

Social Networking API Documentation

Base URL

`http://localhost:8000/api/`

Authentication

The API uses JWT (JSON Web Token) for authentication. Users must be authenticated to access all endpoints except for login and signup.

Signup

Endpoint: `POST /api/signup/`

Description: Create a new user account.

Request Body:

```
{ "email": "user@example.com", "username": "username", "password": "password123" }
```

Response:

- 201 Created

```
{ "id": 1, "email": "user@example.com", "username": "username" }
```

Login

Endpoint: `POST /api/token/`

Description: Obtain JWT tokens for authentication.

Request Body:

```
{ "email": "user@example.com", "password": "password123" }
```

Response:

- 200 OK

```
{ "refresh": "your-refresh-token", "access": "your-access-token" }
```

Search Users

Endpoint: `GET /api/users/search/`

Description: Search for users by email or name.

Request Parameters:

- **q** (string, required): The search keyword.

Example Request:

GET /api/users/search/?q=am

Response:

- 200 OK

```
[ { "id": 1, "email": "amar@example.com", "username": "Amar" },  
  { "id": 2, "email": "aman@example.com", "username": "Aman" },  
  { "id": 3, "email": "abhirama@example.com", "username": "Abhirama" } ]
```

Send Friend Request

Endpoint: POST /api/friend-requests/send/

Description: Send a friend request to another user.

Request Body:

```
{ "receiver_id": 2, "message": "Hi, let's be friends!" }
```

Response:

- 201 Created

```
{ "id": 1, "sender": 1, "receiver": 2, "status": "pending", "message": "Hi, let's be friends!",  
  "created_at": "2024-05-17T18:42:38.767251Z", "updated_at": "2024-05-17T18:42:38.767251Z" }
```

Accept/Reject Friend Request

Endpoint: POST /api/friend-requests/<id>/respond/

Description: Accept or reject a friend request.

Request Body:

```
{ "status": "accepted" // or "rejected" }
```

Response:

- 200 OK

```
{ "id": 1, "sender": 1, "receiver": 2, "status": "accepted", "message": "Hi, let's be friends!",  
  "created_at": "2024-05-17T18:42:38.767251Z", "updated_at": "2024-05-17T19:00:00.000000Z" }
```

List Friends

Endpoint: GET /api/friends/

Description: List all friends (users who have accepted the friend request).

Response:

- 200 OK

```
[ { "id": 2, "email": "friend@example.com", "username": "Friend" } ]
```

List Pending Friend Requests

Endpoint: GET /api/friend-requests/pending/

Description: List all pending friend requests received by the authenticated user.

Response:

- 200 OK

```
[ { "id": 1, "sender": { "id": 1, "email": "sender@example.com", "username": "Sender" }, "status":  
"pending", "message": "Hi, let's be friends!", "created_at": "2024-05-17T18:42:38.767251Z",  
"updated_at": "2024-05-17T18:42:38.767251Z" } ]
```

Models

User

- **email** (EmailField, unique)
- **username** (CharField, max_length=150, unique, blank=True)
- **created_at** (DateTimeField, auto_now_add=True)
- **updated_at** (DateTimeField, auto_now=True)
- **is_active** (BooleanField, default=True)
- **is_staff** (BooleanField, default=False)
- **is_superuser** (BooleanField, default=False)

FriendRequest

- **sender** (ForeignKey to User, related_name='sent_requests', on_delete=models.CASCADE)
- **receiver** (ForeignKey to User, related_name='received_requests', on_delete=models.CASCADE)
- **status** (CharField, choices=[('pending', 'Pending'), ('accepted', 'Accepted'), ('rejected', 'Rejected')])
- **created_at** (DateTimeField, auto_now_add=True)
- **updated_at** (DateTimeField, auto_now=True)
- **message** (TextField, blank=True)

Friendship

- **user1** (ForeignKey to User, related_name='friendship_user1', on_delete=models.CASCADE)
- **user2** (ForeignKey to User, related_name='friendship_user2', on_delete=models.CASCADE)
- **created_at** (DateTimeField, auto_now_add=True)