ESAW Final Lab Report

InspoWorks: Twitter-like application for artists

Ferran Martinez 205821
Paula Sarrà 216886
Daniel Rosa 216835

TABLE OF CONTENTS

Introduction

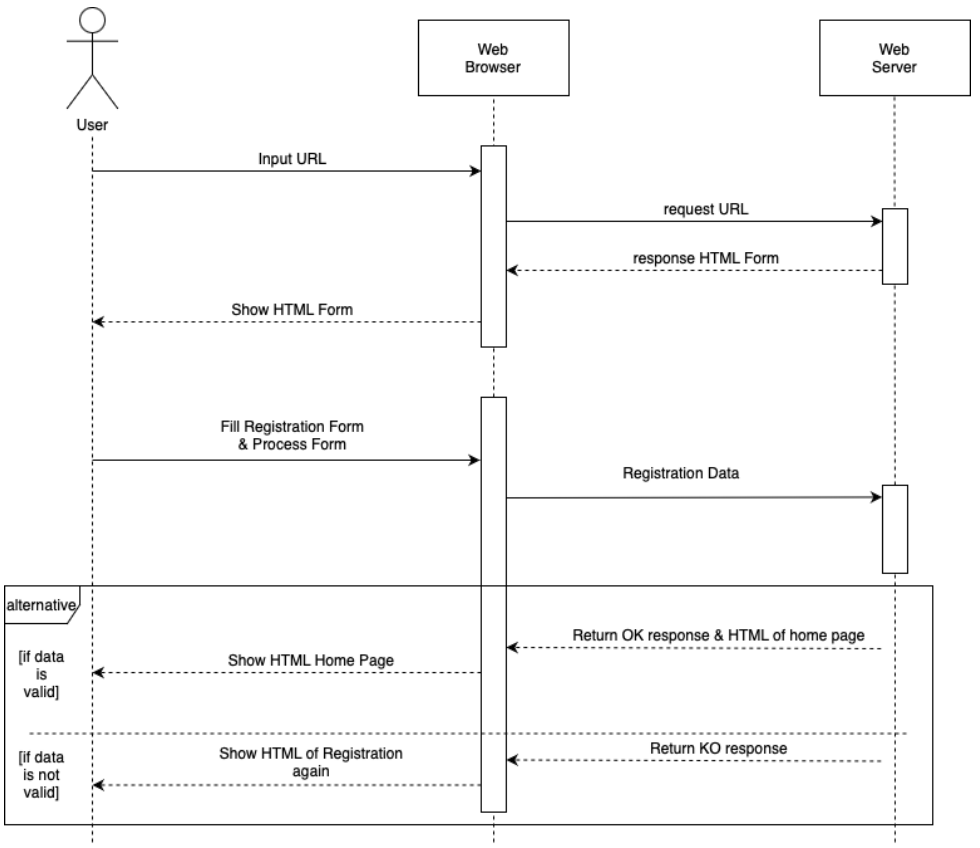
Our aim is to create a Twitter-like application focused on art named “InspoWorks”, in which artists can share their ideas and creations via “inspos” (tweets).
For this, our first approach was to build a website that offers their users the possibility to edit their own feed with the aim to make it their portfolio, as well as focusing the artwork to images.
But after some development of this first idea we thought it was not that appealing, since it would be tricky for the users to customize their profile in a deep level, and we wanted to create an intuitive and minimalistic website. So, we decided to remove that idea of being able to edit almost anything and also we thought it would be more interesting to share text instead of pictures, because there are so many sites such as Pinterest or Instagram that already cover that use case.
Since the site is still art-focused, we built a website in which users can share their creative ideas, reflections and inspirations with the community.

In this report, we will be covering the most important aspects of the development of this project, such as the technologies/libraries being used, the database organization and the technical details of key functionalities.

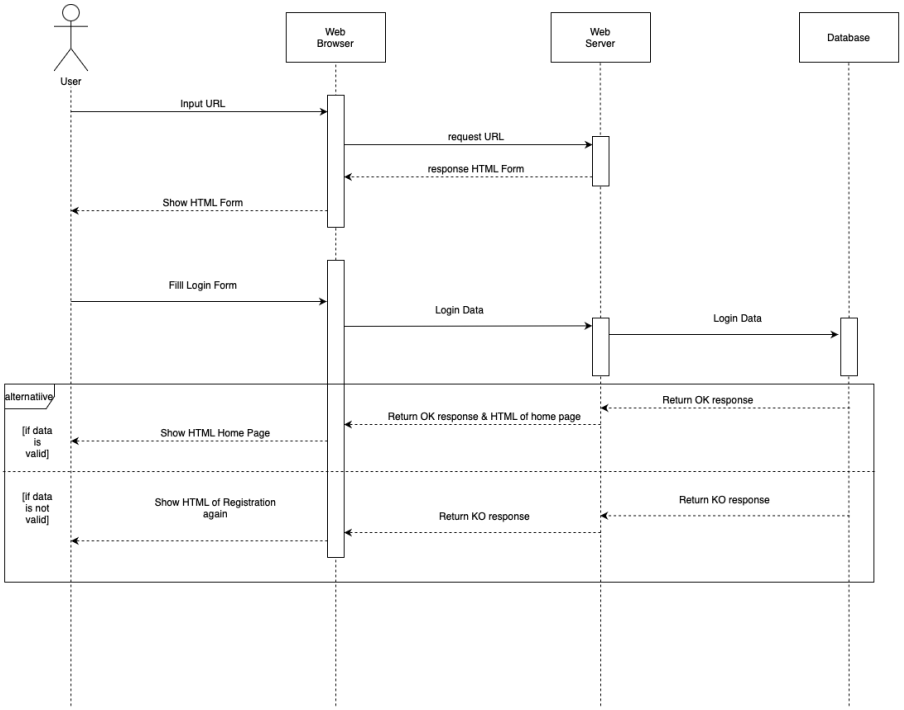
Register & Login

The registration form of our applications consists of the following sections:
name, username, email, birthday date, phone, gender and password.

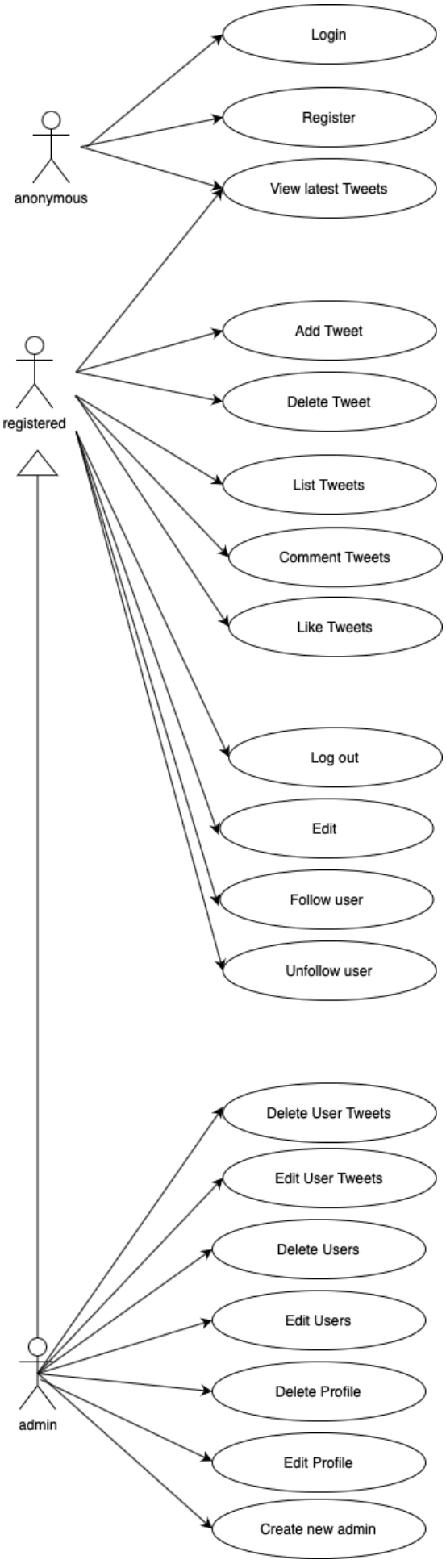
After the user enters all this information and clicks on the “Submit” button, the registration of the user will be performed in the database if all the fields have the correct format. It follows this schema:



When a registered user then submits the login (with username and password), a similar process is executed but instead of a database write we are doing a database read:



Types of Users



In our application we will define three types of users, that is the different situations where our application can be used.

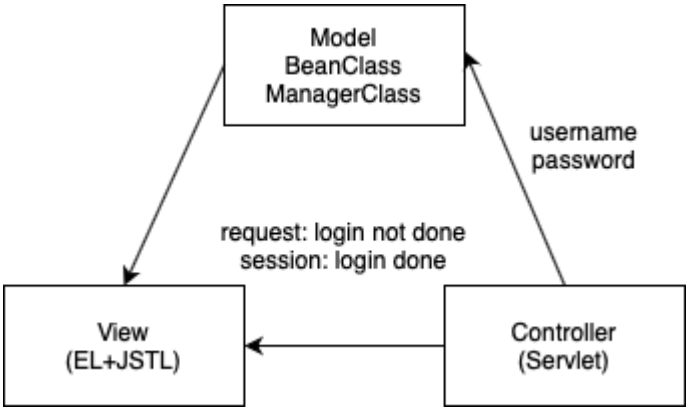
We will define a regular user, that needs to be registered and will have access to the basic functionalities of our website, such as posting messages, commenting on other messages, liking them, following and unfollowing other users and editing its own information.

In order to be able to moderate the website, we will define an admin user, that will have the same privileges as the regular users but will also be able to edit and delete other user accounts, tweets, etc. As well as being able to turn other users into admin users.

Finally, for those who want to take a look at our website without registering in it, we will define an anonymous user, that won't be able to interact with the database by any means but will be able to have a preview of the last post published in InspoWorks.

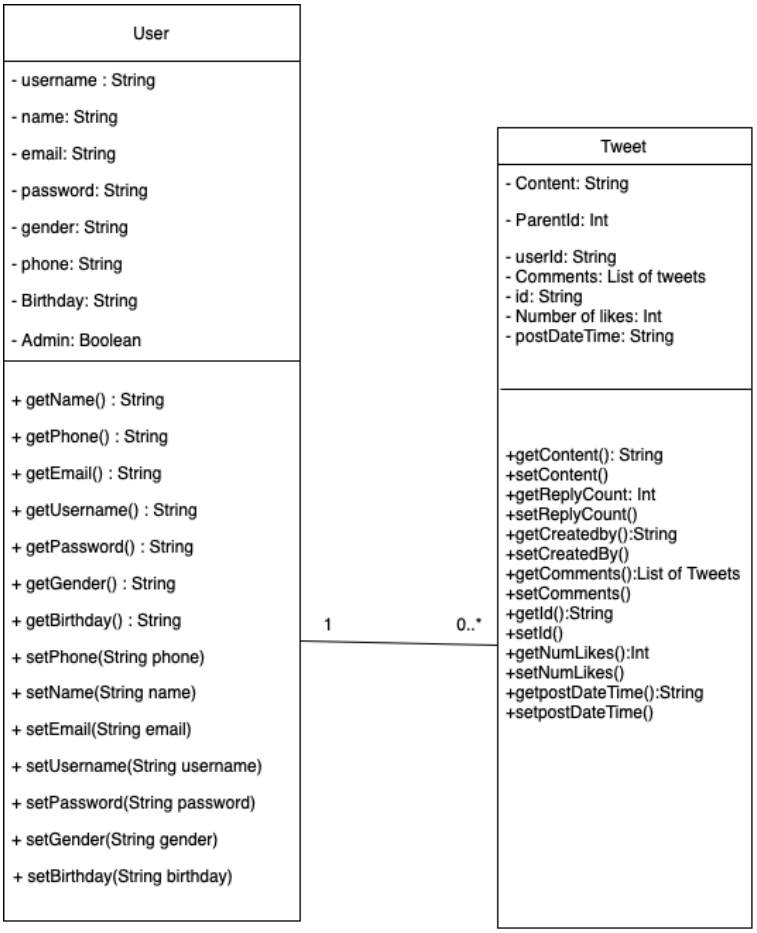
Project Structure

In order to organize the program structure, the SQL database and their respective relations, we will need to define an MVC, class and SQL diagrams.



The MVC diagram shows us the internal structure of the program, and how different parts (frontend and backend) will relate between each other using servlets.

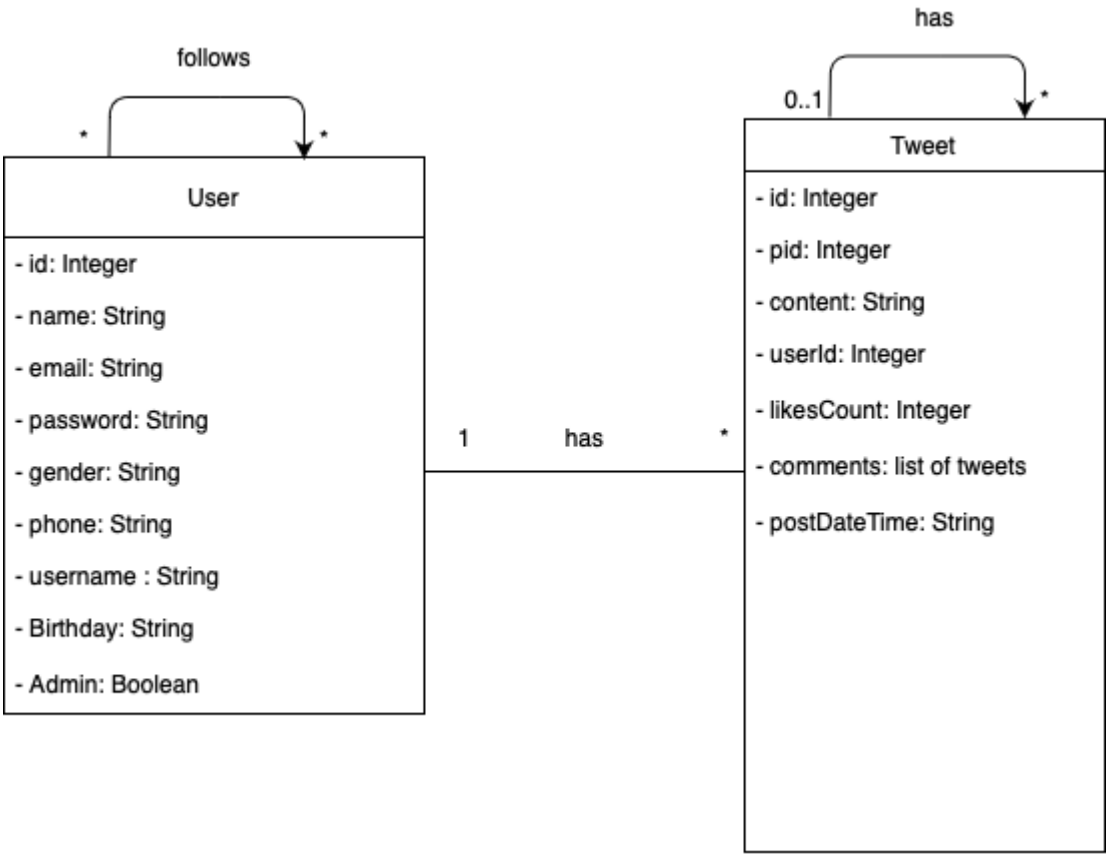
There will be two main classes that will need to be defined during the execution of the program, those will be a USER class and a TWEET class.



Those classes will store all the important information about each post and user, such as the name of a user, its phone number, the content of a tweet, etc.

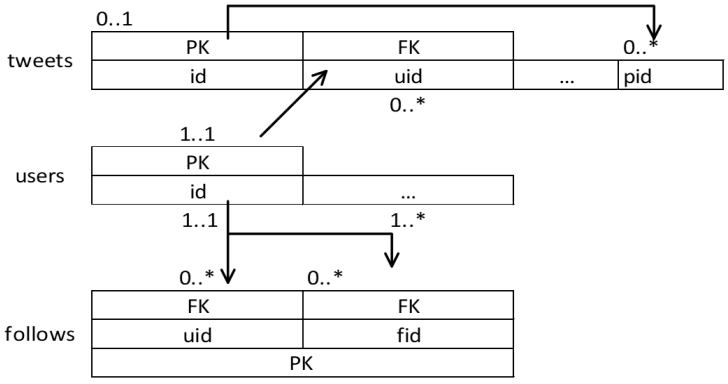
As we can see in the MVC diagram, we will have two manager classes that will be the ones in charge of using these USER and TWEET classes by providing certain functionalities to the program that will be requested by the servlets.

In order to store all the information beyond the execution of the program, we will need a SQL database, and to do that we will need to define the required tables and their relationships.



We will store the information of the USER and TWEET classes in those tables, by adding some information such as an id (unique identifier for each entry), the parent id of a tweet (that is, if a post is a comment to another post), etc. We will define another two tables. called “Follows” (that will be the one in charge to store the recursive relation of following) and “Like” (the one in charge of storing which user has liked which messages).

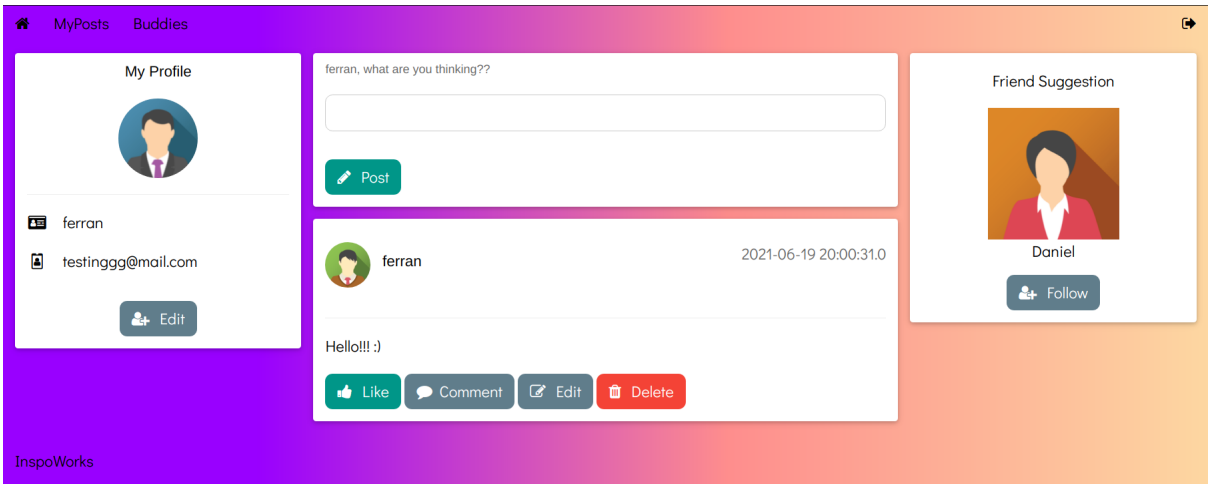
“Follows” will contain the follower ID and the followed ID, while “Like” will contain the user ID and the liked post ID.



Design - CSS

Our application is called InspoWorks because we want to showcase that this twitter is made for artists and creative people. We will use a colorful palette to also emphasize this idea, so we decided to set a background with a color gradient (from colder colors to warmer ones).
This application is not intended to be used in a very formal way so we are using casual and appealing font-types, as well as rounded buttons.

Once the users login, they will be directed to the home page, which looks like this:



We can see in this picture that the website is divided into 3 sections, a left column for my profile, a center wide column for the contents and a right column for the users suggestions.

About the contents column, we can see that there is a top bar for the user to create a new tweet, and below that he/she will see the latest tweets from the people they follow. For each tweet, they will have the option to like it, comment on it, or, if the user made the tweet, edit it or even delete it.

Since we liked the general structure provided by the W3 CSS library, we decided to keep it for our project but also adding another layer of CSS to have a higher customization, meaning that we have a specific CSS file inside the project folder. To access the different elements of the DOM, we are using the global selector (*) to give a general font type to all the elements of the website, and if we need a specific font type for a certain div we just select it by its id or class. Moreover, in order to change the color of the buttons, since we are still using the W3 CSS library, we just had to add the class w3-(required color) to the specifical button we needed to edit.