1. **EndPoint:**

1.1-**Endpoint**: /register

**Method**: POST

**Description**: Create new user and store it in DataBase

**RequestBody (JSON){**

Username = “string”

Pass = “string”

**}**

**ResponseBody (JSON){**

// 201 success

Message = “User registered successfully.”

**}**

**Error codes {**

// 400 Bad request

Message = “Invalid request format.”

// 409 conflict

Message = “username is already exist ”

**}**

**1.2** **Endpoint** /login

**Method** : POST

**Description** : create session and login the user

**Request Body:**

**{**

Username **: “”,**

Password **: “”**

**}**

**Respond Body:**

**{**

// 201 success

Message **: “**Log in successfully**”**

**}**

**//** failed

**{**

Message : “Invaled username or Password”,

Error : 401 Unauthorized

**}**

**1.3 Endpoint** /logout

**Method** POST

**Description** : destroy the user session

**Respond Body:**

**{**

// 201 success

Message **: “**Log out successfully**”**

**}**

**//** failed

**{**

Error : 401 Unauthorized // when the session is expired

**}**

**1.4 Endpoint** /posts

**Method:** GET

**Description** : Show all posts

**Respond Body:**

**{**

// 201 success

Message **: “** null**”**

**}**

**//** failed

**{**

Error : 401 Unauthorized // when the session is expired

**}**

**1.5 Endpoint** /posts/:postId/comments

**Method:** POST

**Description** : write comment

**Request Body:**

**{**

Text: “”

**}**

**Respond Body:**

**{**

// 201 success

Message **: “** successfully add comment**”**

**}**

**//** failed

**{**

Error : 401 Unauthorized // when the session is expired

**}**

**1.6 Endpoint** /user/posts

**Method:** GET

**Description** : display all current posts of user

**Respond Body:**

**{**

// 201 success

Message **: “**null**”**

**}**

**//** failed

**{**

Error : 401 Unauthorized // when the session is expired

**}**

**1.7 Endpoint** /user/posts

**Method:** POST

**Description** : Create new post

**Request Body:**

**{**

Title : “”,

Description: “”,

Image : “”,

**}**

**Respond Body:**

**{**

// 201 success

Message **: “**Created seccessfully**”**

**}**

**//** failed

**{**

Error : failure to add post

**}**

**1.8 Endpoint** /user/posts/:postId

**Method:** PUT

**Description** : Update post

**Request Body:**

**{**

Title : “”,

Description: “”,

Image : “”,

**}**

**Respond Body:**

**{**

// 201 success

Message **: “**Updated seccessfully**”**

**}**

**//** failed

**{**

Error : failure to update post

**}**

**1.9 Endpoint** /user/posts/posts:id

**Method:** DELETE

**Description** : Delete post

**Request Body:**

**{**

Post: objectID

**}**

**Respond Body:**

**{**

// 201 success

Message **: “**Deleted seccessfully**”**

**}**

**//** failed

**{**

Error : failure to Delete post

**}**

**1.10 Endpoint** /admin/users

**Method:** GET

**Description** : Show all users

**Respond Body:**

**{**

// 201 success

Message **: “”**

**}**

**//** failed

**{**

**}**

**1.11 Endpoint** /admin/users/:userId/block

**Method:** PUT

**Description** : Block user

**Request Body:**

**{**

User: objectID

**}**

**Respond Body:**

**{**

// 201 success

Message **: “**Blocked seccessfully**”**

**}**

**//** failed

**{**

**// 404**

Error : User not found

**}**

**1.12 Endpoint** /admin/users/:userId/unblock

**Method:** PUT

**Description** : unBlock user

**Request Body:**

**{**

User: objectID

**}**

**Respond Body:**

**{**

// 201 success

Message **: “**unBlocked seccessfully**”**

**}**

**//** failed

**{**

**// 404**

Error : User not found

**}**

**1.13 Endpoint** /admin/users/:userId/posts

**Method:** GET

**Description** : Show all post of user

**Respond Body:**

**{**

// 201 success

Message **: “”**

**}**

**//** failed

**{**

**// 404**

Error : User not found

**}**

**1.14 Endpoint** /admin/posts/:postId

**Method:** DELETE

**Description** : Admin delete any post

**Request Body:**

**{**

Post : objectID

**}**

**Respond Body:**

**{**

// 201 success

Message **: “**Deleted successfully**”**

**}**

**//** failed

**{**

**// 404**

Error : POST not found

**}**