Since I was the only one who wanted to practice writing E2E test cases in my team, I can present the complete assignment.

Team 6 3. End-to-end test cases

Gätlin Nurk

- Managing notes 1
  - MN01: Create a new deal via API - minimum fields
  - MN02: Deal created via API is visible in the UI
  - MN03: Create a new note via API
  - MN04: Link a note to the deal via API
  - MN05: Note linked to the deal via API is visible in the UI
  - MN06: Create a new note via UI
  - MN07: Newly created note via UI is visible in the API
  - MN08: Delete the first note via API
  - MN09: Deleted note is no longer visible in the UI
- Managing notes 2
  - MN02.01: Create a new funnel via UI
  - MN02.02: Newly created funnel via UI is visible in the API
  - MN02.03: Create a new deal with the created funnel via API
  - MN02.04: Deal created via API is visible in the UI
  - MN02.05: Create a new note via UI
  - MN02.06: Newly created note via UI is visible in the API
  - MN02.07: Modify the note via API
  - MN02.08: Modified note is seen with updated information in UI

Managing notes 1

Description of the user flow: New deal is created via API with minimum fields set. After that a new note is created via API which will be linked to the previously created new deal. Then another note will be made via UI to the created deal. First note is no longer needed and therefore deleted.

Test case ID	MN01: Create a new deal via API - minimum fields
Description	Check that it is possible to create a deal with only mandatory fields.
Precondition	User is authorized; Valid Ids are present in the system.

Test data	{
	  "assignee__user": "1625047373081x778685405315052900",
	  "company_name_text": "Team 6 E2E mandatory fields",
	  "deal_value_estimation_number": 720000,
	  "name_text": "E2E.1",
	  "order_number": 3,
	  "status_option_status": "In progress",
	  "visible_to_list_user": [
	    "1625047373081x778685405315052900"
	  ]
	}

Test steps	1. Send **POST**: `{{baseUrl}}/obj/deal` request

Expected result	• Successful code: 201 Created
	• A new deal is created with only mandatory fields filled:
	
	{
	   "status": "success",
	   "id": "1726060489741x475251642694628740"
	}

Result	Passed

Test case ID	**MN02: Deal created via API is visible in the UI**
Description	Check that the new deal which was created in MN01 is visible in the CRM system.
Precondition	MN01 successfully completed.
Test data	-
Test steps	1. Open Deals page 2. Click on: "Switch to List view" left to the: "New Deal" button 3. In the: "Deal" search bar insert "E2E", other fields leave empty 4. Once: "E2E.1" gets highlighted in UI, click on it 5. Press the pencil icon to enter the: "Edit" mode
Expected result	Step 1: Deals page is opened Step 2: Different search fields are presented Step 5: Deal's information in the UI is the same as the entered information in MN01
Result	Failed, <u>Link to the bug report</u>

Test case ID	**MN03: Create a new note via API**
Description	Check that it is possible to create a new note
Precondition	User is authorized.

Test data	```
	{
	  "content_text": "Team6 E2E note to be deleted after"
	}
	``

Test steps	1. Send **POST**: {{baseUrl}}/obj/note request

Expected result	• Successful code: 201 Created
	• A new note is created without any errors:
	
	``
	{
	   "status": "success",
	   "id": "1726064383927x387254416568281400"
	}
	``

Result	Passed

Test case ID	**MN04: Link a note to the deal via API**
Description	Check that it is possible to link a note from MN03 to the deal made in MN01
Precondition	User is authorized; MN03 successfully completed.

Test data	```
	{
	        "assignee__user": "1625047373081x778685405315052900",
	        "company_name_text": "Team 6 E2E mandatory fields",
	        "deal_value_estimation_number": 720000,
	        "name_text": "E2E.1",
	        "visible_to_list_user": [
	            "1625047373081x778685405315052900"
	        ],
	        "status_option_status": "Won",
	        "note_list_custom_note": [
	        "1726064383927x387254416568281400"
	    ],
	        "order_number": 3
	    }
	``

Test steps	1. Send **PATCH**: `{{baseUrl}}/obj/deal/:UniqueID` request, add correct UniqueId (taken from MN01: "Expected result") and correct "note_list_custom_note" id (taken from MN03: "Expected result")
Expected result	• Successful code: 204 No Content • Adding a note was successful
Result	Passed

Test case ID	**MN05: Note linked to the deal via API is visible in the UI**
Description	Check that the note that was linked in MN04 is visible in the UI
Precondition	MN04 successfully completed. Deal is opened from MN02.
Test data	-
Test steps	1. Scroll down on the opened deal from MN02 until you see notes, refresh if needed
Expected result	• Note is visible in the UI
Result	Passed

Test case ID	**MN06: Create a new note via UI**
Description	Check that it is possible to create a note via CRM app.
Precondition	MN05 deal is opened.
Test data	Note: "Team6 - this note will not be deleted"
Test steps	1. Scroll down on the opened deal from MN05 until you see: "Type your note here…" 2. Write a note and click on: "Add" on the right side
Expected result	• A note is successfully added
Result	Passed

Test case ID	MN07: Newly created note via UI is visible in the API
Description	Check that the new note which was created via UI in MN06 is visible in the API.
Precondition	User is authorized; MN06 successfully completed.
Test data	-
Test steps	1. Send **GET**: `{{baseUrl}}/obj/deal/:UniqueID` request, input correct UniqueID (taken from MN01: "Expected result")
Expected result	• Successful code: 200 OK • Note created in MN06 is visible, "note_list_custom_note" has two notes attached: • `"note_list_custom_note": [` • `"1726064383927x387254416568281400",` • `"1726065724405x206636106405904400"` `]`
Result	Passed

Test case ID	MN08: Delete the first note via API
Description	Check that it is possible to delete a note linked in MN04 via API.
Precondition	User is authorized; MN04 successfully completed.
Test data	-
Test steps	1. Send **DELETE**: `{{baseUrl}}/obj/note/:UniqueID` request, input correct UniqueId (taken from MN03: "Expected result")
Expected result	• Successful code: 204 No Content • The note created in MN03 is deleted
Result	Passed

Test case ID	**MN09: Deleted note is no longer visible in the UI**
Description	Check that the deleted note is no longer visible in the CRM app.
Precondition	MN06 deal is opened; MN08 successfully completed.
Test data	-
Test steps	1. Scroll down on the opened deal from MN06 until you see notes, refresh if needed
Expected result	• There is only one note left which was created in MN06.
Result	Passed

Managing notes 2

Description of the user flow: New deal is created via API in the: "Team6 E2E Funnel1" funnel. After that a new note is created via UI which is linking the note to the previously created new deal. The note will be then modified in the API.

Test case ID	**MN02.01: Create a new funnel via UI**
Description	Check that it is possible to create a new funnel via CRM app.
Precondition	-
Test data	Funnel name: "Team6 E2E Funnel1" Stage name: "Team6 E2E Stage1" Stage color: blue Member: Terrence Ross

Test steps	1. Open: "Funnels" page
	2. Click on: "Add funnel" on the right side
	3. Enter funnel name and click enter
	4. Enter stage name and pick stage color
	5. Click on: "Add stage"
	6. Click on: "Add members"
	7. Click on: "Choose some options" and click on a member
	8. Click on: "Save funnel"
Expected result	• The new funnel is created without any errors.
Result	Passed

Test case ID	MN02.02: Newly created funnel via UI is visible in the API
Description	Check that the new funnel which was created via UI in MN2.01 is visible in the API.
Precondition	User is authorized; MN02.01 successfully completed.
Test data	-
Test steps	1. Send **GET**:`{{baseUrl}}/obj/funnel?constraints=[{"key": "name_text", "constraint_type": "equals", "value":"Team6 E2E Funnel1"}]` request 2. Copy the Id of the funnel for the next test case

Expected result	• Successful code: 200 OK
	• Funnel is visible in the API:
	
	<pre>{
	        "members_list_user": [
	            "1625051658962x581289845978797800",
	            "1625047373081x778685405315052900"
	        ],
	        "name_text": "Team6 E2E Funnel1",
	        "Created Date": "2024-09-11T13:55:23.595Z",
	        "Created By":
	"1625051658962x581289845978797800",
	        "Modified Date": "2024-09-11T13:55:24.390Z",
	        "stage_list_custom_stage": [
	            "1726062924335x630819629448174100"
	        ],
	        "_id": "1726062924449x690382322011734000"
	    }</pre>

Result	Passed

Test case ID	**MN02.03: Create a new deal with the created funnel via API**
Description	Check that it is possible to create a new deal with the funnel created in MN02.01
Precondition	User is authorized; MN02.01 successfully completed; Funnel Id from the MN02.02 Step 2.

Test data	<pre>{
	  "assignee__user": "1625047373081x778685405315052900",
	  "company_name_text": "Team 6 E2E with funnel",
	  "deal_value_estimation_number": 71200,
	  "funnel_custom_funnel1": "1726062924449x690382322011734000",
	  "name_text": "Team6 E2E.2",
	  "order_number": 1,
	  "status_option_status": "In progress",
	  "visible_to_list_user": [
	    "1625047373081x778685405315052900"
	  ]
	}</pre>

Test steps	1. Send **POST**: {{baseUrl}}/obj/deal request

Expected result	• Successful code: 201 Created
	• New deal with the funnel created in MN02.01 is successfully done without any errors
	
	<pre>{
	    "status": "success",
	    "id": "1726069379419x155711965956960130"
	}</pre>

Result	Passed

Test case ID	**MN02.04: Deal created via API is visible in the UI**
Description	Check that the deal created in MN02.03 is visible in the CRM app.
Precondition	MN02.03 successfully completed.
Test data	-

Test steps	1. Open Deals page 2. Click on: "Switch to List view" left to the: "New Deal" button 3. In the: "Deal" search bar insert "Team6", other fields leave empty 4. Once: "Team6 E2E.2" gets highlighted in UI, click on it 5. Press the pencil icon to enter the: "Edit" mode
Expected result	Step 1: Deals page is opened Step 2: Different search fields are presented Step 5: Deal's information in the UI is the same as the entered information in MN02.03
Result	Passed

Test case ID	**MN02.05: Create a new note via UI**
Description	Check that it is possible to create a new note via CRM app for the MN02.04 deal.
Precondition	MN02.04 is successfully completed; Deal is opened from MN02.04.
Test data	Note: "This is a note from team6"
Test steps	1. Scroll down on the opened deal from MN02.04 until you see: "Type your note here…" 2. Write a note and click on: "Add" on the right side
Expected result	• A new note is successfully added
Result	Passed

Test case ID	**MN02.06: Newly created note via UI is visible in the API**
Description	Check that the new note which was created via CRM app in MN02.05 is visible in the API.
Precondition	User is authorized; MN02.05 successfully completed.
Test data	-

Test steps	1. Send **GET**: `{{baseUrl}}/obj/deal/:UniqueID` request, input correct UniqueID (taken from MN02.03: "Expected result") 2. Copy the Id of the note for the future use

Expected result	• Successful code: 200 OK
	• Note is visible in the deal ("note_list_custom_note"):
	
	`"note_list_custom_note": [`
	`    "1726070192509x612500654574010400"`
	`],`

Result	Passed

Test case ID	**MN02.07: Modify the note via API**
Description	Check that it is possible to modify the note which was created in MN02.05 via API.
Precondition	User is authorized; Id from the test case MN02.06 Step 2.

Test data	`{`
	`  "content_text": "Team6 E2E note is now updated"`
	`}`

Test steps	1. Send **PATCH**: `{{baseUrl}}/obj/note/:UniqueID` request, input correct UniqueID
Expected result	• Successful code: 204 No Content • The note is successfully updated
Result	Passed

Test case ID	**MN02.08: Modified note is seen with updated information in UI**
Description	Check that the modified note is seen with the right information in the CRM app.
Precondition	MN02.07 successfully completed; Deal is opened from MN02.05.
Test data	-

Test steps	1. Scroll down on the opened deal from MN02.05 until you see notes, refresh if needed
Expected result	• Updated note is seen and with correct information
Result	Passed