

















18.4 Test classification: Levels of testing: Homework

To the functionality, that you covered with test cases for previous work:

Add 2 integration test cases

Add 1 system test case

Add 1 acceptance test case (choose the most crucial business scenario to cover)

<input type="checkbox"/>	ID ↓	Title	Test Status	Created At	Actions
<input type="checkbox"/>	  M20-TC1715	Integration. API+DB+UI. Creating a New User - Agent (with correct details).	In Progress	January 19, 2024	 
<input type="checkbox"/>	  M20-TC1717	Integration. API+DB+UI. Delete User.	In Progress	January 19, 2024	 
<input type="checkbox"/>	  M20-TC1720	System Test. User Agent has access to all functionality on the UI.	In Progress	January 20, 2024	 
<input type="checkbox"/>	  M20-TC1722	Acceptance. User Agent successfully Booking a flight for a new passenger on the UI	In Progress	January 20, 2024	 

Integration tests:

M20-TC1715 - V. 1 - Integration. API+DB+UI. Creating a New User - Agent (with correct details).

DETAILS

TEST SCRIPT

EXECUTION

SUB TASKS

ATTACHMENTS

Title*

Integration. API+DB+UI. Creating a New User - Agent (with correct details).

Test Objectives

To check Integration of API+DB+UI after API POST method creating a new user (with correct details) with role = agent.

To check integration of 3 levels:
To check that a new user with role agent created/exists when using API GET method.
To check that a new user with role agent created/exists in the DB.
To check that a new user with role agent created/exists in the UI.

Precondition(s)

Table users exists in a system.
Table roles exists in a system.
Role agent exists in a system.
Table users_roles exists in a system.
To be logged in with correct admin credentials.

Postcondition(s)

Delete created user.

Additional Informations

Test Status*

In Progress

Priority

Click to add a value

Responsible - Owner

Click to add a value

Created At

January 19, 2024 5:21 PM

Category: Steps

Steps

#	Test Instructions	Test Data	Expected Results
1	Run POST method (POST /api/v0/users) to create a new user with entered correct credentials.	Request body: <pre>{ "firstname": "Vivien", "middlename": "Sun", "lastname": "Backham", "username": "vivi", "email": "vivi@mail.com", "password": "vivi1234", "roleNames": ["ROLE_AGENT"] }</pre>	Get response: 201 Created. Response body: <pre>{ "id": 5, "firstname": "Vivien", "middlename": "Sun", "lastname": "Backham", "username": "vivi", "email": "vivi@mail.com", "password": "vivi1234", "roleNames": ["ROLE_AGENT"] }</pre>
2	Run GET method (GET /api/v0/users/{id}) to check that a user created.	id = 5 of a user from first step.	Get response: 200 OK Response body: <pre>{ "id": 5, "firstname": "Vivien", "middlename": "Sun", "lastname": "Backham", "username": "vivi", "email": "vivi@mail.com", "password": "vivi1234", "roleNames": ["ROLE_AGENT"] }</pre>
3	Check that a new user with correct role exists in the DB. All personal data of a user entered in 1 step and his role are corrected in DB.	<pre>select * from users u JOIN users_roles ur where u.id = ur.user_id having id = 5</pre>	A new user with role agent created in a DB. All personal data are correct. Role is correct.
4	Check that a new user with correct role can enter in a system from the UI and has rights according to his role (agent).	Use username created in first step. Use password created in first step.	On the UI a new user is able to enter into a system with his credentials. Righths of a user on the UI is according to his role agent.

M20-TC1717 - V. 1 - Integration. API+DB+UI. Delete User.

M20-TC1717 - V. 1 - Integration. API+DB+UI. Delete User.

3/4

DETAILS

TEST SCRIPT

EXECUTION

SUB TASKS

ATTACHMENTS

Title*

Integration. API+DB+UI. Delete User.

Test Status*

In Progress

Test Objectives

To check Integration of API+DB+UI after API DELETE method deleting user from a DB .

To check integration of 3 levels.

To check that a user deleted when using API.

To check that a user deleted from the DB.

To check that a user deleted from the UI.

Precondition(s)

To be logged in with correct admin credentials.

To have at least 2 users in a system (admin + user for test).

Check that a user for test exists in the DB, UI and API.

Postcondition(s)

Restore deleted user.

Additional Informations

Click to add a value

Priority

Click to add a value

Responsible - Owner

Click to add a value

Created At

January 19, 2024 6:31 PM

DETAILS

TEST SCRIPT

EXECUTION

SUB TASKS

ATTACHMENTS

Category: Steps

Steps

#	Test Instructions	Test Data	Expected Results
1	Run DELETE method: DELETE /api/v0/users/{id}	Requested details: id of a chosen for test user.	Response code 200, OK.
2	Run GET method (GET /api/v0/users/{id}) to check that a user deleted /not found by API.	id of a chosen for test user.	Get response: 404 Not Found. <div>Response body: { "timestamp": "YYYY-MM-DD HH:MM:SS", "status": "404 NOT_FOUND", "message": "Entity not found.", "details": ["User User was not found by id=5"] }</div>
3	Check that a user deleted from the DB.	<div>select * from users where id = 5</div>	User not found in the DB
4	Check that a user deleted from the UI.	Use username and password of a chosen for test user.	On the UI user is not able to enter into a system with his credentials. Get error message 'Invalid username and/or password.'

System test case:

M20-TC1720 - V. 1 - System Test. User Agent has access to all functionality on the UI.

M20-TC1720 - V. 1 - System Test. User Agent has access to all functionality on the UI.

2 / 4

DETAILS

TEST SCRIPT

EXECUTION

SUB TASKS

ATTACHMENTS

Title*

System Test. User Agent has access to all functionality on the UI.

Test Status*

In Progress

Test Objectives

To check that a user with role Agent has access to all described in documentation functionality on the UI.

Precondition(s)

There is at least 1 user with role agent.
There are at lest 2 airports in a system
There is at least 1 flight in a system
There is at least 1 passenger in a system.

Postcondition(s)

Click to add a value

Additional Informations

Click to add a value

Priority

Click to add a value

Responsible - Owner

Click to add a value

Created At

January 20, 2024 2:03 PM

DETAILS

TEST SCRIPT

EXECUTION

SUB TASKS

ATTACHMENTS

Category: Steps

Steps

#	Test Instructions	Test Data	Expected Results	
1	Log in on the UI with correct agent credential.	Click to add text	User (agent) successfully logged in. Opened the main page of FTB application.	+ ...
2	Check that all tabs in main menu presented according to role 'agent'	Click to add text	All tabs in main menu presented according to role 'Agent': Home Flights Book flight Search flight Verify booking	+ ...
3	Click on Tab 'Home' in Main menu.	Click to add text	Stayed opened the same main page.	+ ...
4	Click on Tab 'Flights' in Main menu. Check that information about flights available in the page is for 'reading only'. Ckeck that available button 'Aircraft'.	Click to add text	Opened a page 'Flights list'. Information available for 'reading only'. Available button 'Aircraft'.	+ ...

5	<p>Click on button 'Aircraft'. Check that opened a page 'Search a Flight'.</p> <p><i>Click to add text</i></p>	<p>Opened a page 'Search a Flight'. In the page presented all fields for searching a flight: From, To, Departure Time. Available button 'search'.</p>	<p>+</p> <p>...</p>
6	<p>Click on Tab 'Book flight' in Main menu.</p> <p>Check that there are 3 field for searching a flight for booking: From, To, Departure Time</p> <p>Available button 'Search'.</p> <p><i>Click to add text</i></p>	<p>Opened a page 'Book a Flight'. In the page presented all fields.</p> <p>Available button 'Search'.</p>	<p>+</p> <p>...</p>
7	<p>Enter details of Departure and Destination airport, departure time. Click on button 'Search'.</p> <p><i>Click to add text</i></p>	<p>Found flight / flights according to search parameters. Shown 'Result'. Available button 'Book'.</p>	<p>+</p> <p>...</p>
8	<p>Click 'Book' button. Check that 'Passenger Information' page is available and opened. Check that available all fields on this page.</p> <p><i>Click to add text</i></p>	<p>Opened 'Passenger Information' page. Fields: First Name, Last Name, Phone number, Passport Nimber, Email address, Address are presented. Available button 'Save'.</p>	<p>+</p> <p>...</p>
9	<p>Click on Tab 'Search flight' in Main Menu. Check that there are 3 field for searching a flight: From, To, Departure Time. Available button 'Search'.</p> <p><i>Click to add text</i></p>	<p>Opened a page 'Search a Flight'. In the page presented all fields.</p> <p>Available button 'Search'.</p>	<p>+</p> <p>...</p>
10	<p>Click on Tab 'Verify booking' in Main menu. Check that there are 2 field for verifying a booking: Flight ID, Passenger ID, Available button 'Search'.</p> <p><i>Click to add text</i></p>	<p>Opened a page 'Verify booking'. In the page presented all fields.</p> <p>Available button 'Search'.</p>	<p>+</p> <p>...</p>

Acceptance test case:

M20-TC1722 - V. 1 - Acceptance. User Agent successfully Booking a flight for a new passenger on the UI

M20-TC1722 - V. 1 - Acceptance. User Agent successfully Booking a flight for a new passenger on the UI

DETAILS

TEST SCRIPT

EXECUTION

SUB TASKS

ATTACHMENTS

Title*

Acceptance. User Agent successfully Booking a flight for a new passenger on the UI

Test Objectives

To check that user with role Agent can successfully book a flight on the UI for a new passenger.

Precondition(s)

There is at least 1 user with role agent.
There are at lest 2 airports in a system
There is at least 1 flight in a system.

To be logged in with correct Agent credentials.
Use for test data from excel (should be stored in companies data storage) (for example attached to this test case).

Postcondition(s)

Cancel created booking.

Additional Informations

Click to add a value

Test Status*

In Progress

Priority

Click to add a value

Responsible - Owner

Click to add a value

Created At

January 20, 2024 3:52 PM

DETAILS

TEST SCRIPT

EXECUTION

SUB TASKS

ATTACHMENTS

Category: Steps

Steps

#	Test Instructions	Test Data	Expected Results
1	Click on Tab 'Book flight' in Main menu.	Click to add text	Opened a page 'Book a Flight'. Shown fields for details of Departure and Destination airports, departure time.

<div>⋮</div> <div>2</div>	<p>Enter details of Departure and Destination airports, departure time. Click on button 'Search'.</p>	<p>From: Helsinki-Vantaa Airport To: Liverpool John Lennon Airport Departure Time: 27/12/2023</p>	<p>Found a flight according to search parameters. Shown 'Result':</p> <p>ID: Flight Number: From: To: Departure: YYYY-MM-DD HH:MM Arrival: YYYY-MM-DD HH:MM Flight charge: Aircraft: (fields described above filled according to presented in file information). Available Seats: (filled according to DB calculations).</p> <p>Available button 'Book'.</p> <div>+</div> <div>...</div>
<div>⋮</div> <div>3</div>	<p>Click 'Book' button.</p>	<p>Click to add text</p>	<p>Opened 'Passenger Information' page. Fields: First Name, Last Name, Phone number, Passport Number, Email address, Address are presented.</p> <p>Available button 'Save'.</p> <div>+</div> <div>...</div>
<div>⋮</div> <div>4</div>	<p>Enter correct details of a new passenger. Click on button 'Save'.</p>	<p>Click to add text</p>	<p>Opened 'Confirmation Page'. Displayed information about Flight ID (the same as in step 2), Passenger ID (new data).</p> <p>Displayed Detailed information about a new passenger (Customer info). Details are filled according to entered data (presented in file information).</p> <p>Displayed Detailed information about a flight (Flight info). Details are displayed according to entered data (presented in file information.)</p> <div>+</div> <div>...</div>