**Social Media Application - Django**

**Project Overview**

This project is a simple social media application built using **Django** (for the backend) and **HTML, CSS, JavaScript** (for the frontend). The application mimics a basic version of Facebook, where users can:

* Register and log in to the platform.
* Create, edit, and delete posts.
* View posts made by all users on the homepage.
* View and manage their personal posts on their profile page.

**Table of Contents**

1. Core features
2. [Setup and Installation Instructions](#setup-and-installation-instructions)
3. [Technologies Used](#technologies-used)
4. [Sample User Credentials](#sample-user-credentials)
5. [ER Diagram](#er-diagram)
6. [Additional Notes](#additional-notes)

**Core Features:**

* User Management:
  + User can register through a Registration Form
  + It has login and logout functionality
  + User can change their own password
* Post Mangement:
  + All posts are available in Home page
  + User-based posts (Logged-in users’ posts) are available at My Blog link. User can edit and delete their posts
  + Users are allowed to create their posts.
* Access Control:
  + Post creation can only be done by authenticated users.
  + Edition and deletion can be done by creator of such posts.
  + Related feedback messages are shown after every event such as create/edit/delete post, password change/user creation.

[**Setup and Installation Instructions**](#setup-and-installation-instructions)

* 1. **Clone the repository**

git clone https://github.com/your-username/social-media-app.git

cd social-media-app

* 1. **Create a virtual environment**

For Windows:

python -m venv venv

venv\Scripts\activate

For macOS/Linux:

python3 -m venv venv

source venv/bin/activate

* 1. **Install the required dependencies**

pip install django

pip install pillow

* 1. **Apply migrations**

python manage.py makemigrations

python manage.py migrate

* 1. **Create a superuser (optional, for admin access)**

python manage.py createsuperuser

You will be prompted to provide a username, email, and password.

Username: admin

Password: A147852369

* 1. **Run the server**

python manage.py runserver

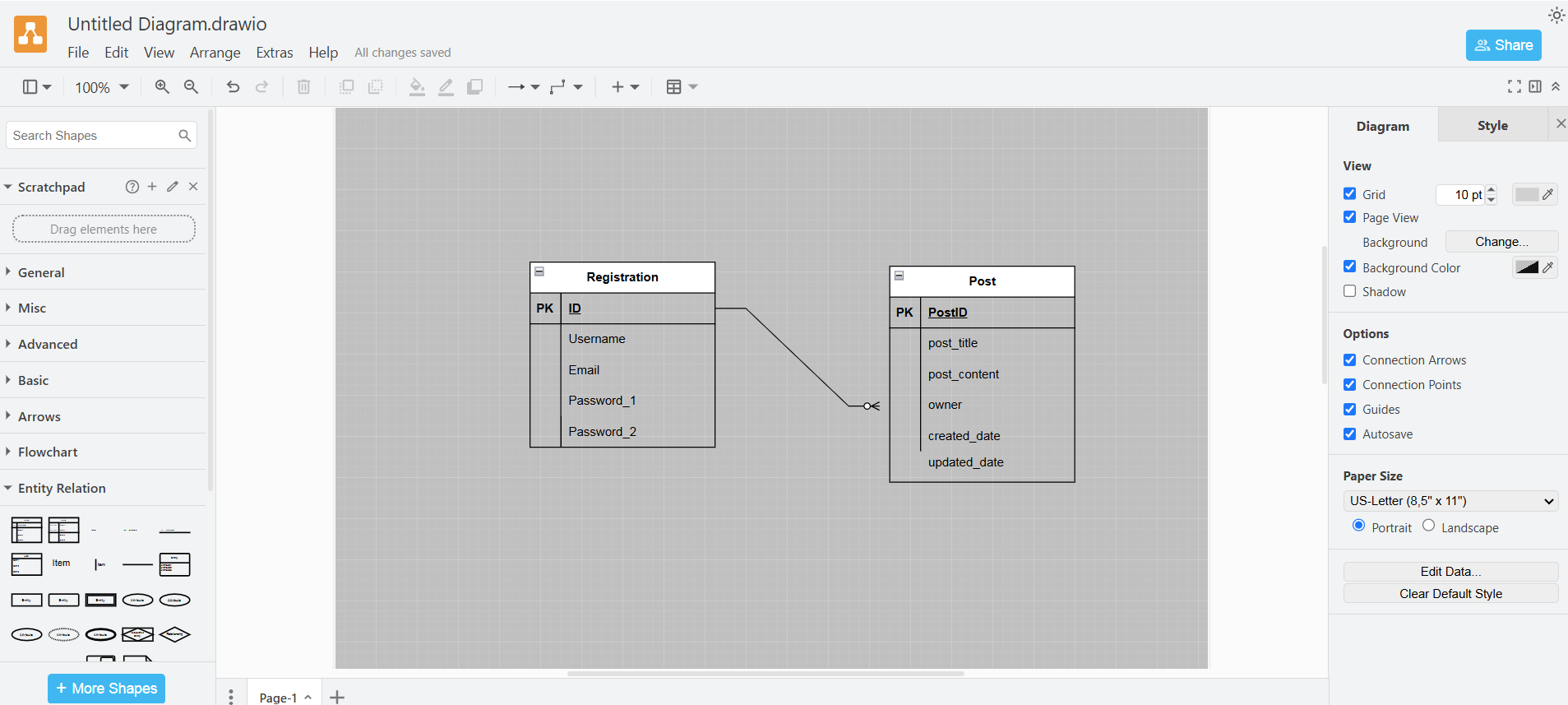
**Technologies Used:**

* **Backend:** Django (Python)
* **Frontend:** HTML, CSS, JavaScript
* **Database:** SQLite (default for Django projects)
* **Version Control**: Git (recommended to use GitHub for submissions)
* **Other:** Utilize Django’s built-in user model and authentication views.

**Sample User Credentials**

* **Admin User:**
  + **Username:** admin
  + **Password:** A147852369
* **Sample User 1:**
  + **Username:** test\_user
  + **Password:** t1234567
* **Sample User 2:**
  + **Username:** test\_user2
  + **Password:** t1234567

**ER Diagram**



**Additional Notes**

* Make sure to test all the core functionalities before submitting.
* The application is optimized for a simple interface and basic user management.
* The image upload feature is optional but included to extend the functionality of posts.

This README file covers all the key elements needed for setting up and using the social media application, including how to handle image file uploads. Let me know if you need any more adjustments!