Test plan

“Contact List App”

Judah Sleibi
Maisam Salameh
Tasneim Othman

Introduction:

The following test plan is designed for the *Contact List App,* the aim of testing to verify the API state connecting the UI of the app with the database that is related to the web. The testing in general way will cover the following cases, the state of the API is it user friendly, is the API secure enough Checks for authentication and authorization as well.

The URL of the site:
https://thinking-tester-contact-list.herokuapp.com/

Contents

Introduction: 2

Scope: 4

Out of the scope: 4

Environment: 4

Timelines 4

DOD: 4

Methodology: 4

- Functional test: 5
- Non-functional test: 5

Test responsibilities: 5

- Developers: 5
- QA testers: 5

Business requirements: 5

Scope:

1. Verifies the ability of the user to open the URL.
2. Verifies the ability to add and delete contact.
3. Verifies the ability to retrieve contact and its list.
4. Verifies the ability to update contact.
5. Verifies the ability to add new users.
6. Verifies the ability to retrieve user profile.
7. Verifies the ability to update specific entry in the user profile.
8. Verifies the ability of the users to log in and out from their accounts.
9. Verifies the need of token (authorization) for each operation and the ability of the UI to provide token for the user.
10. Verifies the ability of the system to rejects non valid operations and missing authorization.

Out of the scope:

1. The type of the database.
2. How the data stored in the database.

Environment:

- Windows 11 pro.
- Google chrome version131.0.6778.109.
- Testing environment.
- Production environment.
- Postman.

Timelines:

Test	Scheduled time
User test suits	9-12-2024 ➔ 12-12-224
Contact test suits	9-12-2024 ➔ 12-12-224

DOD:

1. If the requirements of the business team are applied and the workflow of the UI matches the requirements.
2. If the founded bugs are low in severity and accepted by the project manager.

Methodology:

The agile methodology will be used through the project, as well as the process will be tracked through Trello boards.

Functional test:

1. Black box test:

Ensuring every feature in the API works as intended.

2. Smoke test:

Quik general test to confirm that all features work as intended.

Non-functional test:

1. Load test:

Check the API ability to handle particular workload at the sametime.

2. Performance testing:

Check the overall performance of the API, in matters of request timing and state.

3. Stress testing:

Check the ability to the system to withstand high number of requests without crashing or lagging.

4. Security test:

Check the authentication and authorization of the API.

Test responsibilities:

Developers:

- Design codes to satisfy the needs of the business team and the stack holder.
- Fix and handle bugs.

QA testers:

- Ensure the produced applications are designed based on business team
- requirements.
- Ensure the highest number of bugs are detected.
- Verifies and validates the produced products.

Business requirements:

1. Verifies the ability to add new contact.
2. Verifies the ability to retrieve contact list.
3. Verifies the ability to retrieve contact.
4. Verifies the ability to update contact.
5. Verifies the ability to update specific entry in the contact information.
6. Verifies the ability to delete contact.
7. Verifies the ability to add new users.
8. Verifies the ability to retrieve user profile.
9. Verifies the ability to update specific entry in the user profile.
10. Verifies the ability of the users to log out from their accounts.

11. Verifies the ability of the users to log into their accounts.
12. Verifies the ability to delete users.
13. Verifies the ability to delete contact.
14. Verifies the ability to reject non-valid data.
15. Verify the need of token (authorization) for each operation.
16. Verifies the ability of the system to reject operations with missing authorization.
17. Verifies the ability of the API to provide token for the user.
18. Verifies the ability of the system to reject non-valid operations, fake account logout.
19. Authorization is gained by the Bearer token.