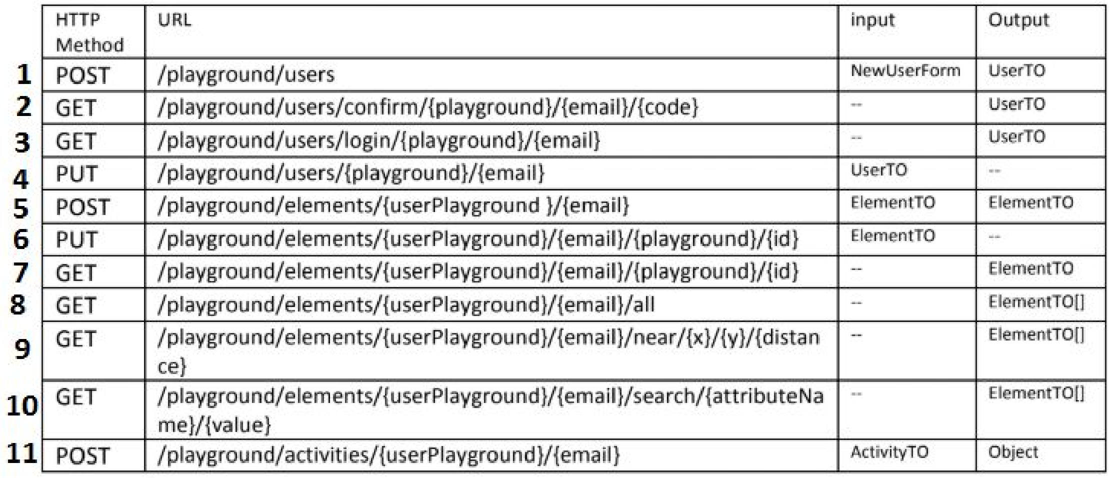
Gherkins

Table of content:



	HTTP Method	URL	input	Output
1	POST	/playground/users	NewUserForm	UserTO
2	GET	/playground/users/confirm/{playground}/{email}/{code}	--	UserTO
3	GET	/playground/users/login/{playground}/{email}	--	UserTO
4	PUT	/playground/users/{playground}/{email}	UserTO	--
5	POST	/playground/elements/{userPlayground }/{email}	ElementTO	ElementTO
6	PUT	/playground/elements/{userPlayground}/{email}/{playground}/{id}	ElementTO	--
7	GET	/playground/elements/{userPlayground}/{email}/{playground}/{id}	--	ElementTO
8	GET	/playground/elements/{userPlayground}/{email}/all	--	ElementTO[]
9	GET	/playground/elements/{userPlayground}/{email}/near/{x}/{y}/{distance}	--	ElementTO[]
10	GET	/playground/elements/{userPlayground}/{email}/search/{attributeName}/{value}	--	ElementTO[]
11	POST	/playground/activities/{userPlayground}/{email}	ActivityTO	Object

Rest Template Tests

1. Feature - Register User

1.1 Scenario: Test register with wrong email (no “@” and web address afterwards)
Scenario status: Pass
When: I POST /playground/users
With headers: Accept:application/json, content-type: application/json
With "email" : "WrongEmail” in NewUserForm body
Then: RuntimeException appears

1.2 Scenario: Test successful register
Scenario status: Pass
When: I POST /playground/users
With headers: Accept:application/json, content-type: application/json
With Body:
{"email" : "nudnik@mail.ru", "username":"Curiosity", "avatar":"MyAvatar.jpg", "role":"PLAYER"}
Then: the response body contains new UserTO with the "email" : "nudnik@mail.ru", “username”: “Curiosity”, "avatar:"MyAvatar.jpg", "role":"PLAYER""
AND database contains new UserTO with the same fields as in the body

1.3 Scenario: Test register user with the same email as the one that another user already have in the database
Scenario status: Pass
Given: user with email “nudnik@mail.ru” and with playground=”playground.rolnik” exists in userService
When: I POST /playground/users
With headers: Accept:application/json, content-type: application/json
With Body:
{"email" : "nudnik@mail.ru", "username":"Curiosity", "avatar":"MyAvatar.jpg", "role":"PLAYER"}
Then: a null user is returned in JSON

2. Feature - Confirm User

2.1 Scenario: Email not registered.
Scenario status: Pass
Given: Email is not on the database
When: I GET /playground/users/confirm/playground.rolnik/nudnik@mail.ru/1234
With headers: Accept:application/json Content-Type: application/json
Then: the response status is <> 2xx

2.2 Scenario : Successful confirmation
Scenario status: Pass
Given: database contains a User with mail test@test.com in
playground.rolnik with a generated verificationCode
When: I GET /playground/users/confirm/playground.rolnik/test@test.com/{code}
AND {code} is verificationCode
Then: the retrieved UserTO email and playground matches test@test.com and
playground.rolnik with a null verificationCode

2.3 Scenario: Playground doesn’t match user
Scenario status: Pass
Given: database contains a User with mail test@test.com in
playground.rolnik with a generated verificationCode
When: I GET /playground/users/confirm/playground.other/test@test.com/{code}
AND {code} is verificationCode
Then: the response status is <> 2xx

2.4 Scenario: Email is registered but verification code is wrong
Scenario status: Pass
Given: database contains a User with mail test@test.com in
playground.rolnik with a generated verificationCode
When: I GET /playground/users/confirm/playground.other/test@test.com/{code}
AND {code} is retrieved from the verificationCode+”X”
Then: the response status is <> 2xx

3. Feature - Login

3.1 Scenario: Successful Login

Scenario status: Pass

Given: database contains a User with mail test@test.com in
playground.rolnik AND User is verified
When: I GET /playground/users/login/playground.rolnik/test@test.com
Then: The retrieved UserTO email matches test@test.com and playground matches playground.rolnik.

3.2 Scenario: Email not in Database

Scenario status: Pass

Given: Database does not contain User with email test@test.com
When: I GET /playground/users/login/playground.rolnik/test@test.com
Then: The response status is <> 2xx

3.3 Scenario: User doesn’t belong in playground

Scenario status: Pass

Given: User is verified in database with email: test@test.com AND playground is playground.other AND User is verified.
When: I GET /playground/users/login/playground.other/test@test.com
Then: The response status is <> 2xx

3.4 Scenario: User not verified

Scenario status: Pass

Given: User is in database with email: test@test.com AND playground is playground.other AND User is NOT verified.
When: I GET /playground/users/login/playground.other/test@test.com
Then: The response status is <> 2xx

4. Feature - Update user information (avatar,name,role...)

4.1 Scenario: User updates his info
Scenario status: Pass
Given: USER is in verified on the database
moderatorTest@test.com in playground.rolnik
When: I PUT /playground/users/playground.rolnik/moderatorTest@test.com
With headers: Accept:application/json, content-type: application/json
With Body:
{"email" : moderatorTest@test.com", "username":ModeratorName,
"avatar":MyAvatar.jpg, "role":MODERATOR, “playground”:playground.rolnik}
Then: The response body contains moderatorTest@test.com USER with updates.

4.2 Scenario: User updates another user
Scenario status: Pass
Given: USER is in verified on the database
moderatorTest@test.com in playground.rolnik AND
USER2 is in verified on the database
moderatorTest2@test.com in playground.rolnik
When: I PUT /playground/users/playground.rolnik/moderatorTest@test.com
With headers: Accept:application/json, content-type: application/json
With Body:
{"email" : moderatorTest2@test.com", "username":ModeratorName,
"avatar":MyAvatar.jpg, "role":MODERATOR, “playground”:playground.rolnik}
Then: the response status is <> 2xx

5. Feature - Save element

5.1 Scenario: Test Save element in database

Scenario status: Pass

Given: database contains verified user with mail : "nudnik@mail.ru" in playground playground.rolnik

When: I POST /playground/elements/playground.rolnik /nudnik@mail.ru

AND I post new element with body contains: playground="playground",creatorPlayground="playground.rolnik" ,name="nameOfElement"

Then: The element is saved in the ElementService AND I receive element with body:playground="playground",creatorPlayground="playground.rolnik" ,name="nameOfElement"

5.2 Scenario: Add message board as Manager

Scenario status: Pass

Given: I have a user with playground “playground.rolnik” and email nudnik@mail.ru in database role manager

When: I POST /playground/activities/playground.rolnik/nudnik@mail.ru

With headers: Accept:application/json, content-type: application/json

With element in body which contains fields: type “ELEMENT_MESSAGEBOARD_TYPE”

Then: An element object of type ELEMENT_MESSAGEBOARD_TYPE is returned

5.3 Scenario: Add message board as Player

Scenario status: Pass

Given: I have a user with playground “playground.rolnik” and email nudnik@mail.ru in database role player

When: I POST /playground/activities/playground.rolnik/nudnik@mail.ru

With headers: Accept:application/json, content-type: application/json

With element in body which contains fields: type “ELEMENT_MESSAGEBOARD_TYPE”

Then: response status is <>2xx

5.4 Scenario: Adding question to Playground as Manager.

Scenario status: Pass

Given: I have a user of type Manager with playground “playground.rolnik” and email test@test.com in database

When: I POST /playground/activities/playground.rolnik/nudnik@mail.ru

With headers: Accept:application/json, content-type: application/json

With element in body which contains fields: type

ELEMENT_QUESTION_TYPE,
ElementName= “question title”,
ELEMENT_QUESTION_KEY= “question”,
ELEMENT_ANSWER_KEY=”answer”
ELEMENT_POINT_KEY=5

Then: The element is returned after being added to database

5.5 Scenario: Adding question to Playground as Player.
Scenario status: Pass
Given: I have a user of type Player with playground “playground.rolnik” and email nudnik@mail.ru in database
When: I POST /playground/activities/playground.rolnik/nudnik@mail.ru
With headers: Accept:application/json, content-type: application/json
With element in body which contains fields: type
ELEMENT_QUESTION_TYPE,
ElementName= “question title”,
ELEMENT_QUESTION_KEY= “question”,
ELEMENT_ANSWER_KEY=”answer”
ELEMENT_POINT_KEY=5

Then: response status is <> 2xx

5.6 Scenario: Adding to Playground a question with a missing attribute.
Scenario status: Pass
Given: I have a user with playground “playground.rolnik” and emailnudnik@mail.ru in database
When: I POST /playground/activities/playground.rolnik/nudnik@mail.ru
With headers: Accept:application/json, content-type: application/json
With activity in body which contains fields: type ADD_QUESTION_ACTIVITY, playground = "Playground" ,
ACTIVITY_SET_QUESTION_QUESTION_TITLE= “question title”,
ACTIVITY_SET_QUESTION_QUESTION= “question”,
ACTIVITY_SET_QUESTION_POINTS=5

Then: response status is <> 2xx

6. Feature - Update Element

6.1 Scenario: Test successfully update element in database
Scenario status: Pass
Given: Database contains element with fields userplayground = "playground.rolnik", id = “123”, creator email = “nudnik@mail.ru”, playground = “playground”

AND database contains verified user with "nudnik@mail.ru" in playground userPlayground with name "Rolnik_element”
When: I PUT /playground/elements/userPlayground/nudnik@mail.ru/playground/123
With headers: Accept:application/json, content-type: application/json
With element in body contains fields user playground = "userPlayground", playground
= "forTest", id = “123”, creator email = “updated@email.com”
Then: element service contains successfully updated element with playground = “playground.rolnik” and name “changedName”

6.2 Scenario: Test update non-existent element
Scenario status: Pass
Given: Database doesn’t contain element with fields userPlayground = “playground.rolnik”,
playground = “playground”
AND database contains verified user with nudnik@mail.ru in playground userPlayground
When: I PUT /playground/elements/userPlayground/nudnik@mail.ru/playground/123
With headers: Accept:application/json, content-type: application/json
With element in body contains fields user playground = "userPlayground", playground
= "wrongPlayground"
Then: the response status is <> 2xx

6.3 Scenario : Test update existing element with non existing email in playground
Scenario status: Pass
Given: Database doesn’t contains a User with mail test@test.com in
“random_playground” and {generated id}
When: I PUT /playground/elements/userPlayground/test@test.com/playground/{generated id}
With headers: Accept:application/json, content-type: application/json
With body: {“userPlayground” = “userPlayground”, “creatorEmail” = “wrong@email.com”, “id” ={generated id}}
Then: the response status is <> 2xx

6.4 Scenario : Test update ID on existing elements
Scenario status: Pass
Given: Database contains element with fields userPlayground = "playground.rolnik",
id = “123”, creator email = “nudnik@mail.ru”, playground = “playground”
AND database contains verified user with "nudnik@mail.ru" in playground
userPlayground
When: I PUT /playground/elements/playground.rolnik/nudnik@mail.ru/playground/123

With headers: Accept:application/json, content-type: application/json

With element in body contains fields user playground = "userPlayground", playground = "forTest", id = “1234567”, creator email = “nudnik@mail.ru”

Then: the response status is <> 2xx

6.5 Scenario : Test update Playground on existing elements

Scenario status: Pass

Given: Database contains element with fields userplayground = "playground.rolnik", id = “123”, creator email = “nudnik@mail.ru”, playground = “playground”
AND database contains verified user with "nudnik@mail.ru" in playground userPlayground

When: I PUT /playground/elements/playground.rolnik/nudnik@mail.ru/playground/123

With headers: Accept:application/json, content-type: application/json

With element in body contains fields user playground = "OtheruserPlayground", playground = "forTest", id = “123”, creator email = “updated@email.com”

Then: the response status is <> 2xx

7. Feature - Getting an element by id

7.1 Scenario: Get element with incorrect login details

Scenario status: Pass

Given: User in database with mail nudnik@mail.ru and playground.rolnikis playground exists in database AND
Element in database with playground: playground and id: 123 exists

When: I get /playground/elements/otherPlayground/user@test.com/playground.rolnik/123
Then: the response status is <> 2xx

7.2 Scenario: Get element with correct login details and element doesn’t exist
Scenario status: Pass
Given: User in database with mail nudnik@mail.ru and userPlayground is playground exists in database AND
Element with playground: playground and id: 123 not in database
When: I get /playground/elements/playground.rolnik/user@test.com/playground.rolnik/123
Then: the response status is <> 2xx

7.3 Scenario: Get Element with correct login details and element exists
Scenario status: Pass
Given: User in database with mail nudnik@mail.ru and playground.rolnikis playground exists in database AND
Element in database with playground: playground and id: 123 exists
When: I get /playground/elements/playground.rolnik/nudnik@mail.ru/playground.rolnik/123
Then: I get the element that contains playground playground: playground and id: 123

8. Feature - Get all elements

8.1 Scenario: Test get all elements from database
Scenario status: Pass
Given: Database contains verified user with "nudnik@mail.ru" in playground userPlayground AND elements in database are with bodies:
1.playground="playground",creatorPlayground="creator",name="nameOfElement" ,location=("1,2")
2.playground="playground",creatorPlayground="creator",name="nameOfElement" ,location=("2,1")

When: I GET /playground/elements/userPlayground/nudnik@mail.ru/all

Then: all elements :

1.playground="playground",creatorPlayground="creator",name="nameOfElement"
,location=("1,2")
2.playground="playground",creatorPlayground="creator",name="nameOfElement"
,location=("2,1")
are sent back as ElementTO[].

8.2 Scenario: Test get all elements from empty database
Scenario status: Pass
Given: Database containing elements is empty
AND database contains verified user with "nudnik@mail.ru" in playground userPlayground
When: I GET /playground/elements/userPlayground/nudnik@mail.ru/all
Then:empty list of ElementTO[] is received.

8.3 Scenario: Test get all elements from database with pagination
Scenario status: Pass
Given: Database contains Database contains verified user with "nudnik@mail.ru" in playground
AND Database contains elements in database are with bodies:

1.playground="playground",creatorPlayground="creator",name="1nameOfElement"
2.playground="playground",creatorPlayground="creator",name="2nameOfElement"
3.playground="playground",creatorPlayground="creator",name="3nameOfElement"
4.playground="playground",creatorPlayground="creator",name="4nameOfElement"
5.playground="playground",creatorPlayground="creator",name="5nameOfElement"
6.playground="playground",creatorPlayground="creator",name="6nameOfElement"

When: I GET /playground/elements/userPlayground/nudnik@mail.ru/all/page=1&size=3

Then: all elements will be returned from page=1 and size=3:

4.playground="playground",creatorPlayground="creator",name="4nameOfElement"
5.playground="playground",creatorPlayground="creator",name="5nameOfElement"
6.playground="playground",creatorPlayground="creator",name="6nameOfElement"

9. Feature: Get elements in radius

9.1 Scenario: Negative Distance
Scenario status: Pass
Given: Ddatabase contains verified user with "nudnik@mail.ru" in playground userPlayground AND elements in database are with bodies:

1.playground="playground",creatorPlayground="creator",id="idOfElement"
,location=("1,2")

When: I GET /playground/elements/userPlayground/nudnik@mail.ru/near/1/2/%2D
1
Then:the response status is <> 2xx

9.2 Scenario: Distance is Zero

Scenario status: Pass

Given: Ddatabase contains verified user with "nudnik@mail.ru" in playground userPlayground AND elements in database are with bodies:

1.playground="playground",creatorPlayground="creator",id="idOfElement" ,location=("1,2")

2.playground="playground",creatorPlayground="creator",id="idOfElement" ,location=("1,3")

When: I GET /playground/elements/userPlayground/nudnik@mail.ru/near/1/2/0

Then: I get ElementTO[] only elements on coordinate location=("1,2").

9.3 Scenario: Distance is greater than Zero

Scenario status: Pass

Given: Database contains verified user with "nudnik@mail.ru" in playground userPlayground AND Database contains elements in database with bodies:

1.playground="playground",creatorPlayground="creator",name="nameOfElement1" ,location=("0,1")

2.playground="playground",creatorPlayground="creator",name="nameOfElement2" ,location=("0,2")

3.playground="playground",creatorPlayground="creator",name="nameOfElement3" ,location=("3,7")

4.playground="playground",creatorPlayground="creator",name="nameOfElement4" ,location=("0,7")

5.playground="playground",creatorPlayground="creator",name="nameOfElement5" ,location=("7,7")

When: I GET /playground/elements/userPlayground/nudnik@mail.ru/near/0/0/7

Then: I get ElementTO[] with the elements with location:

1..location=("0,1")

2.location=("0,2")

3.location=("0,7")

9.4 **Scenario: Distance is greater than Zero with Pagination**

Scenario status: Pass

Given: Database contains verified user with "nudnik@mail.ru" in playground userPlayground AND Database contains elements in database with bodies:

1.playground="playground",creatorPlayground="creator",name="1nameOfElement" ,location=("0,1")

2.playground="playground",creatorPlayground="creator",name="2nameOfElement" ,location=("0,2")

3.playground="playground",creatorPlayground="creator",name="3nameOfElement" ,location=("0,3")

4.playground="playground",creatorPlayground="creator",name="4nameOfElement" ,location=("0,4")

5.playground="playground",creatorPlayground="creator",name="5nameOfElement5" ,location=("0,5")

6.playground="playground",creatorPlayground="creator",name="6nameOfElement5" ,location=("0,6")

When: I GET /playground/elements/userPlayground/nudnik@mail.ru/near/5/6/6/page=1&size=3
Then: I get ElementTO[] with the elements:

5.playground="playground",creatorPlayground="creator",name="5nameOfElement5" ,location=("5,5")

6.playground="playground",creatorPlayground="creator",name="6nameOfElement5" ,location=("6,6")

10. Feature: GET all elements with attribute name and attribute value

10.1 Scenario: Successfully Get Elements By Attribute Name and Attribute Value with Pagination

Scenario status: Pass

Given: Database contains Database contains verified user with "nudnik@mail.ru" in playground
AND Database contains elements in database are with bodies:

1.playground="playground",creatorPlayground="creator"
2.playground="playground",creatorPlayground="creator"
3.playground="playground",creatorPlayground="creator"
4.playground="playground",creatorPlayground="creator"
5.playground="playground",creatorPlayground="creator"
6.playground="playground",creatorPlayground="creator", creatorEmail=”6nudnik@mail.ru”
7.playground="playground",creatorPlayground="creator", creatorEmail=”7nudnik@mail.ru”
8.playground="playground",creatorPlayground="creator", creatorEmail=”8nudnik@mail.ru”
9.playground="playground",creatorPlayground="creator"
10.playground="playground",creatorPlayground="creator"

AND elements 3-9 contain attribute with attribute name = “type” and attribute value = "Element_Question_type"

When: I GET /playground/elements/creatorPlayground/nudnik@mail.ru/search/type/Element_Question_type?page=1&size=3
Then: all elements from page=1 and size=3 will be returned:
6.playground="playground",creatorPlayground="creator",name="Rolnik_element", creatorEmail=”6nudnik@mail.ru”
7.playground="playground",creatorPlayground="creator",name="Rolnik_element", creatorEmail=”7nudnik@mail.ru”
8.playground="playground",creatorPlayground="creator",name="Rolnik_element", creatorEmail=”8nudnik@mail.ru”

10.2 Scenario: Test get Elements by non existing attribute name in ElementService
Scenario status: Pass
Given: Element service doesn't contains element with attributeName = “no_such_attribute”
With headers: Accept:application/json, content-type: application/json
When: I GET /playground/elements/creatorPlayground/nudnik@mail.ru/search/no_such_name/Element.ELEMENT_QUESTION_TYPE
Then: response status is <> 2xx

10.3 Scenario: Test no Elements in ElementService with searched {value} returns empty array of ElementTO
Scenario status: Pass
Given: Element service doesn’t contains element of type question with value “no_such_attribute_value”
With headers: Accept:application/json, content-type: application/json
When: I GET /playground/elements/creatorPlayground/nudnik@mail.ru/search/type/no_such_attribute_value
Then: server returns empty ElementTO[] array

11. Feature - Send Activity to server

11.1 Scenario: Sending Echo activity
Scenario status: Pass

Given: I have a user with playground “playground.rolnik” and email nudnik@mail.ru in database
When: I POST /playground/activities/playground.rolnik/nudnik@mail.ru
With headers: Accept:application/json, content-type: application/json
With activity in body which contains fields: type “echo”, playground = "playground.rolnik"

Then: An activity object with is returned with body which contains fields: type “echo”,

11.2 Scenario: Sending Message activity
Scenario status: Pass
Given: I have a user with playground “playground.rolnik” and email nudnik@mail.ru in database AND element of type messageboard with id 1 exists in the database
When: I POST /playground/activities/playground.rolnik/nudnik@mail.ru
With headers: Accept:application/json, content-type: application/json
With activity in body which contains fields: type “get_message”, playground = "Playground" and messageAttribute = “message”, messageBoard_id=”1”

Then: An activity object with is returned with body which contains fields: type “message”, playground = "playground.rolnik"

11.3 Scenario: Sending Message activity to non existing board
Scenario status: Pass
Given: I have a user with playground “playground.rolnik” and email nudnik@mail.ru in database AND element of type messageboard with id 1 doesnt exist in the database
When: I POST /playground/activities/playground.rolnik/nudnik@mail.ru
With headers: Accept:application/json, content-type: application/json
With activity in body which contains fields: type “get_message”, playground = "Playground" and messageAttribute = “message”, messageBoard_id=”1”

Then: response status is <> 2xx

11.4 Scenario: Read existing question activity
Scenario status: Pass
Given: I have a user with playground “playground.rolnik” and email nudnik@mail.ru in database

AND element of type ELEMENT_QUESTION_TYPE with id q1 exist in the database

When: I POST /playground/activities/playground.rolnik/nudnik@mail.ru

With headers: Accept:application/json, content-type: application/json

With activity in body which contains fields: type “QUESTION_READ_ACTIVITY”, and ElementId=”q1”

Then: response return the question element with id=q1 from the database

11.5 Scenario: Reading a question that not exist in database.

Scenario status: Pass

Given: I have a user with playground “playground.rolnik” and email nudnik@mail.ru in database

AND element of type ELEMENT_QUESTION_TYPE with id q1 doesn’t exist in the database

When: I POST /playground/activities/playground.rolnik/nudnik@mail.ru

With headers: Accept:application/json, content-type: application/json

With activity in body which contains fields: type “QUESTION_READ_ACTIVITY”, and ElementId=”q1”

Then: response status is <>2xx

11.6 Scenario: Answering a question with correct answer.

Scenario status: Pass

Given: I have a user with playground “playground.rolnik” and email nudnik@mail.ru, points=0 in database

AND element of type question with id q1 exist in the database and ELEMENT_ANSWER_KEY=”answer ”, point_value=5

When: I POST /playground/activities/playground.rolnik/nudnik@mail.ru

With headers: Accept:application/json, content-type: application/json

With activity in body which contains fields: type “QUESTION_ANSWER_ACTIVITY”, playground = "Playground" and , ACTIVITY_USER_ANSWER_KEY=”answer” ,question_id=”q1”,

Then: value true is returned, and user points=5

11.7 Scenario: Answering a question with incorrect answer.

Scenario status: Pass

Given: I have a user with playground “playground.rolnik” and email nudnik@mail.ru, points=0 in database

AND element of type question with id q1 exist in the database and the ELEMENT_ANSWER_KEY =”answer ”, point_value=5

When: I POST /playground/activities/playground.rolnik/nudnik@mail.ru
With headers: Accept:application/json, content-type: application/json
With activity in body which contains fields: type “QUESTION_ANSWER_ACTIVITY”, playground = "Playground" and , ACTIVITY_USER_ANSWER_KEY=”bad_answer” ,question_id=”q1”,

Then: value false is returned, and user points=0

11.8 Scenario: getting score from database from highest to lowest with pagination.
Scenario status: Pass
Given:
User with ‘creatorPlayground’ = “UserPlayground” and email ‘0nudnik@mail.ru’ with points =100 is in database
User with ‘creatorPlayground’ = “UserPlayground” and email ‘1nudnik2@mail.ru’ with points =90 is in database
User with ‘creatorPlayground’ = “UserPlayground” and email ‘2nudnik@mail.ru’ with points =80 is in database
User with ‘creatorPlayground’ = “UserPlayground” and email ‘3nudnik@mail.ru’ with points =75 is in database
User with ‘creatorPlayground’ = “UserPlayground” and email ‘4nudnik@mail.ru’ with points =60 is in database
User with ‘creatorPlayground’ = “UserPlayground” and email ‘5nudnik@mail.ru’ with points =50 is in database
User with ‘creatorPlayground’ = “UserPlayground” and email ‘6nudnik7@mail.ru’ with points =40 is in database
User with ‘creatorPlayground’ = “UserPlayground” and email ‘7nudnik8@mail.ru’ with points =30 is in database
User with ‘creatorPlayground’ = “UserPlayground” and email ‘8nudnik9@mail.ru’ with points =20 is in database
User with ‘creatorPlayground’ = “UserPlayground” and email ‘9nudnik9@mail.ru’ with points =20 is in database

When: I POST /playground/activities/UserPlayground/test@test.com

With headers: Accept:application/json, content-type: application/json

With activity in body which contains fields: type “GET_SCORES_ACTIVITY”,

Then: response return users:

User with ‘creatorPlayground’ = “UserPlayground” and email ‘0nudnik@mail.ru’ with points =100 is in database

User with ‘creatorPlayground’ = “UserPlayground” and email ‘1nudnik@mail.ru’ with points =90 is in database

User with ‘creatorPlayground’ = “UserPlayground” and email ‘2nudnik@mail.ru’ with points =80 is in database

User with ‘creatorPlayground’ = “UserPlayground” and email ‘3nudnik@mail.ru’ with points =75 is in database

User with ‘creatorPlayground’ = “UserPlayground” and email ‘4nudnik@mail.ru’ with points =60 is in database

User with ‘creatorPlayground’ = “UserPlayground” and email ‘5nudnik@mail.ru’ with points =50 is in database

User with ‘creatorPlayground’ = “UserPlayground” and email ‘6nudnik@mail.ru’ with points =40 is in database

User with ‘creatorPlayground’ = “UserPlayground” and email ‘7nudnik@mail.ru’ with points =30 is in database

User with ‘creatorPlayground’ = “UserPlayground” and email ‘8nudnik@mail.ru’ with points =20 is in database

User with ‘creatorPlayground’ = “UserPlayground” and email ‘9nudnik9@mail.ru’ with points =20 is in database

11.9 Scenario: Getting game rules

Scenario status: Pass

Given: I have a user with playground “playground.rolnik” and email nudnik@mail.ru in database and his role is player

When: I POST /playground/activities/playground.rolnik/nudnik@mail.ru

With headers: Accept:application/json, content-type: application/json

With activity in body which contains fields: type “GET_GAME_RULES_ACTIVITY”

Then: I get in body a string containing the rules of the game

JPA Tests

1. Feature - Add Element

1.1 Scenario: Adding non existing element
Scenario status: Pass
Given: There is no element in database
When: When I add an element with id 1 and playground.rolnik
Then: Then it exists in the database

1.2 Scenario: Adding an existing element to jpa
Scenario status: Pass
Given: There is an element with id 1 and playground.rolnik
When: When I add an element with id 1 and playground.rolnik
Then: It is not added to the database

2. Feature - Add User

2.1 Scenario: Adding non existing user
Scenario status: Pass
Given: There is no user in database
When: When I add a user with userPlayground playground.rolnik and email@mail.com
Then: Then it exists in the database

2.2 Scenario: Adding an existing user to jpa
Scenario status: Pass
Given: There is a user with userPlayground playground.rolnik and email@mail.com
When: When I add a user with userPlayground playground.rolnik and email@mail.com
Then: the response status is <> 2xx