

SWE 573 – SOFTWARE DEVELOPMENT PRACTICE
SOCIO Community Specific Management Information System
Final Report
Seher İrem Kurt
ID: 2022719114
Date: 20.05.2024

- Socio Deployment link → <http://35.219.175.104:8000/>
- Github link → <https://github.com/irem-kurt/Swe573-SeherIremKurt>
- Git tag version → <https://github.com/irem-kurt/Swe573-SeherIremKurt/releases/tag/v0.9.1>

HONOR CODE

Related to the submission of all the project deliverables for the Swe573 2024 Spring semester project reported in this report, Seher İrem Kurt I declare that:

- I am a student in the Software Engineering MS program at Bogazici University and am registered for Swe573 course during the Spring 2024 semester.
- All the material that I am submitting related to my project (including but not limited to the project repository, the final project report, and supplementary documents) have been exclusively prepared by myself.
- I have prepared this material individually without the assistance of anyone else with the exception of permitted peer assistance which I have explicitly disclosed in this report.

Seher İrem Kurt

Contents

Project Details.....	3
Design Documents.....	4
Initial Database Design.....	4
Use Case Diagram.....	5
Installation Instructions.....	6
Prerequisites.....	7
Database.....	7
Docker.....	8
Login Credentials for Logging Into The System.....	8
Demo Video of The Application.....	8

Project Details

Wiki of repo can be found in this link:

<https://github.com/irem-kurt/Swe573-SeherIremKurt/wiki/Community>

Issues and bug details can be found is this link:

<https://github.com/irem-kurt/Swe573-SeherIremKurt/issues>

Project follow up in repo can be found in this link:

<https://github.com/users/irem-kurt/projects/3>

Overview

SRS

Introduction

This page is to document the requirements of Community specific information management system.

Descriptions

- Content: Contributions in the form of text or various media types.
- Community: A collective of individuals sharing common interests.
- Creator: Administrator of a community, whether originally established or subsequently appointed by the initial creator.
- Moderator: A designated user responsible for overseeing content within a community.
- User: An individual utilizing the platform.
- Profile: Personal pages created by users to represent their information.
- Standard Search: Keyword-based exploration across communities or users.
- Advanced Search: Template-driven search, enabling specific queries such as posts within a particular community with image fields.
- Template: A framework established by the community creator, guiding members in structuring their posts and comprising mandatory and/or optional fields.

Functional Requirements

The requirements are consist of the following sections;

- Sign up and Login
- User Abilities
- Search
- Private and Public Communities
- Creating A Community
- Posts
- Reporting abuse

- Feed
- Content Creation and Management

Sign up and Login

1. Visitors shall be able to sign up the website.
2. Users shall be able to login the website.
3. Users shall use their e-mail and password to login to the website.
4. User shall login to the system once they correctly enter the credentials.
5. Users shall not use their username and password to login to the website.
6. The system shall check existing usernames and emails to prevent duplicate users when there is a new account.
7. If a user signs up successfully, the system shall navigate to the login screen.
8. If a user is on the sign-up page and s/he provides an existing email address the system shall display a warning message as "This email address is already registered."
9. If a user on the sign-up page and s/he provides a non-English character in the password the system shall display a warning message as "Invalid character."
10. If user does not provide correct password, the system shall display a warning message as "This password is not correct."
11. If a user attempts login in successfully, the system shall navigate to the main screen.
12. If user does not provide an email address, then the system shall display a warning message as "The email address is missing." in both register and login page.
13. If user does not provide a password, then the system shall display a warning message as "The password is missing." in both register and login page.

User Abilities

1. Type of users shall be User, Moderator, Community builders.
2. Community builders shall be the initial moderator but then they shall transfer moderation to the others if wanted.
3. Moderators shall transfer ownership before they delete their account.
4. Moderator shall be able to remove a community member if the community member does not comply with the community rules.
5. Moderator shall be able to assign moderator role to various community members.
6. Users shall have a profile page.
7. Users shall have personal pictures.
8. Users shall have their actual or nicknames.
9. Users should be able to edit username, email and profile picture.

10. A registered user shall be able to create community with the following information: unique community name, short description, privacy rules.
11. User shall be able to post content.
12. User can delete or edit her/his posts and comments
13. Users shall be able to follow each other.
14. User shall be able to delete his/her account.

Search

1. User shall search for posts and communities.
2. User shall do advanced search by selecting the template items to be searched, and then typing in the keywords for those items before initiating the search.
3. Users shall be able to search events by delivery time.
4. Users shall be able to search and filter communities.
5. Users shall be able to search and filter communities from main screen.
6. Users shall not be able to search and filter posts from main screen. 1.User shall search by entering a keyword into the search box and then clicking on the search button.

Private and Public Communities

1. Communities shall be private and public.
2. Communities shall have unique names.
3. Communities shall include posts with the post ids.
4. Users shall join private communities by only invitation.
5. Moderator shall be the only one that approve invitations.
6. Users shall join multiple communities.
7. Approving to join the private communities shall be provided by moderators of that communities.
8. The system shall have a default template with the following fields: title, description and image.
9. On the Create Community page, users shall select the community privacy as private.

Creating A Community

1. User shall create a community.
2. On the main page, s/he shall click on the create community button to create a community.
3. On the Create Community page, users shall click on the add template button to add a template to that community.

4. On the Create Community page, when they click on the add template button, the system shall navigate to the create template page.
5. On the Create template page, if users want to add new sections, they shall click on the add field button to add a field to that template.
6. On the Create template page, users shall save the content by clicking save button.
7. On the Create Community page, when users click on the save button the system shall navigate to that community page.
8. On the add field section, the user shall determine mandatory fields by clicking mandatory button in the related boxes.

Posts

1. Posts shall have a unique post ID.
2. The system shall assign a unique post ID to newly created post when a post is created by a user.
3. Post shall have location, text and picture.
4. Users shall be able to comment on other posts.
5. Upvoting and downvoting of posts shall be possible for users.

Reporting abuse

1. User shall report an abuse to moderator.
2. Moderators shall be able to view the abuse reports.
3. Moderators must respond to abuse reports and inform the reporter of an abuse.

Feed

1. The feed shall be consisting of user-generated content data.
2. The feed shall show random public communities.
3. The feed shall show random posts from communities available in the system after the registration.
4. The feed shall show random communities with the name, pictures and number of members icon.

Content Creation and Management

1. Mandatory fields shall be shown with "*" if they are left empty.
2. The user shall enter a link input if there is a link input field.
3. The user shall enter an image input by clicking on the select image button and selecting the image in PNG format on the local computer if there is an image input field.

4. The user shall enter an audio input by clicking on the select audio button and selecting audio in MP4 format on the local computer if there is an audio input field.
5. The user shall enter a video input by clicking on the select video button and selecting a video in MP4 format on the local computer if there is a video input field.
6. The user shall be able to enter text in upper of lower case if there is a text input field.
7. The user shall enter a date-time input including year, month, day, hour, minute, second, and time zone if there is a date-time input field.
8. The system shall create a post on that community when the user fills in all the mandatory fields and clicks on the create post button.
9. The system shall display an error message as "All mandatory fields must be filled" if the user does not fill in at least one mandatory field and clicks on the create post button.

Status of Functionalities

Functionality	Completed Tasks	Notes	Status
Register	Name,username , email,password, password	-requirement states login with email and password but we login with username and password.	Partly Done
Logout	Logout button log user out		Done
Login	username, password	email login is mising	email-passwo rd not implemented. Instead username-pa ssword is done
Create public community	authenticated user can create	-	Done
Create private community	authenticated user can create	-	Done
Create default post	Title and Description	-	Done
Create tempate post	Date, Titlei Audio, Video, Geolocation fiels	Only moderators and community owners can build	Done
Mandatory field in template creation	-	-	Done
Optional field in template creation	-	-	Done

User Roles	Moderator, Owner and Member	-	Done
User Roles	Owner makes Member as Moderator,	-	Done
User Roles	Owner remove Moderator dury	-	Done
User Profile	Edit Profile	-	Done
User Profile	View profile	-	Done
Post, Community, User	remove and delete	-	no
Search	Search in feed	-	Done
Search	Search posts in Community	-	Done
Advanced Search	Advanced Search in template specific in Community	-	Done
Like post	Like number is shown	-	Done
Dislike post	dislike number is shown	-	Done
Comment post	Comment with username shown	-	Done
Add audio content	User can post this content	-	Done
Add video content	User can post this content	-	Done
Add geolocation content	User can post this content	-	Done
Add txt content	User can post this content	-	Done
Add date content	User can post this content	-	Done

Status of Requirements

Implemented

Notes

Requirement

Yes

Visitors shall be able to sign up the website.

Yes

Users shall be able to login the website.

No	username and password login	Users shall use their e-mail and password to login to the website.
Yes		User shall login to the system once they correctly enter the credentials.
No	username and password login	Users shall not use their username and password to login to the website.
Yes		The system shall check existing usernames and emails to prevent duplicate users when there is a new account.
Yes		If a user signs up successfully, the system shall navigate to the login screen.
No		If a user is on the sign-up page and s/he provides an existing email address the system shall display a warning message as "This email address is already registered."
Yes		If a user on the sign-up page and s/he provides a non-English character in the password the system shall display a warning message as "Invalid character."
No		If user does not provide correct password, the system shall display a warning message as "This password is not correct."
Yes		If a user attempts login in successfully, the system shall navigate to the main screen.
No		If user does not provide an email address, then the system shall display a warning message as "The email address is missing." in both register and login page.
No		If user does not provide a password, then the system shall display a warning message as "The password is missing." in both register and login page.
Yes		Type of users shall be User, Moderator, Community builders.
Yes		Community builders shall be the initial moderator but then they shall transfer moderation to the others if wanted.
No		Moderators shall transfer ownership before they delete their account.
No		Moderator shall be able to remove a community member if the community member does not comply with the community rules.

No	Moderator shall be able to assign moderator role to various community members.
Yes	Users shall have a profile page.
Yes	Users shall have personal pictures.
Yes	Users shall have their actual or nicknames.
Yes	Users should be able to edit username, email and profile picture.
Yes	A registered user shall be able to create community with the following information: unique community name, short description, privacy rules.
Yes	User shall be able to post content.
No	User can delete or edit her/his posts and comments
Yes	Users shall be able to follow each other.
No	User shall be able to delete his/her account.
Yes	User shall search for posts and communities.
	User shall do advanced search by selecting the template items to be searched, and then typing in the keywords for those items before initiating the search.
Yes	
No	Users shall be able to search events by delivery time.
Yes	Users shall be able to search and filter communities.
Yes	Users shall be able to search and filter communities from main screen.
Yes	Communities shall be private and public.
No	Communities shall have unique names.
Yes	Communities shall include posts with the post ids.
Yes	Users shall join private communities by only invitation.
Yes	Moderator shall be the only one that approve invitations.
Yes	Users shall join multiple communities.
Yes	Approving to join the private communities shall be provided by moderators of that communities.

Yes		The system shall have a default template with the following fields: title, description .
Yes	This is implemented by lock emoji	On the Create Community page, users shall select the community privacy as private.
Yes		User shall create a community
Yes		On the main page, s/he shall click on the create community button to create a community.
Yes		On the Create Community page, users shall click on the add template button to add a template to that community.
Yes		On the Create Community page, when they click on the add template button, the system shall navigate to the create template page.
Yes		On the Create template page, if users want to add new sections, they shall click on the add field button to add a field to that template.
Yes		On the Create template page, users shall save the content by clicking save button.
Yes		On the Create Community page, when users click on the save button the system shall navigate to that community page.
Yes		On the add field section, the user shall determine mandatory fields by clicking mandatory button in the related boxes.
Yes		Posts shall have a unique post ID.
Yes		The system shall assign a unique post ID to newly created post when a post is created by a user.
Yes		Post shall have location, text and picture.
Yes		Users shall be able to comment on other posts.
Yes	Like and Dislike is implemented	Upvoting and downvoting of posts shall be possible for users.
No	Abuse not implemented	User shall report an abuse to moderator.

	Abuse not implemented	Moderators shall be able to view the abuse reports.
	Abuse not implemented	Moderators must respond to abuse reports and inform the reporter of an abuse.
Yes		The feed shall be consisting of user-generated content data.
Yes	Random community suggestion is not implemented but user see created communities.	The feed shall show random public communities.
No		The feed shall show random posts from communities available in the system after the registration.
Yes	not random	The feed shall show random communities with the name, pictures and number of members icon.
No		Mandatory fields shall be shown with "*" if they are left empty.
No		The user shall enter a link input if there is a link input field.
Yes		The user shall enter an image input by clicking on the select image button and selecting the image in PNG format on the local computer if there is an image input field.
Yes		The user shall enter an audio input by clicking on the select audio button and selecting audio in MP4 format on the local computer if there is an audio input field.
Yes		The user shall enter a video input by clicking on the select video button and selecting a video in MP4 format on the local computer if there is a video input field.
Yes		The user shall be able to enter text in upper of lower case if there is a text input field.
Yes		The user shall enter a date-time input including year, month, day, hour, minute, second, and time zone if there is a date-time input field.

Yes

The system shall create a post on that community when the user fills in all the mandatory fields and clicks on the create post button.

No

The system shall display an error message as "All mandatory fields must be filled" if the user does not fill in at least one mandatory field and clicks on the create post button.

Status of Deployment

My app is deployed: <http://35.219.175.104:8000/>

Chapter7- USER MANUAL

1- Opening screen of Socio

In navigate bar, user only sees Login and Register and can Search Communities.

Fig7-1: Navbar items before authenticating

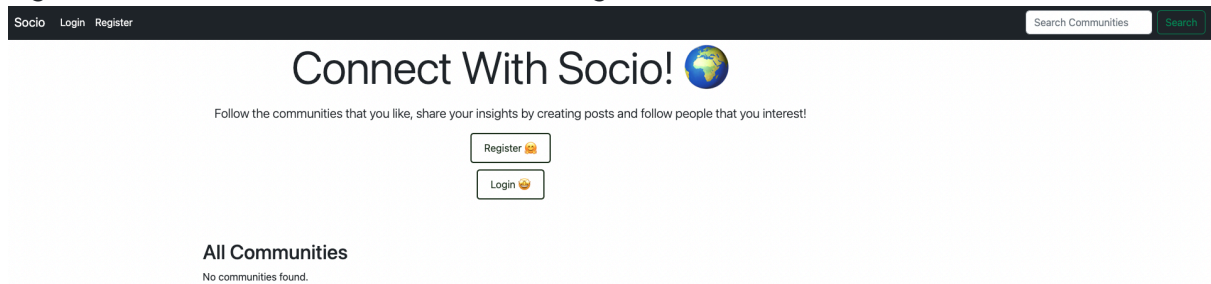


Table 7-1: Navbar items definitions before authenticating

Button	Role
Socio	Navigates to the feed that shows all public and private communities in website
Login	With username and password, signin to website.
Register	A stereotype register page that has name, username, location and password.

2- Click register and fill empty spaces. System navigates to login page after successfully being registered. Username and password is used for login to the system.

Not: If register is unsuccessful, user keeps staying in register page until errors like password not match are fixed.

3- Once register and then login, My Communities, Profile, Edit Profile and Logout buttons appear.

Fig 7-2: Navbar items after authenticating



Table 7-2: Navbar items definitions before authenticating

Button	Role
1- Create Community	Creates community
2-Socio	Navigates to the feed that shows all public and private communities in website
3- My Communities	Lists all the created or/and joined communities
4- Profile	Shows current profile setup
5-Edit Profile	User edits the profile page. Change options include "Name, Bio/referring to biography), Birth date, Location(Longitude,Lattitude), Picture (referring to profile picture)
6-Logout	Logout from website and navigates to feed.
7- Search in Communities box	User types a word, letter and to be returned results from communities if they exist in communities.
8-Search	User clicks the button after write the desired content in search box.

1- CREATE COMMUNITY PAGE

1a- Click Create Community to create community.

2a- Fill empty boxes. An example filled community form is given in the Figure7-3.

Fig 7-3: Create community form page

Socio
My Communities
Profile
Edit Profile
Logout
Search Communities
Search

Create Community

Name:

Description:

Rules:
2- Use picture template if you want to share a picture
3- Use health care and vet template for health issues
4- Feel free to add your likes and comments to

Picture: lazy cat.jpeg

Location:

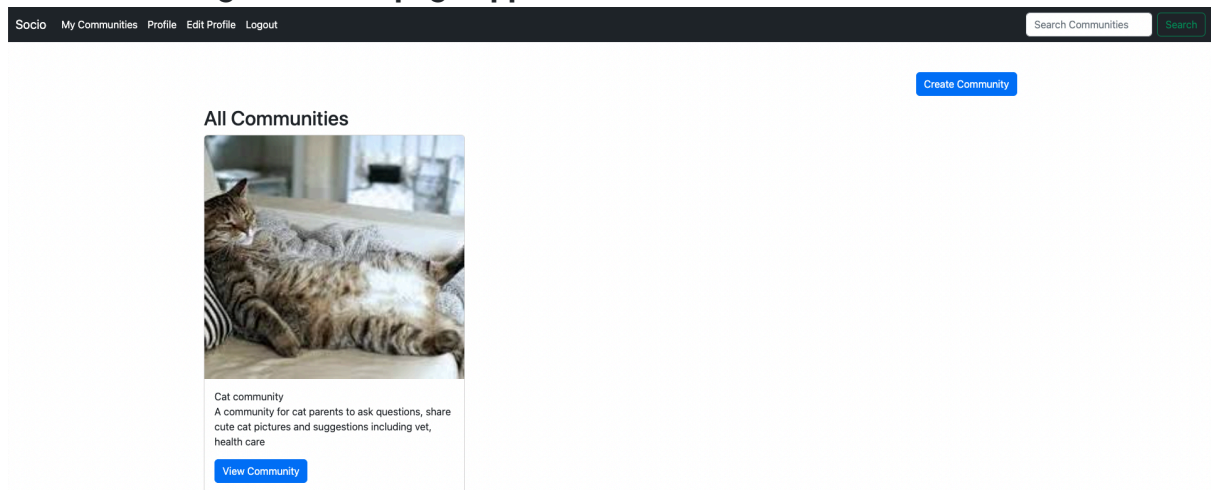
Is private: ☐

Create

Note: All fields are mandatory.

Note2: "Community creation is successful. Click My Communities to see your community." error appears once the community is created. Click My communities to see it.

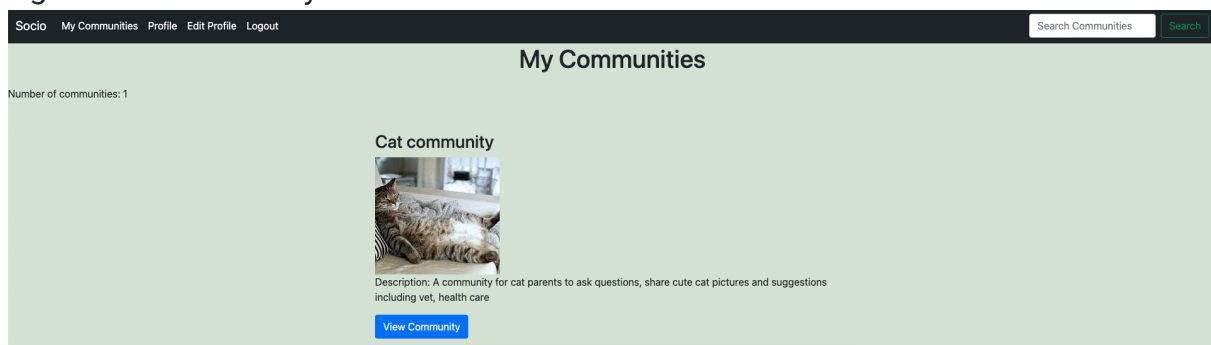
2- After clicking Socio this page appears.



3- After click My Communities this page appears

Not: Number of communities means the created or joined communities.

Fig: Public community



4- Click Edit Profile to add bio birthdate and upload profile picture and change name.

SocioMy CommunitiesProfileEdit ProfileLogout

Search CommunitiesSearch

Edit Profile

Name:

Bio:

Software Test Engineer and love art and good songs. A cat mum!

Bio:

Birth date:

Location:

Picture: Currently: [uploads/profile_pictures/default.png](#)
Change: ik.jpeg

5- After clickin Save change button, see the updated user profile by clicking "Profile" button.

User Profile



Name:

İrem Kurt

Bio:

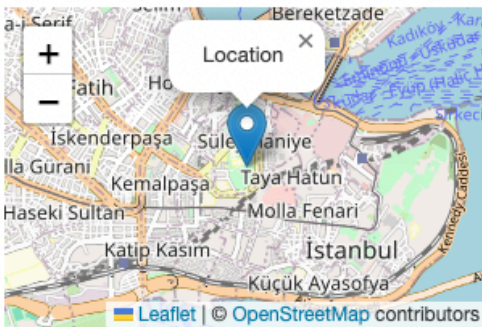
Software Test Engineer and love art and good songs. A cat mum!

Birth Date:

Dec. 12, 2000, midnight

Location:

41.012791,28.964593



Followers:

No followers yet.

Not: Followers are appear at the button once a user follows.

Community Detail

- 1- Click create post and select template.
- 2- Type Template Name

Create Template

Template Name: Vet Suggestion Template

Template Items:

Name	Type	Optional
Name of Vet	text	No
Location of Vet	location	No

Name ☐ Optional

3- Fill the box here location of vet is written. Then, select type of field as location. Click Add Item button.

Create Template

Template Name: Vet Suggestion Template

Template Items:

Name	Type	Optional
Name of Vet	text	No
Location of Vet	location	No

Location of Vet ☐ Optional

- ✓ Text
- Image
- Video
- Audio
- Date
- Location

4- Field is added. You can add many field as you wish. This is the logic.

Create Template

Template Name: Vet Suggestion Template

Template Items:

Name	Type	Optional
Name of Vet	text	No
Location of Vet	location	No

Name ☐ Optional

Default Post Detail

Please note that the profile emoji next to username means user “irku” post this.

- Comments has two sections that are separated by “ : ”. Left side the user that comments and right side is the actual comment.
- Like and dislike options are functioning.

irku

This is my default post

Default post definition

Like

0

Dislike

1

Comments

• irku: Please be more creative!

Enter your comment

Add Comment

Template Post Detail

Search in Posts

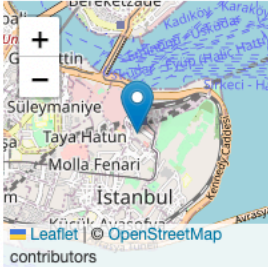
Advanced Search

Posts

irem Kurt1

Name of Vet
Efe Hakan

Location of Vet
41.01224,28.976018



Details of treatment
Teeth operation is done successfully to my 2 years old cat.

Date
March 15, 2023, 10:47 a.m.

Like

0

Dislike

0

Comments

Enter your comment

Add Comment

CHAPTER 8

8.1 USER TESTING

Suggested browser: Chrome

TC1- Register a user and then login:

Prerequisite: N/A			
Steps	Expected	Actual	Notes
1- Go to http://35.219.175.104:8000/	Homepage is opened in browser	Pass	
2- Click Register button	Register page is opened	Pass	
3- Fill name, username, email, password, confirm password	name, username, email, password, confirm password are filled	Pass	Some Invalid errors are not shown. Some of them are shown. -Only not matching password error is given - This password is too short. It must contain at least 8 characters. - The password is too similar to the username.
3- Click “Register” button	User is successfully created	Pass	Automatically navigates to login page
4- Enter “username” and “password”	Login is successful and navigates to home page	Pass	
5- Click login button	Goes to feed page	Pass	

TC2-Create a public community:

Prerequisite: Go to: http://35.219.175.104:8000/ and Login to system			
Steps	Expected	Actual	Notes
1- Click Socio(logo)	All communities are listed	Pass	
2- Click “Create New Community” button	Create community form is opened	Pass	

3- Fill name, rules, description, geo location, chose picture	Except selecting “is_private” button, other sections are filled	Pass	
4- Click “Create” button	Creation is successful	Pass	

TC3- Create a private community:

Prerequisite: Go to: http://35.219.175.104:8000/ and Login to system			
Steps	Expected	Actual	Notes
1- Click Socio(logo)	All communities are listed	Pass	
2- Click “Create New Community” button	Create community form is opened	Pass	
3- Fill name, rules, description, geo location, chose picture	Boxes are filled	Pass	
4- Select “Is_private” box	Box is selected	Pass	
5- Click “Create” button	Private Community Creation is successful	Pass	

8.2UNIT TESTING

I had very limited time and unfortunately, I could not organize myself to use test driven development approach. Therefore, I could not set up pytest to my environment. Instead, I created “tests” file and inside I created “test_views.py” file. I created unit tests of community create and post create as seen below.

```
(venv) irku@192 commansys % python manage.py test socio
Found 5 test(s).
Creating test database for alias 'default'...
System check identified no issues (0 silenced).
.....
-----
Ran 5 tests in 1.139s

OK
Destroying test database for alias 'default'...
(venv) irku@192 commansys %
```

To run unit tests in your local:

1- I use sqlite so please paste this to settings(or alternatively you can uncomment this part in settings.py:

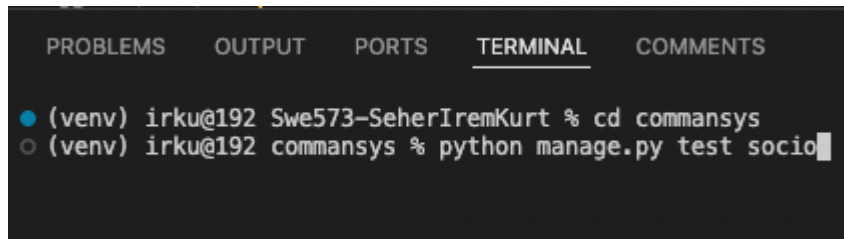
Use SQLite for testing

```
DATABASES = {
    'default': {
```

```
'ENGINE': 'django.db.backends.sqlite3',  
'NAME': BASE_DIR / 'db.sqlite3',  
}  
}
```

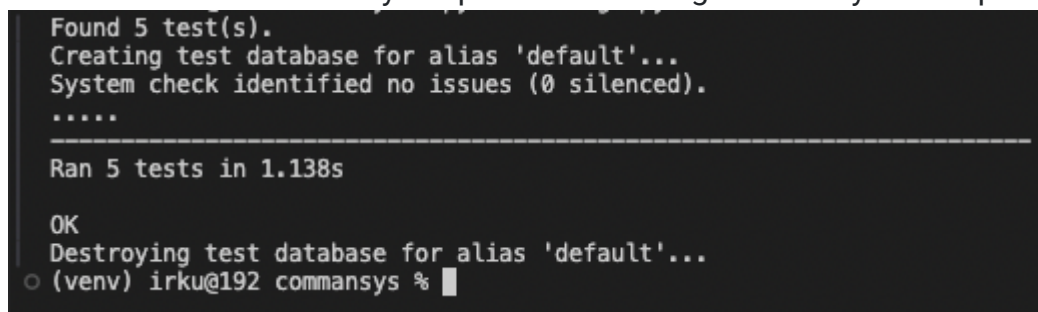
2- Save settings.py

3- Write these commands to the terminal in order.



```
PROBLEMS OUTPUT PORTS TERMINAL COMMENTS  
● (venv) irku@192 Swe573-SeherIremKurt % cd commansys  
○ (venv) irku@192 commansys % python manage.py test socio
```

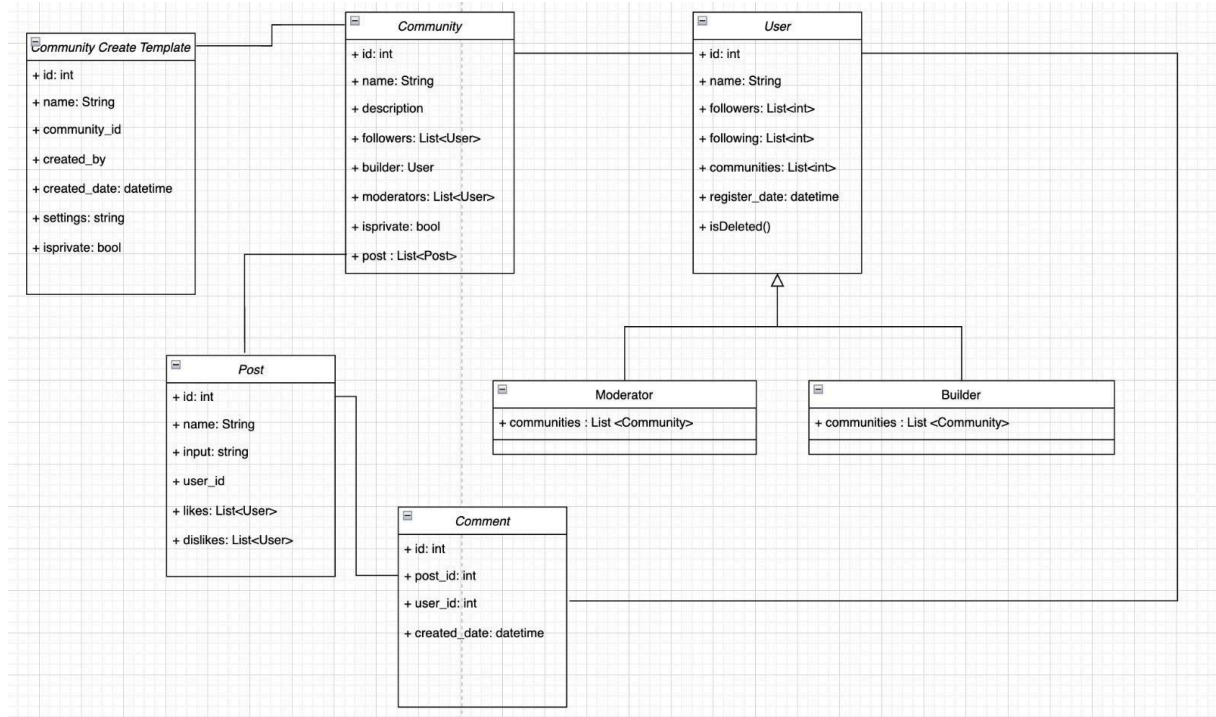
3- Find the test status. They are passed as stating "OK" in my local repo.



```
Found 5 test(s).  
Creating test database for alias 'default'...  
System check identified no issues (0 silenced).  
.....  
-----  
Ran 5 tests in 1.138s  
  
OK  
Destroying test database for alias 'default'...  
○ (venv) irku@192 commansys %
```

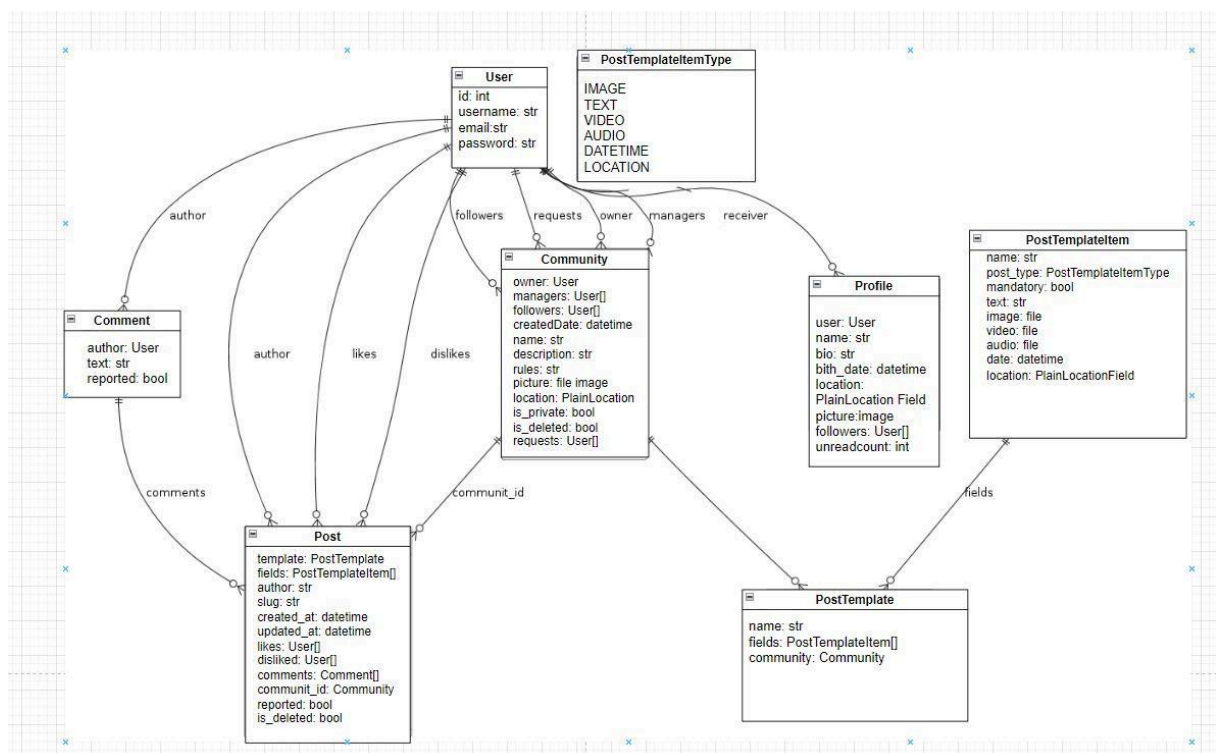

Design Documents

Initial Database Design



Updated Database Design

Crow's Foot Notation Entity Design Diagram



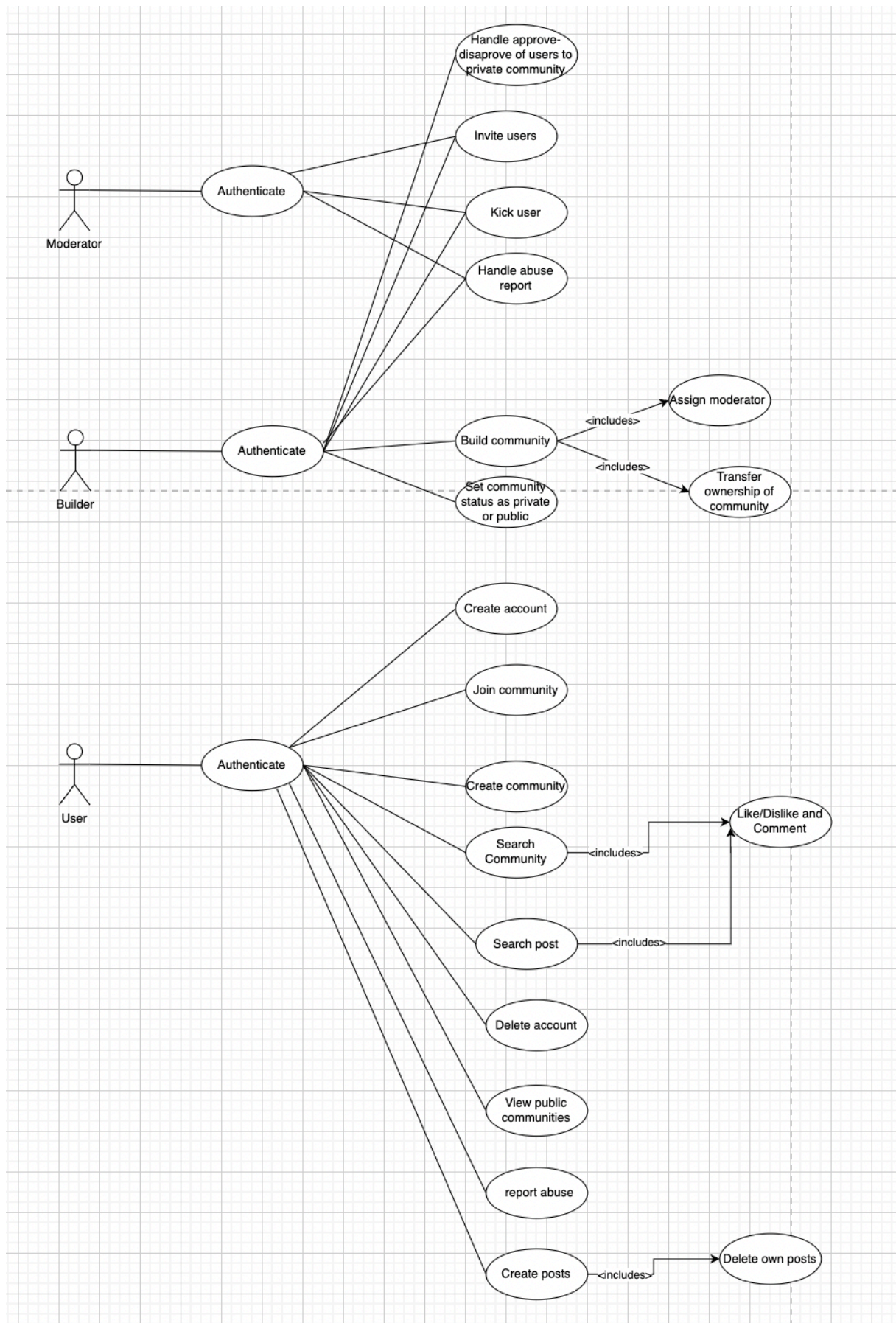
Key Differences between previous and current database designs

- **User and Profile Separation:**
The updated design separates Profile from User to handle profile-specific details separately. This makes the design more modular and easier to extend, as profile information can be managed independently of the core user data.
- **Community Model Expansion:**
The updated Community model includes additional fields such as rules, picture, location, is_deleted, and requests. This provides a more detailed view of communities, allowing for richer functionality and better data organization.
- **Post and Post Template:**
The Post model in the updated design uses template and fields to allow for dynamic post structures. This is an improvement over the static post structure in the initial design, making the system more flexible and adaptable to different content types.
- **PostTemplateItem and PostTemplateType:**
The addition of PostTemplateItem and PostTemplateType models introduces a way to define various types of post content (e.g., text, image, video). This makes sure the ability to customize and validate posts based on their type.
- **Comment Model Simplification:**
The Comment model in the updated design is simplified, focusing on the author, text, and reported status, rather than linking directly to post_id and user_id. This reflects a more straightforward approach to managing comments and their authorship.
- **Removed Community Create Template:**
The initial design had a Community Create Template model, which is not present in the updated design. This might indicate a shift towards integrating community creation functionalities directly within the Community model itself or handling it through a different process in the application logic.

Reflection on Database Design Update

The transition to the Crow's Foot notation and the restructuring of the database design show a significant improvement in the modularity, flexibility, and clarity of the system. By introducing specialized models like Profile, PostTemplate, and PostTemplateItem, the updated design supports more complex and dynamic interactions within the application. Additionally, the new relationships and attributes provide a more comprehensive representation of the entities involved, enhancing the system's ability to manage and scale user interactions, community features, and content customization effectively. This update demonstrates a thoughtful progression towards a more robust and maintainable database architecture.

Use Case Diagram



Installation Instructions

Prerequisites

1. Python 3.11
2. Django 2.2.24
3. Visual Studio Code
4. Postgres 14
5. Pg Admin
6. Docker Desktop

Database

- 1- Postgres 14 is used as database.
- 2- Pg Admin is used to manage the database relation tables

Database info is kept in “.env file” and called in settings.py.

Steps to run app in local:

- 1- For database, the db configuration must be like following in settings.py.

Settings.py database configuration is given below:

```
DATABASES = {  
    'default': {  
        'ENGINE': 'django.db.backends.postgresql',  
        'NAME': os.environ.get('DB_NAME', 'postgres'),  
        'USER': os.environ.get('DB_USER', 'postgres'),  
        'PASSWORD': os.environ.get('DB_PASSWORD', 'postgres'),  
        'HOST': os.environ.get('DB_HOST', 'postgres'),  
        'PORT': os.environ.get('DB_PORT', '5432'),  
    }  
}
```

- 2- You must create “.env” file in commansys app by right click, create file, name file as “.env” and click enter and paste the following section inside of it. (DB_HOST=localhost must be)

.env file database configuration is given below:

```
DEBUG=True  
SECRET_KEY=django-insecure-xudd*ez8i-fj529qdbt-$7u594-b^18sh!8rmqv(+0vc%)*wzh  
DB_NAME=postgres  
DB_USER=postgres  
DB_PASSWORD=postgres  
DB_HOST=postgres  
DB_PORT=5432
```

Docker

- You need to download docker from <https://www.docker.com/products/docker-desktop/>

- cd to App folder

To run Docker for my project, follow the following steps in your local.

- 1- Open vscode
- 2- Open docker desktop (must be ready to run)
- 3- Open pg admin having postgres version 14.(not required)
- 4- git clone <https://github.com/irem-kurt/Swe573-SeherIremKurt>
- 5- cd commansys
- 6- docker compose build
- 7- docker compose up
- 8- go to: <http://0.0.0.0:8000/>
- 9- If any tcp port error is gotten, open by double clicking docker-compose.yaml which is located in the same directory. If no error occurs, proceed to step 12.
- 10- Change ports to this by removing the second 5432.

```
12
13     postgres:
14         image: postgres:14
15         ports:
16         - "5432"
17         environment:
18         - "POSTGRES_HOST_AUTH_METHOD=trust"
```

11- Reapply the steps between 6 to 8.

12- Register to the system and then login.

Login Credentials for Logging Into The Deployed System

User1:

Username: irku

Password: iremkurt19

User2:

Username: ali

Password: irku1919

Demo Video of The Application

Demo video of the Application can be reached below with the following link:

<https://drive.google.com/drive/folders/16vLC6LJaPSBdEJUOsplok67AqdRhrK4E?usp=sharing>