VIBE: DEVELOPMENT OF A SOCIAL PLATFORM FOR MICROBLOGGING

DAVID SANTIAGO TELLEZ MELO-JOHAN NICOLAS LOPEZ HERNANDEZ

UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

INTRODUCTION

Vibe is a social platform that facilitates the posting and viewing of content. Its design allows users to share short messages, images, and links, and interact with other posts through reactions.
The project focuses on providing a simple, functional, and straightforward experience.
To achieve this, an object-oriented programming structure was used, which allowed for a clear definition of the system's main components: users, posts, notifications, and content display.
Each element of Vibe has been modeled as an independent class, with its own attributes and methods, allowing the internal logic to be organized, modular, and easily extensible.
Additionally, tools such as UML diagrams, CRC cards, and visual mockups were used to define the relationships between components and plan user interaction with the interface.
Vibe was conceived as a lightweight and functional platform, centered around the essential elements of a social network: user registration, post creation, dynamic feed display, and simple reactions.

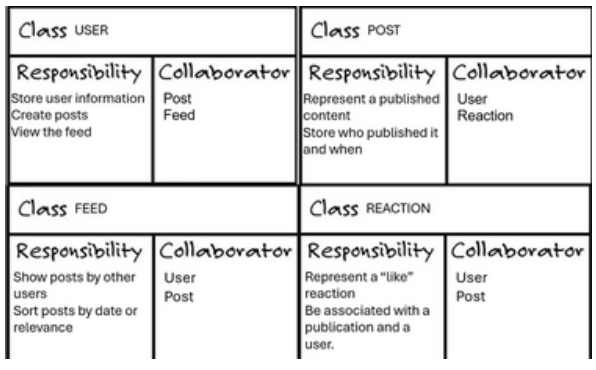
OBJECTIVES

Among the objectives we have for this application are:

-to provide a simple to understand interface
-basic user information should be able to be entered: name, age, profile picture, etc.
-Possibility to publish texts of up to 280 characters, links and images.
-Have a button with which users can react to each other's content.
-Have a main feed where you can see the content of other users.

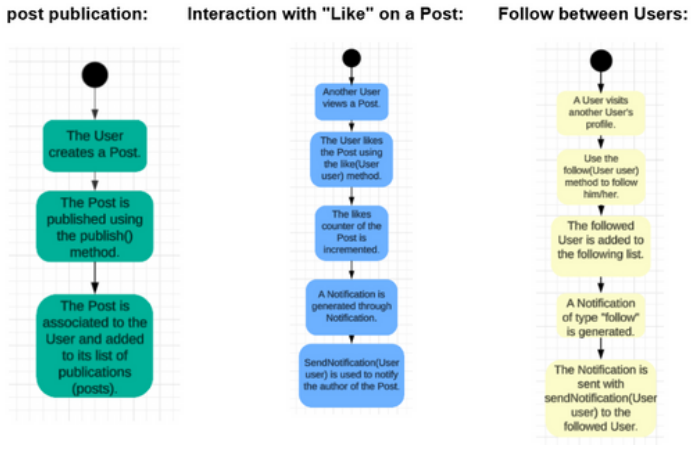
METHODS

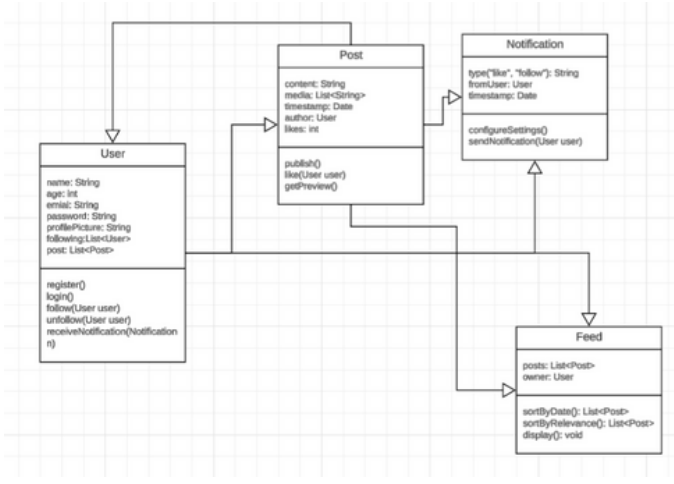
Vibe was developed in Java using object-oriented programming to organize the system into well-defined classes such as User, Post, Reaction, Notification, and Feed. Each class has specific attributes and methods that allow for managing actions such as registering users, creating posts, generating notifications, and updating the feed.
UML diagrams were used to plan the system's structure, CRC cards to distribute responsibilities among classes, and use flows to represent key actions within the application. Additionally, interface mockups were designed to guide the visualization of the system before implementation.



They were used to define what each class does and who it interacts with. For example, the User class manages user data and collaborates with Post and Notification. They helped organize the logic before programming.

They represent the main classes of Vibe (User, Post, Feed, etc.), their attributes and methods. They allowed for planning the system's structure and visualizing how the components are connected.





They show key actions such as posting, reacting, or following. They were used to understand which classes are involved in each process and how information flows within the application.

REFERENCES

[1] Gamma, E., et al. Patrones de Diseño: Elementos de Software Orientado a Objetos Reutilizable. Addison-Wesley, 1994.
[2] Smith, J., Doe, A. Construcción de Servicios Escalables de Microblogueo. Journal of Web Systems, 2020.