

TEST CASES → API

- **TEST CASE No.1:** User Sign in [SMOKE]

Description: Verify that a new user can sign up successfully.

Endpoint: POST /addUser

Preconditions: None

	Steps	Expected Result
1.	Send a POST request with valid user details (first name, last name, email, password).	Response: 201 Created, body contains user details (excluding password).
2.	Send a POST request with missing required fields (e.g., last name).	Response: 400 Bad Request, error message: "lastName and password are required."
3.	Send a POST request with a password shorter than the required length.	Response: 422 Unprocessable Entity. Error message: "Password must be at least 7 characters long."
4.	Send a POST request with an invalid email format	Response: 422 Unprocessable Entity. Error message: "Invalid email format."
5.	Attempt to register with an already-used email address.	Response: 409 Conflict. Error message: "Email already in use."

- **TEST CASE No.2:** User Log in [SMOKE]

Description: Verify that a new user can log in successfully.

Endpoint: POST /

Preconditions: User is already registered.

	Steps	Expected Result
1.	Send a POST request with valid credentials (email and password).	Response: 200 OK. Response contains token and user details
2.	Send a POST request with an incorrect password.	Response: 401 Unauthorized. Error message: "Incorrect username or password."
3.	Send a POST request with an unregistered email.	Response: 401 Unauthorized, error message: "Incorrect username or password."

- **TEST CASE No.3:** Fetch Contact List [SMOKE]

Description: Verify that a user can fetch their list of contacts.

Endpoint: GET /contactList

Preconditions: User is logged in and has a valid JWT token.

	Steps	Expected Result
1.	Send a GET request with a valid authentication token.	Response: 200 OK, body contains a list of contacts.
2.	Send a GET request without an authentication token.	Response: 401 Unauthorized. Error message: "Token is missing or invalid."
3.	Send a GET request with an invalid token.	Response: 403 Forbidden, error message: "Invalid token."

- **TEST CASE No.4:** Add a new contact

Description: Verify that a user can add a new contact to their list.

Endpoint: POST /addContact

Preconditions: User is logged in and has a valid JWT token.

	Steps	Expected Result
1.	Send a POST request with valid data.	Response: 201 Created, body contains the new contact's details.
2.	Send a POST request with missing required fields (e.g., first name).	Response: 400 Bad Request, error message: "firstName is required."
3.	Send a POST request with invalid email.	Response: 400 Bad Request, error message: "Email is invalid."

- **TEST CASE No.5: Update a contact**

Description: Verify that a user can update an existing contact's details.

Endpoint: PUT /editContact

Preconditions: User is logged in, and a contact exists in the list.

	Steps	Expected Result
1.	Send a PUT request with valid updated contact details	Response: 200 OK, body contains the updated contact's details.
2.	Send a PUT request for a non-existent contact ID	Response: 404 Not Found, error message: "Contact not found."
3.	Send a PUT request with invalid data formats (e.g., invalid email).	Response: 400 Bad Request, error message: "Email is invalid."

- **TEST CASE No.6: Delete a contact**

Description: Verify that a user can delete a contact.

Endpoint: DELETE /contactDetails

Preconditions: User is logged in, and a contact exists in the list.

	Steps	Expected Result
1.	Send a DELETE request for an existing contact ID.	Response: 200 OK, body confirms deletion
2.	Send a DELETE request for a non-existent contact ID	Response: 404 Not Found, error message: "Contact not found."
3.	Send a DELETE request for a contact owned by another user.	Response: 403 Forbidden, error message: "Access denied."

- **TEST CASE No.6: Log out [SMOKE]**

Description: Validate user logout functionality.

Endpoint: POST /logout

Preconditions: User is logged in.

	Steps	Expected Result
1.	Send a POST request with a valid authentication token	Response: 200 OK. Session is terminated.
2.	Send a POST request with an invalid or expired token.	Response: 401 Unauthorized. Error message: "Invalid token."
3.	Send a POST request without token	Expected Result: 400 Bad Request. Error message: "Token is required."