**Criterion B**

1. **Database diagrams**
   1. **User model**

|  |  |  |
| --- | --- | --- |
| **Field** | **Datatype** | **Comments** |
| id (primary key) | String | UUID of a fixed length unique for every user. |
| first\_name | String | Holds the first name of a user in order to display it to other users. |
| email | String | E-mail address used for logging in. |
| password | Alphanumeric | Holds an encrypted password of a user. |
| isd\_code | String | ISD code of user’s phone number. |
| phone\_number | String | User’s phone number. |
| rating | Integer | Used to position user’s posts on the website. When one’s posts are reported the rating decreases. |
| is\_superuser | Boolean | Holds a value, whether the user is an admin or not. |

* 1. **Post model**

|  |  |  |
| --- | --- | --- |
| **Field** | **Datatype** | **Comments** |
| id (primary key) | String | UUID of a fixed length unique for every post. Different alphabet is used so that users and posts have different styles of ids. |
| author | Foreign Key | Foreign key to the author’s User model. |
| name | String | Holds the first name of a user in order to display it to other users. |
| description | String | Description of the post. |
| date\_created | Datetime | Date and time when the post was created. |
| image | Multimedia | An optional image for the post. |
| category | Foreign Key | Foreign key to the Category model, which holds the categories for posts. |
| rating | Integer | Used to position user’s posts on the website. When one’s posts are reported the rating decreases. |
| area | String | The area which is covered by the post. |
| type | String | Either ‘looking for help’ or ‘offering help’. |
| visible | Boolean | Holds a value, whether the post is visible or not. Used for example to hide instead of delete posts when they are being reported. |

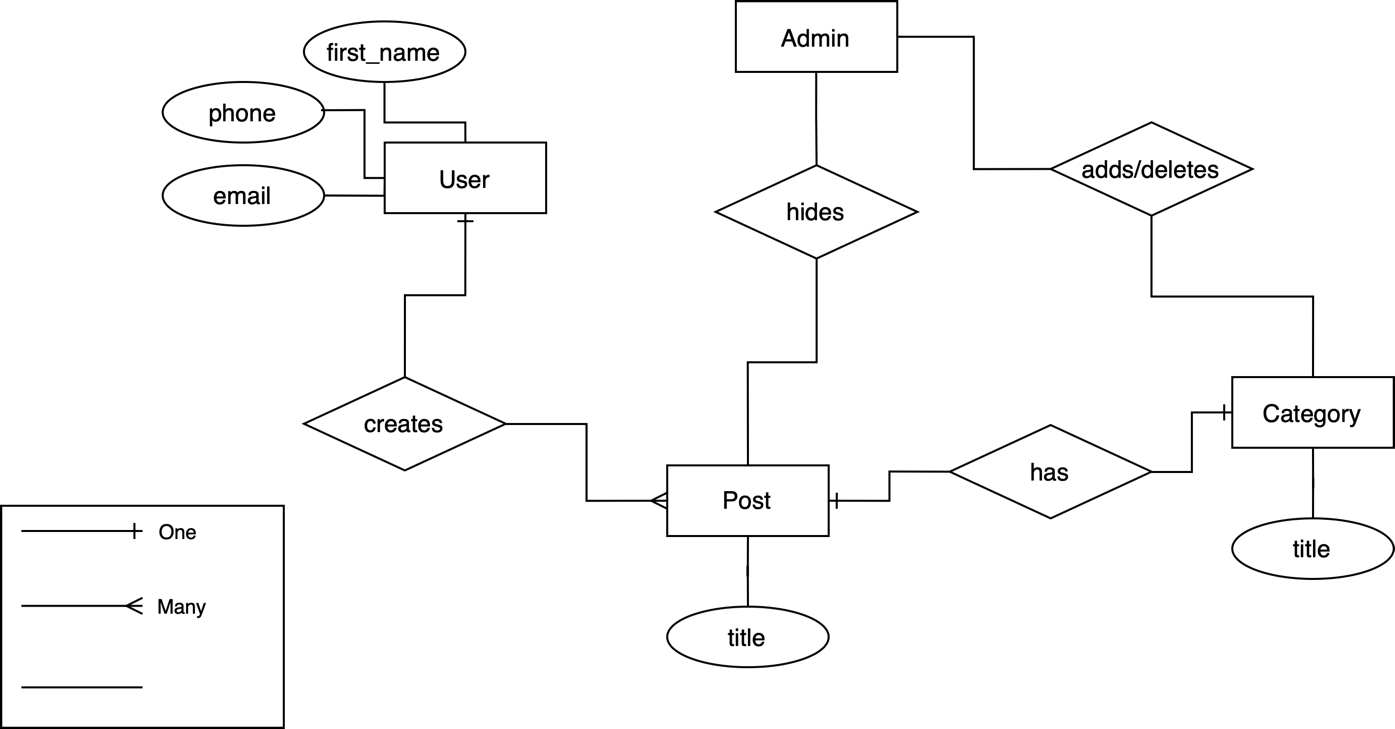
* 1. **Category model**

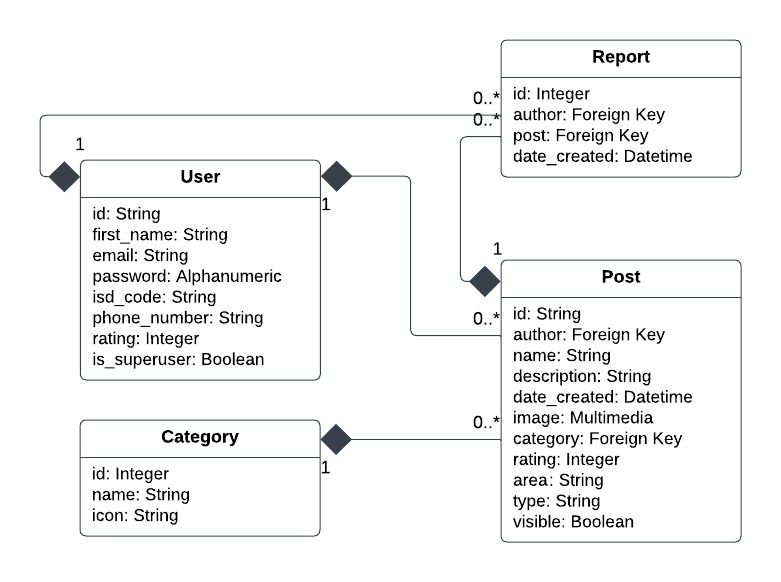
|  |  |  |
| --- | --- | --- |
| **Field** | **Datatype** | **Comments** |
| id (primary key) | Integer | Autoincremented ID. |
| name | String | Name of the category. |
| icon | String | Holds the name of the class for a specific icon for the class. |

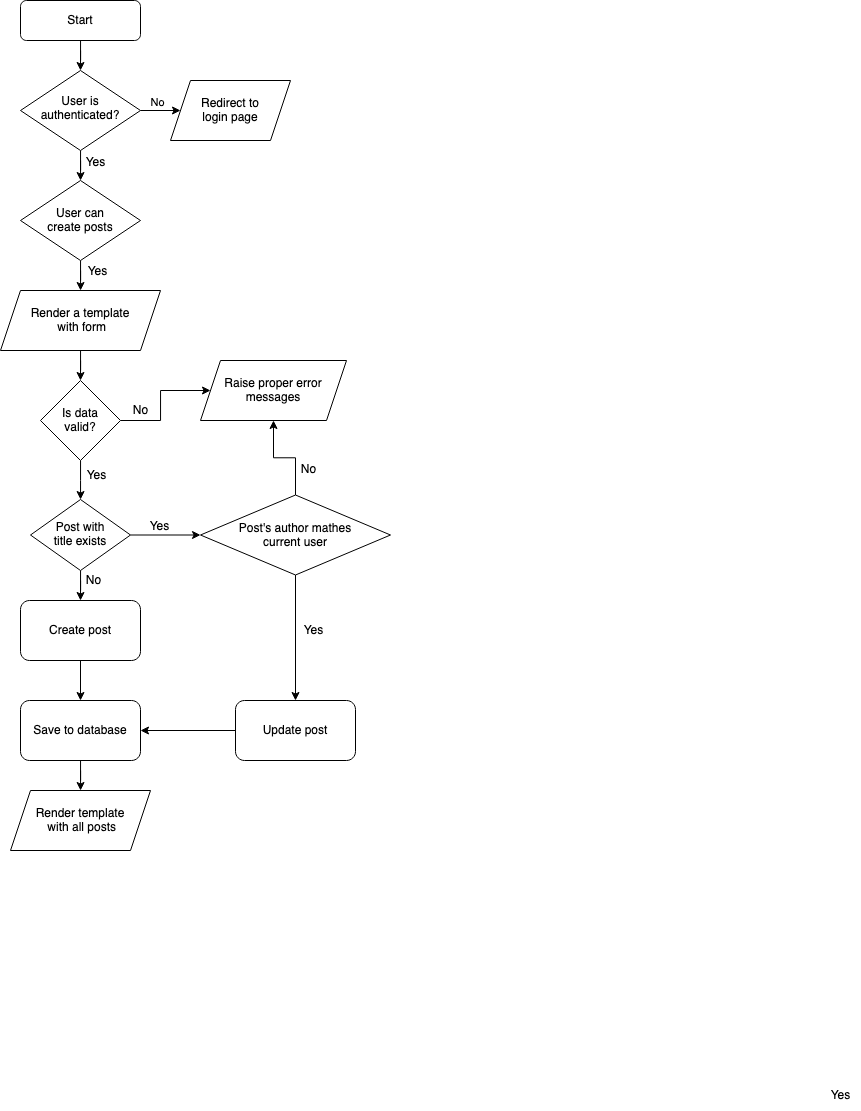
* 1. **Post report model**

|  |  |  |
| --- | --- | --- |
| **Field** | **Datatype** | **Comments** |
| id (primary key) | Integer | Autoincremented ID. |
| author | Foreign Key | User that created the report. |
| post | Foreign Key | Post that is being reported. |
| date\_created | Datetime | Date and time when the report was made. |

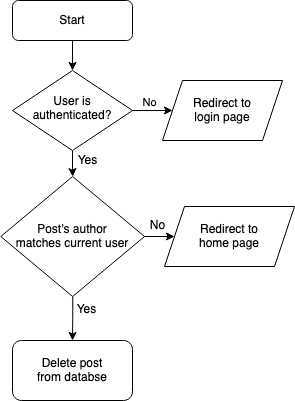
1. **Entity-Relationship Diagram**

****

1. **UML diagram**
2. **Flowcharts**
   1. **Create/update post**



* 1. **Delete post**

****

* 1. **Report post**

1. **Interface**
   1. **Home page**
   2. **Posts page**



* 1. **Create post**



* 1. **Login page**
  2. **Signup page**



1. **Test plan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Action to be tested** | **Test method** | **Expected result** | **Outcomes of the test** | **Success criteria** |
| Users are able to create accounts | Go to the Join page. Fill form with appropriate data. Submit. | User object created in the database with given data. | As expected. | 1 |
| Users are able to edit accounts | Go to Account page. Fill the form with appropriate data. Save. | User object updated in the database with given data. | As expected. | 1 |
| Users can login with the account | Go to Login page. Fill the form with credentials of a user account that already exists in the database. Submit. | User object is saved in the cookies on user’s device. | As expected. | 2 |
| Users can reset their password | Go to Forgot my password page. Fill the form with appropriate data. Submit.  Click link from the e-mail. Fill form with appropriate data. Save. | An e-mail sent to the user with a unique link. Link leads to a page with reset password form.  User’s password updated in the database. | As expected. As expected. | 2 |
| Users can create posts | Go to Add page. Fill the form with appropriate data. Submit. | Post object is created in the database with given data. | As expected. | 3 |
| Users can modify posts | Go to a post created with currently logged in account. Click Edit. Fill form with appropriate data. Submit. | New updated Post object is created in the database, and the old one is deleted. | As expected. | 3 |
| Users can delete their posts | Go to a post created with currently logged in account. Click Delete. | Post object is created from the database. | As expected. | 3 |
| Users can report posts | Go to a post created by a different user. Click Report. | A Report object is created in the database with the id of the post that has been reported and the user that did it. | As expected. | 4 |
| Users can search for a post with given keyword in the title/description, specific area/category | Type data in the search bar. | List of Post objects (filtered according to the data given by the user) is displayed. | As expected. | 5 |
| User’s ‘rights’ based on rating | Change rating of a Post object. | If post’s author’s rating is less than 1, user’s fields, can create posts, can report, can message, are changed to false in the database. | As expected. | 6 |
| Post’s visibility based on its rating | Change rating of a Post object. | If post’s rating is less than 1, the field is visible is changed to false in the database. | As expected. | 6 |
| Users can message each other | Go to Messages page. Enter a user. Type message. Send. | Message object is created in the database with appropriate data. | As expected. | 7 |
| Validation of inputs | Type invalid data to any input field. | Proper error message raised and shown to the user. | As expected. | 8 |
| Responsive design of the website | Change the width of the device on which the website is viewed on. | Elements on the website changing width, height based on the screen’s size. | As expected. | 9 |

1. **Record of tasks**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Task number** | **Planned action** | **Planned outcome** | **Time estimate** | **Target completion** | **Criterion** |
|  | Find a client |  | 5 days | 10.04.2022 | A (analysis) |
|  | Examine and research the problem | Come out with a solution for the problem that the client has assigned me | 7 days | 17.04.2022 | A |
|  | Initial interview my client | Discuss with the client about the project to have a clear understanding of the issue he has | 1 day | 28.04.2022 | A (Collecting human resources) |
|  | Write Criterion A | Plan the project and write the success criteria for it | 3 days | 30.04.2022 | A |
|  | Design basic UI | Make a rough sketch of how the user interface will look like | 2 days | 5.05.2022 | B |
|  | Design database scheme | Design the database scheme based on which the data will be stored inside of the database | 2 days | 10.05.2022 | B |
|  | Design flowcharts | Create flowcharts for the website structure | 3 days | 14.05.2022 | B |
|  | Create test plan | Create test plan to add new fields throughout the project | 1 day | 15.05.2022 | B |
|  | Asdfnaslfasd | Feedback from the client. Meeting with the supervisor |  |  |  |
|  | Start a Django project | Create a Django project on a local machine | 1 day | 6.06.2022 | C |
|  | Create and connect to a database | Create a MySQL database and connect to it with Django | 1 day | 7.06.2022 | C |
|  | Write code | Write code for the website | 14 days | 30.06.2022 | C |
|  | 1st feedback from the client | Consult the first draft of the website with the client and get feedback | 1 day | 5.09.2022 | A |
|  | Update the website | Update the website to match with client’s expectations | 7 days | 11.09.2022 | C |
|  | Final feedback from the client | Present the website to the client and consult and additional fixes | 1 day | 24.10.2022 | A |
|  | Last corrections to the code | Make last corrections to the code and fix any bugs | 7 days | 6.11.2022 | C |
|  | Finish Criterion B | Finish Criterion E | 2 days | 7.11.2022 | B |
|  | Write evaluation | Write Criterion E | 2 days | 13.11.2022 | E |
|  | Submit IA |  |  |  | All |