VibeCheck

<https://github.com/comp195/Spring2020Project-vibecheck>

Devin French (d\_french3@u.pacific.edu)

Seth Kith (s\_kith@u.pacific.edu )

Table of Contents

[**System Description and Project Overview:**](#_45sm23cvcam7)3

[**Non-Functional Requirements**](#_7m00omu4l3ye)4

[**Functional Requirements**](#_sp2yh0nr4qqz)5

[**User Interface Design Sketches:**](#_rmdzd71lvgmy)8

[**Glossary of Terms**](#_6wq0jqo7l74n)11

[**References**](#_9eq86i1wwnaq)10

## System Description and Project Overview:

VibeCheck is a new web social media platform. With the rise of different social media platforms with their own unique aspect to it, VibeCheck will have its own unique aspect by providing the users to customize their own profile. With this idea in mind, we want to know if users prefer having a customizable profile or a simple profile for other users to view. Most users follow a saying, “the simpler, the better” or “simplicity is the best” and with the current social media platforms, they seem to follow it as well based on their design. As we stray away from having unique profiles and assuming the saying is true, we expect to see users orientate towards a simple style profile.

## 

## 

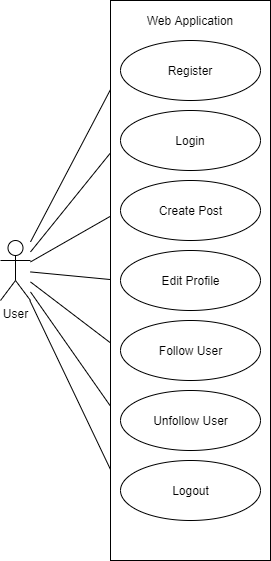
## Non-Functional Requirements:

* Availability
* Scalability
* Reliability
* Maintainability
* Usability

## 

## Functional Requirements:

Use Case Diagram



**Use Cases**

Register

* Pre-condition: User is logged out
* User inputs data into the sign up form, filling out the username, email, and password fields. The user submits the form by clicking a button. A request to create a new user account is submitted and a response is received.
* Post-condition:   
  Success: Account is created and user is logged into the created account and an authentication token is stored in local storage.  
  Fail: An alert appears with an error message and no account is created.

Login

* Pre-condition: User is logged out
* User inputs data into the login form, filling out the email, and password fields. The user submits the form by clicking a button. A request to authenticate the credentials is submitted and a response is received.
* Post-condition:   
  Success: Credentials matched a valid user, the user is now logged in and an authentication token is stored in local storage.  
  Fail: An alert appears with an error message and the user is not logged in.

Create Post

* Pre-condition: User is logged in
* User either inputs text into the textarea, inputs an image from the user’s computer, or both. User clicks the form submission button. A request to create a new post is submitted and a response is received.
* Post-condition: New post is added to the posts list

Edit Profile

* Pre-condition: User is logged in and is viewing their profile
* User clicks the Edit Profile link. A form with editable profile inputs appears. The user inputs changes to their profile on the form. The user submits the form and a request containing the changes is submitted and a response is received.
* Post-condition: Update user’s profile with the new changes

Follow User

* Pre-condition: User is logged in and is viewing another user’s profile that they do not follow
* User clicks on the Follow button. A request is submitted and a response is received.
* Post-condition: User added to logged in user’s following

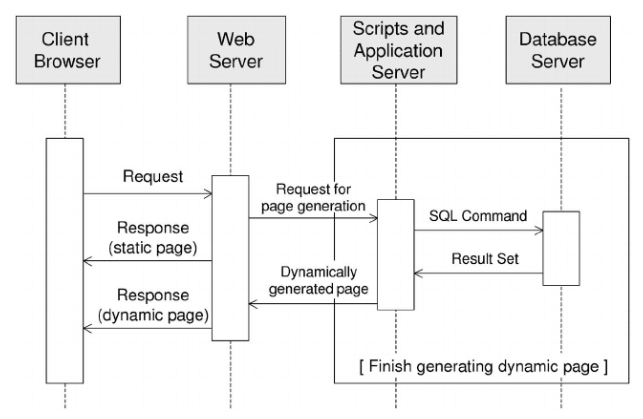
Unfollow User

* Pre-condition: User is logged in and is viewing another user’s profile that they follow
* User clicks on the Unfollow button. A request is submitted and a response is received.
* Post-condition: User removed from logged in user’s following

Logout

* Pre-condition: User is logged in
* User clicks on the logout button. User authentication token1 is removed from local storage.
* Post-condition: User is logged out. The user is redirected to the login screen if viewing a restricted page.

**System Sequence Diagram**

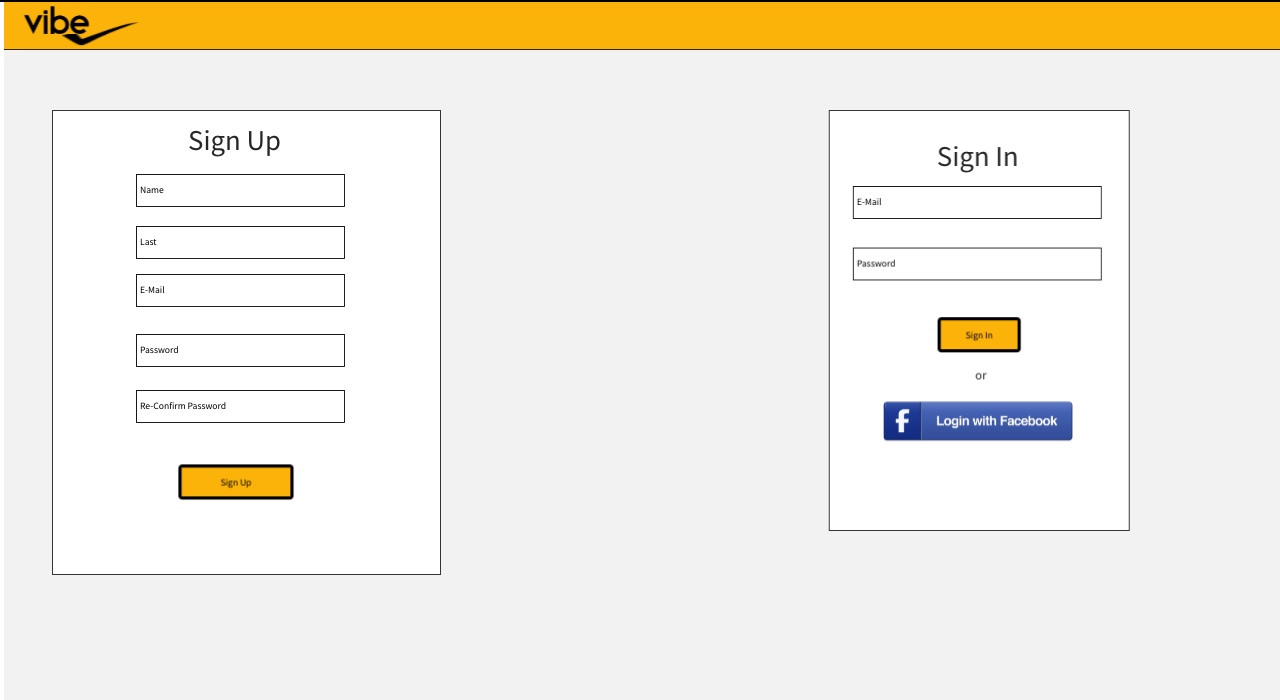
****

## 

## User Interface Design Sketches:

When accessing the website, the user is greeted with a “login screen” (Fig. 1). The users are given 2 options to gain access to the website. The first choice would be to “Sign up” if the user doesn’t have an account. The second choice would be for those who have an existing account. After signing up or logging in, it will transfer the user to the “home screen” (Fig. 3) (NOTE: Fig. 1 does show a Facebook login option, but it’s not implemented.)

Fig. 1 (Login)



If the user forgot their password, in the login screen there is an option to retrieve their password. Selecting that option will transfer the user to the “forgot password” (Fig. 2) page.

Fig. 2 (Forgot password)

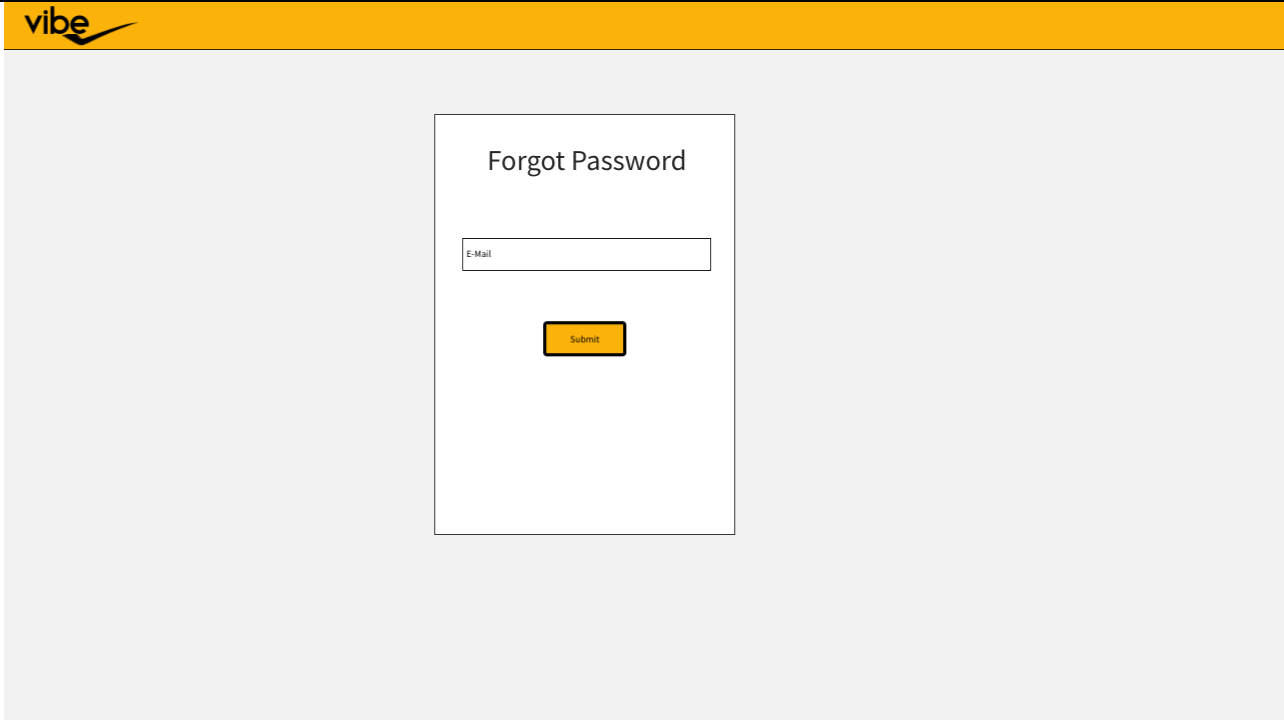
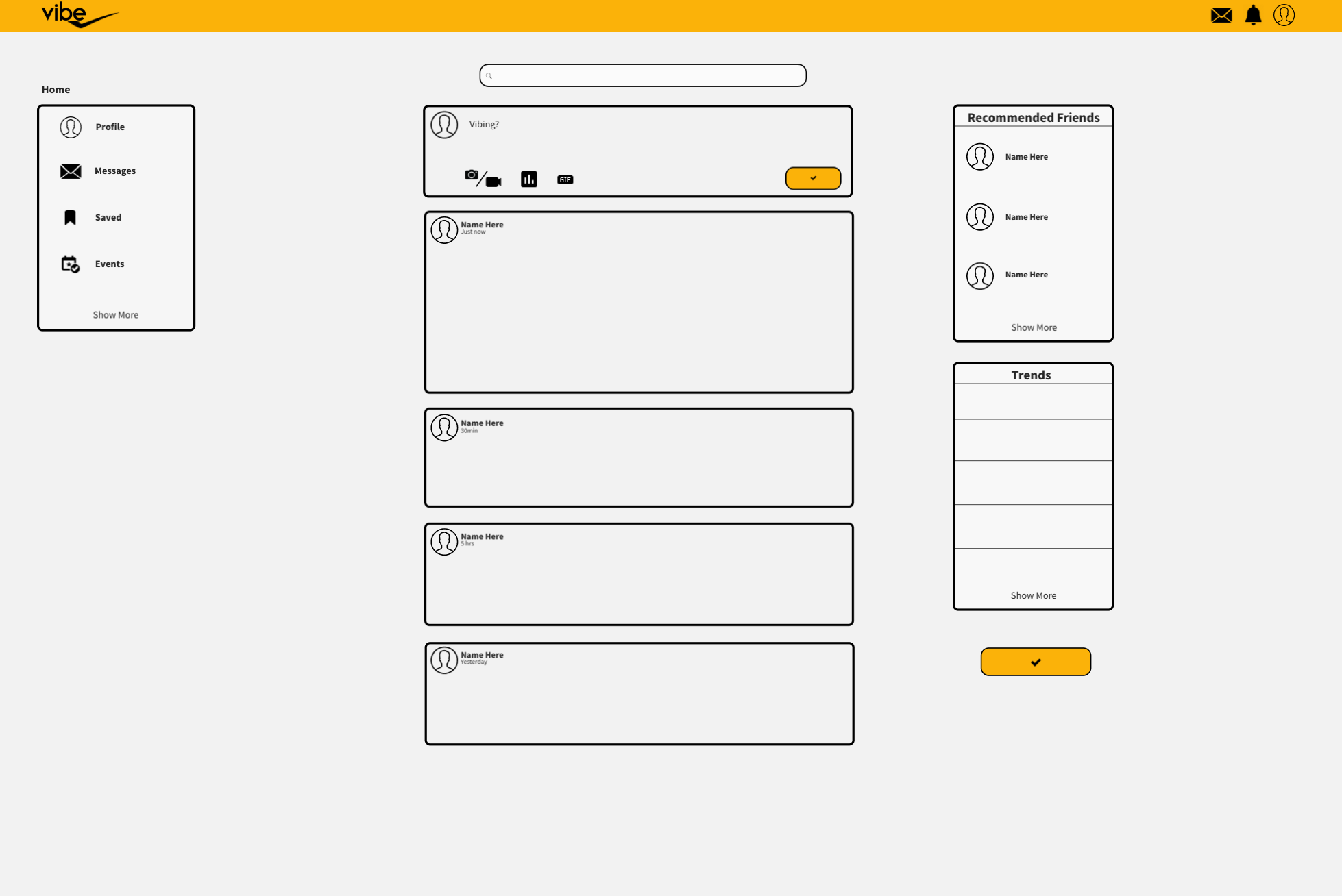


Fig. 2 (Home screen)



Every user will have a basic profile (Fig. 3). The user has an option to customize/edit their profile by interacting with the cog on the top right as shown in the picture and edit their profile. It will bring the user to an “edit profile” (Fig. 4) page.

Fig. 3

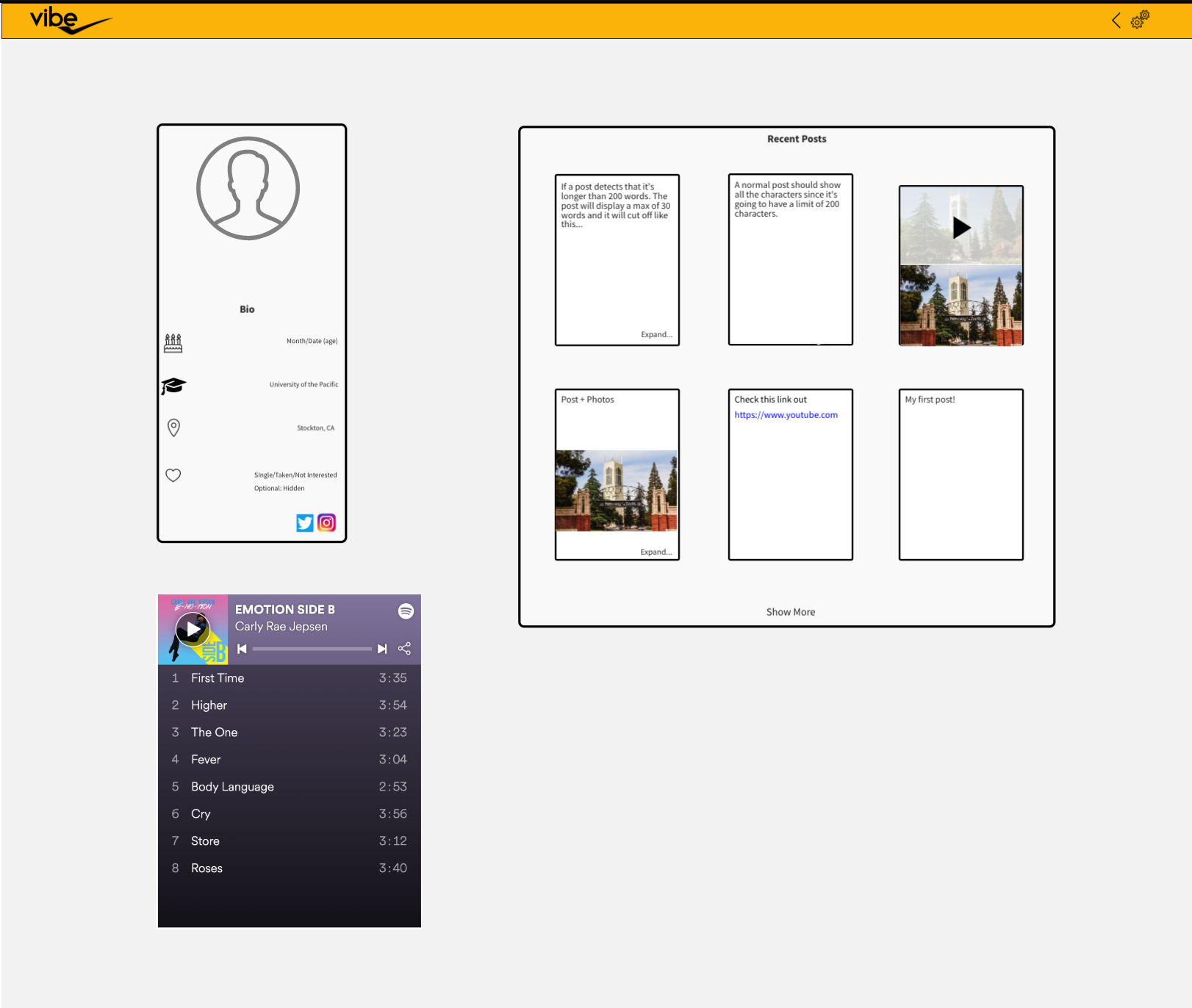
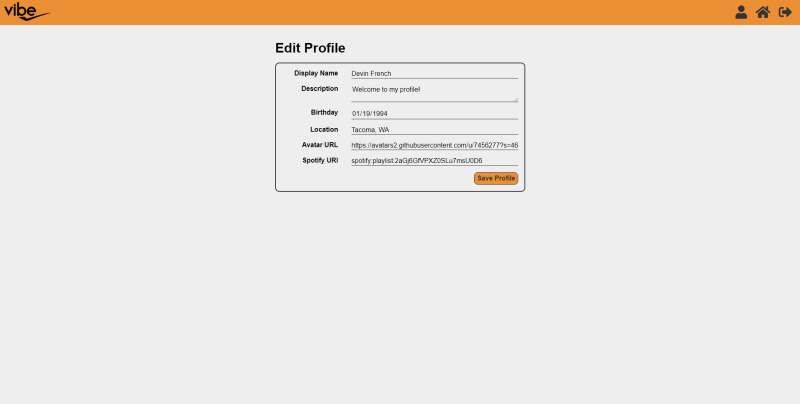


Fig 4.



## Glossary of Terms:

Token1 - Keyword, operator, or punctuation

## References:

N/A