Social-Media

User:

- Register:

Method: POST,
Path: /users/register
Request:
Body { login, password, email, firstName, secondName?, lastName }
Response:

Status: 201 { message: „Added”, status: 201 },
Status: 400 { message: „Wrong data”, status: 400 },
Status: 409 { message: „User exist”, status: 409 }

- Login:

Method: POST,
Path: /users/login
Request:
Body { login, password }
Response:

Status: 200 { message: „Success”, status: 200, token },
Status: 400 { message: „Body is empty”, status: 400 },
Status: 406 { message: „Login or password is wrong”, status: 406 }

- Logout:

Method: GET,
Path: /users/logout
Request:
Headers { token }
Response:

Status: 200 { message: „Success log out”, status: 200 },
Status: 400 { message: „Token does not exist”, status: 400 },
Status: 406 { message: „You no longer log in”, status: 406 }

- Get user info:

Method: GET,
Path: /users
Request:

Headers { token }

Response:

Status: 200 { message: „User info” status: 200, user },
Status: 400 { message: „Token does not exist”, status: 400 },
Status: 401 { message: „You must be log in”, status: 401 }
Status: 403 { message: „Verify account”, status: 403 },
Status: 404 { message: „User does not exist”, status: 404 }

- Get verify code:

Method: GET,
Path: /users/verify
Request:
Headers { token }

Response:

Status: 200 { message: „Code sent”, status: 200 },
Status: 400 { message: „Token does not exist”, status: 400 },
Status: 401 { message: „You must be log in”, status: 401 }

- Verify user using code:

Method: PATCH,
Path: /users/verify
Request:
Body { code }
Response:

Status: 200 { message: „Account verified”, status: 200 },
Status: 400 { message: „Code is empty”, status: 400 },
Status: 409 { message: „Cod eis not correct”, status: 409 }

- Delete account:

Method: DELETE,
Path: /users
Request:
Headers { token }
Response:

Status: 200 { message: „Account deleted", status: 200 },
Status: 400 { message: „Token does not exist", status: 400 },
Status: 401 { message: „You must be log in", status: 401 },
Status: 403 { message: „Verify account", status: 403 },
Status: 404 { message: „User does not exist", status: 404 }

- Change password:

Method: PATCH,
Path: /users/password
Request:
Body { password, newPassword}
Headers { token }

Response:
Status: 200 { message: „Password changed", status: 200 },
Status: 400 { message: „Password field is empty", status: 400 },
Status: 400 { message: „Token does not exist", status: 400 },
Status: 401 { message: „You must be log in", status: 401 },
Status: 403 { message: „Verify account", status: 403 },
Status: 404 { message: „User does not exist", status: 404 }

- Change email:

Method: PATCH,
Path: /users/email
Request:
Body { email }
Headers { token }
Response:

Status: 200 { message: „Email updated", status: 200 },
Status: 400 { message: „Email field is empty", status: 400 },
Status: 400 { message: „Token does not exist", status: 400 },
Status: 401 { message: „You must be log in", status: 401 }
Status: 403 { message: „Verify account", status: 403 },
Status: 404 { message: „User does not exist", status: 404 }

- Get recover code:

Method: GET,
Path: /users/recover
Request:
    Body: { login }
Response:

    Status: 200 { message: „Code sent”, status: 200 },
    Status: 400 { message: „Login field is empty”, status: 400 },

- Chance password using recover code:

Method: PATCH,
Path: /users/recover
Request:
    Body: { password, code }
Response:

    Status: 200 { message: „Password changed”, status: 200 },
    Status: 400 { message: „Password field is empty”, status: 400 },
    Status: 400 { message: „Code is empty”, status: 400 },
    Status: 409 { message: „Code is not correct”, status: 409 }

Friends:

- Get friends list:

Method: GET,
Path: /friends
Request:
Headers { token }
Response:
Status: 200 { message: „Friend list”, status: 200, friendList },
Status: 400 { message: „Token does not exist”, status: 400 },
Status: 401 { message: „You must be log in”, status: 401 },
Status: 403 { message: „Verify account”, status: 403 },
Status: 404 { message: „User does not exist”, status: 404 }

- Get all sent friend request:

Method: GET,
Path: /friends/request/sent
Request:
Headers { token }
Response:
Status: 200 { message: „Friend request - sent”, status: 200, friendList },
Status: 400 { message: „Token does not exist”, status: 400 },
Status: 401 { message: „You must be log in”, status: 401 },
Status: 403 { message: „Verify account”, status: 403 },
Status: 404 { message: „User does not exist”, status: 404 }

- Get all friend request to accept:

Method: GET,
Path: /friends/request/pending
Request:
Headers { token }
Response:
Status: 200 { message: „Friend request - pending”, status: 200, friendList },
Status: 400 { message: „Token does not exist”, status: 400 },
Status: 401 { message: „You must be log in”, status: 401 }
Status: 403 { message: „Verify account”, status: 403 },
Status: 404 { message: „User does not exist”, status: 404 }

- Send friend request:

Method: POST,
Path: /friends
Request:
Headers { token }
Body { login }
Response:
Status: 200 { message: „Friend request accepted”, status: 200 },
Status: 201 { message: „Friend request sent”, status: 201 },
Status: 400 { message: „Friend’s login field is empty”, status: 400 },
Status: 400 { message: „Token does not exist”, status: 400 },
Status: 401 { message: „You must be log in”, status: 401 },
Status: 403 { message: „Verify account”, status: 403 },
Status: 404 { message: „User does not exist”, status: 404 },
Status: 404 { message: „User does not exist”, status: 404 },
Status: 406 { message: „Request already sent”, status: 406 },
Status: 409 { message: „You cannot invite yourself”, status: 409 }

- Accept friend request:

Method: PATCH,
Path: /friends/request
Request:
Headers { token }
Body { login }
Response:
Status: 200 { message: „Friend request accepted”, status: 200 },
Status: 400 { message: „Login field is empty”, status: 400 },
Status: 400 { message: „Token does not exist”, status: 400 },
Status: 401 { message: „You must be log in”, status: 401 },
Status: 403 { message: „Verify account”, status: 403 },
Status: 404 { message: „User does not exist”, status: 404 },
Status: 404 { message: „Request does not exist”, status: 404 }

- Decline friend request:

Method: DELETE,
Path: /friends/request
Request:
    Headers { token }
    Body { login }
Response:
    Status: 200 { message: „Friend request declined”, status: 200 },
    Status: 400 { message: „Login field is empty”, status: 400 },
    Status: 400 { message: „Token does not exist”, status: 400 },
    Status: 401 { message: „You must be log in”, status: 401 },
    Status: 403 { message: „Verify account”, status: 403 },
    Status: 404 { message: „User does not exist”, status: 404 }

- Delete friend:

Method: DELETE,
Path: /friends
Request:
    Headers { token }
    Body { login }
Response:
    Status: 200 { message: „Friend deleted”, status: 200 },
    Status: 400 { message: „Login field is empty”, status: 400 },
    Status: 400 { message: „Token does not exist”, status: 400 },
    Status: 401 { message: „You must be log in”, status: 401 },
    Status: 403 { message: „Verify account”, status: 403 },
    Status: 404 { message: „User does not exist”, status: 404 }

Posts:

- Get all user posts:

Method: GET,
Path: /posts
Request:
Headers { token }
Response:
Status: 200 { message: „All user’s posts”, status: 200, posts },
Status: 400 { message: „Token does not exist”, status: 400 },
Status: 401 { message: „You must be log in”, status: 401 },
Status: 403 { message: „Verify account”, status: 403 },
Status: 404 { message: „User does not exist”, status: 404 }

- Get one user post:

Method: GET,
Path: /posts/id/:id
Request:
Headers { token }
Response:
Status: 200 { message: „One user’s post”, status: 200, post },
Status: 400 { message: „Token does not exist”, status: 400 },
Status: 401 { message: „You must be log in”, status: 401 },
Status: 403 { message: „Verify account”, status: 403 },
Status: 404 { message: „User does not exist”, status: 404 }

- Get other user’s post:

Method: GET,
Path: /posts/user/:login
Request:
Headers { token }
Response:
Status: 200 { message: „User’s posts”, status: 200, posts },
Status: 400 { message: „Token does not exist”, status: 400 },
Status: 401 { message: „You must be log in”, status: 401 },
Status: 403 { message: „Verify account”, status: 403 },
Status: 404 { message: „User does not exist”, status: 404 }

- Create new post:

Method: POST,
Path: /posts
Request:
Headers { token },
Body { isPrivate, content }
Response:
Status: 201 { message: „Post created”, status: 201 },
Status: 400 { message: „Content or private field is empty”, status: 400 },
Status: 400 { message: „Token does not exist”, status: 400 },
Status: 401 { message: „You must be log in”, status: 401 },
Status: 403 { message: „Verify account”, status: 403 },
Status: 404 { message: „User does not exist”, status: 404 }

- Modify post:

Method: PATCH,
Path: /posts
Request:
Headers { token }
Body { id, content }
Response:
Status: 200 { message: „Post modified”, status: 200 },
Status: 400 { message: „Content or id field is empty”, status: 400 },
Status: 400 { message: „Token does not exist”, status: 400 },
Status: 401 { message: „You must be log in”, status: 401 },
Status: 403 { message: „Verify account”, status: 403 },
Status: 404 { message: „User does not exist”, status: 404 }

- Delete post:

Method: DELETE,
Path: /friends
Request:
Headers { token }
Body { id }
Response:
Status: 200 { message: „Post deleted”, status: 200 },
Status: 400 { message: „Id field is empty”, status: 400 },
Status: 400 { message: „Token does not exist”, status: 400 },
Status: 401 { message: „You must be log in”, status: 401 },
Status: 403 { message: „Verify account”, status: 403 },
Status: 404 { message: „User does not exist”, status: 404 }

Comments:

- Get all user’s comments:

Method: GET,
Path: /comments
Request:
  Headers { token }
Response:
  Status: 200 { message: „Comments”, status: 200, comments },
  Status: 400 { message: „Token does not exist”, status: 400 },
  Status: 401 { message: „You must be log in”, status: 401 },
  Status: 403 { message: „Verify account”, status: 403 },
  Status: 404 { message: „User does not exist”, status: 404 }

- Get all post’s comments:

Method: GET,
Path: /comments/post/:id
Request:

Response:
  Status: 200 { message: „Post’s comments”, status: 200, comments }

- Create new post:

Method: POST,
Path: /comments
Request:
  Headers { token }
  Body { postId, content }
Response:
  Status: 201 { message: „Comment added”, status: 201 },
  Status: 400 { message: „Post ID or content field is empty”, status: 400 },
  Status: 400 { message: „Token does not exist”, status: 400 },
  Status: 401 { message: „You must be log in”, status: 401 },
  Status: 403 { message: „Verify account”, status: 403 },
  Status: 404 { message: „User does not exist”, status: 404 },
  Status: 404 { message: „Post does not exist”, status: 404 }

- Edit comment:

Method: PATCH,
Path: /comments
Request:
  Headers { token }
  Body { id, content }
Response:
  Status: 200 { message: „Comment edited”, status: 200 },
  Status: 400 { message: „ID or content field is empty”, status: 400 },
  Status: 400 { message: „Token does not exist”, status: 400 },
  Status: 401 { message: „You must be log in”, status: 401 },
  Status: 403 { message: „Verify account”, status: 403 },
  Status: 404 { message: „User does not exist”, status: 404 }

- Delete comment:

Method: DELETE,
Path: /comments
Request:
  Headers { token }
  Body { id }
Response:
  Status: 200 { message: „Comment deleted”, status: 200 },
  Status: 400 { message: „ID field is empty”, status: 400 },
  Status: 400 { message: „Token does not exist”, status: 400 },
  Status: 401 { message: „You must be log in”, status: 401 },
  Status: 403 { message: „Verify account”, status: 403 },
  Status: 404 { message: „User does not exist”, status: 404 }