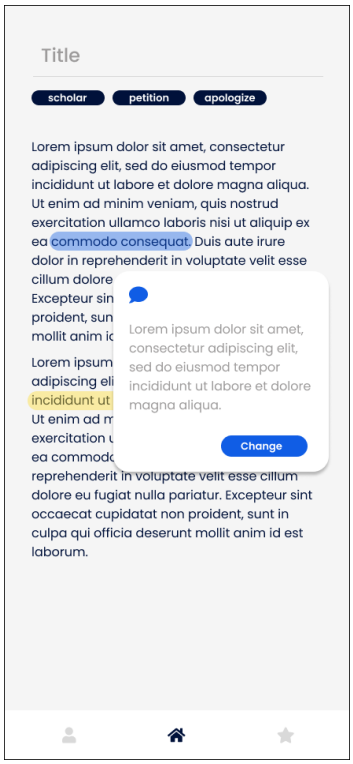

Figure 3. Recommendations

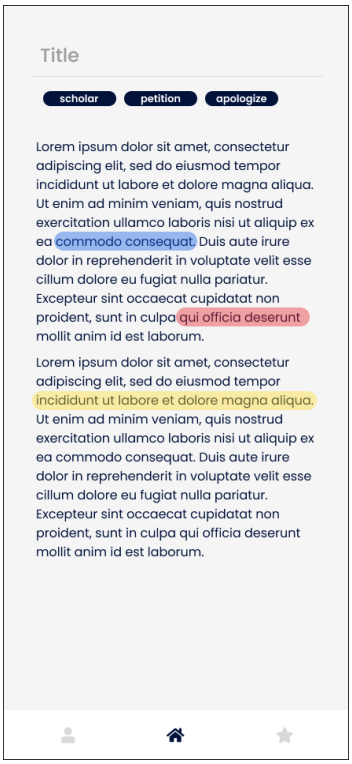

Figure 4. Text

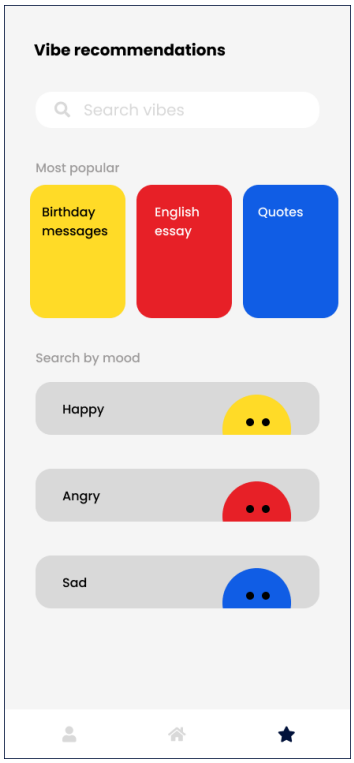

Figure 5. Vibe Recommendations

Challenges & Risks:

- Subjectivity in Tone Analysis:
- Complexity of emotional language.
- Cultural and linguistic diversity.
- Training the appropriate models

Expected Outcome & Impact:

- Reduction in misunderstandings and negativity.
- Improved digital Well-Being.
- Fostering empathy and respect.
- Encouragement of constructive content creation.

Deliverable	Description	Due Date

Requirements Specification	Define functional and non-functional requirements	Tuesday, February 13th, 2025
UI/UX Design	Create wireframes, prototypes	Friday, February 13th, 2025
Frontend Development	Mobile app UI and features implementation	Friday, February 20st, 2025
Backend Development	API development database setup and training models	Friday, March 20st, 2025
Testing & QA	Unit, integration, and user testing	Friday, March 28th , 2025
Deployment	Submission to Google Play/ App Store	Date to be set

Breakdown structure

Requirements specifications

- Define the project by identifying and analyzing user needs, the technical constraints and having a Risk Mitigation Plan to avoid wasting time.
- The key functionalities will be defined to have a plan of work in wich all the functions of the app will be developed on time.
    - Text emotion recognition using AI models.
    - Automated suggestions for improving tone and clarity.
    - Multilingual support for better communication across different languages.
    - Secure user authentication and profile management.
    - A feedback system that allows users to compare original and suggested edits.
- A document with the requirements will be developed to ensure that everything that the app needs is complete.
- Define functional and non-functional requirements.
- Outline the expected user interactions and system behavior.
- The app integrates emotion recognition in text, suggests edits and improvements, and supports multiple languages.

UI/UX Desing

- Develop wireframes and interactive prototypes using design tools.
- Establish a design system with consistent typography, colors, and UI components.
- The Use Cases and System Workflows will be defined in this step, and example may be:
    - Use Case 1: Emotion Detection and Suggestions

- ♣ The user enters or pastes a social media post.
- ♣ The AI model analyzes the tone and suggests modifications if needed.
- ♣ The user selects one of the suggested edits or ignores the recommendation.
- ♣ If accepted, the modified text replaces the original post.

Frontend Development

- Develop the user interfaces using SwiftUI.
    - Created using Figma to design interactive user journeys.
- Implement layout, animations, and navigation flow.
- Ensure UI responsiveness and adaptability across different iOS devices.
    - Dynamic UI layouts that adapt to different screen sizes.
    - Real-time text analysis and inline suggestions.
    - Smooth transitions and animations for improved user experience.
- Exclude backend logic or functional requirements at this stage.

Backend Development

- Develop the functional backend logic, including authentication, data processing, and user management.
    - User authentication and secure session management.
    - AI-powered sentiment analysis and text refinement.
    - API development to connect frontend with backend processing.
- Implement API endpoints for communication between frontend and backend.
    - A connection will be used for ensuring that the user have an option to choose which comment is better for their situation
- Set up and configure the database for data storage and retrieval.
    - The database will be used for the password and saving information from the user like their corrections for their comments.
- Train or deploy machine learning models.
    - Preprocess and clean datasets for training.
    - Train, validate, and fine-tune machine learning models.
    - Deploy models to the backend and optimize inference speed.
    - If the model is a pretrained model the steps mentioned above may change.

Testing & QA

- Unit Testing: Validate individual components and modules.
- Integration Testing: Ensure seamless interaction between frontend, backend, and APIs.
- User Testing: Conduct usability testing with real users to gather feedback.
- Performance Testing: Assess app performance under different conditions.

Deployment

- If needed the deployment will consist in:
    - Prepare the application for release by optimizing performance and fixing bugs.
    - Conduct final security checks and compliance verifications.
    - Package the app and submit it to the Apple App Store (iOS)
    - Monitor app performance post-launch and address issues through updates.

Risk Mitigation Plan

Risk	**Impact**	**Mitigation Strategy**
App Store Rejection	High	Conduct a thorough review of Apple's App Store guidelines, including safety, performance, business policies, design requirements, and legal compliance before submission. Perform pre-submission testing using Apple's TestFlight to identify potential rejection reasons early. Consult Apple's review guidelines regularly and ensure all necessary documentation, such as privacy policies and terms of service, is included in the submission.
Data Security Issues	High	A specific strategy for ensuring user data security has not yet been determined. An in-depth investigation will be conducted to identify the best security practices, considering approaches such as data encryption, authentication protocols, vulnerability prevention, and regulatory compliance. The goal is to select the most suitable strategy to protect user information and ensure adherence to privacy regulations.
Delays in Development	Medium	Establish a structured Agile workflow with sprints and defined milestones. Hold bi-weekly sprint planning meetings every Tuesday and Thursday to track progress, address blockers, and reallocate resources as needed. Implement automated CI/CD pipelines to streamline development and testing processes, reducing manual workload and improving efficiency.
Compatibility Issues	Medium	Conduct rigorous cross-device and cross-platform testing using real devices and emulators, Ensure backward compatibility by following Apple's Human Interface Guidelines and regularly testing app updates before rollout.

Role Assignment

Project Manager: Melisa Sampieri Espinoza

Development Lead: Cecilia Soriano Tochimani

QA Manager: Luis Felipe Jarquin Romero

Functional Requirements

- The user must be able to register using email and password.
- The app must analyze and correct text in real time.
- The system shall be able to rewrite texts without changing the meaning through the use of an API.
- The system shall use a database to save data from the user.
- The app must be able to identify by color the emotion detected.
- The app will rate the comments by stars (1 – 5).
- The user must be able to save texts in their profile.
- The app must be able to recommend messages or texts by labels.

Non-functional Requirements

- The application must efficiently handle large amounts of text without compromising speed or accuracy.
- User data must be securely processed and protected through encryption and compliance with privacy regulations.
- The interface must be intuitive, user-friendly, and accessible to a wide range of users.
- The system must process text and provide feedback within 6 seconds.
- High accuracy in emotion detection with a confidence level above 85%.
- Compliance with data privacy regulations.
- Cross-device compatibility, ensuring a seamless experience on different iOS devices.