

Testing source:<https://open.spotify.com/>  
Device: PC  
OS: Windows 11/64  
Type: functional testing

Test Prepared By: Max Petrov

Test Case ID	Business Requirement ID	Test Priority	Test Title (REQ)	Step description	Test Data	Expected Result	Actual Result	Status (Pass/Fail) Google Chrome	Status (Pass/Fail) Mozilla Firefox	Status (Pass/Fail) Microsoft Edge	Notes
Functional Positive Testing											
TC01	BR01	High	The site must be accessible at the following URL: <a href="https://open.spotify.com/">https://open.spotify.com/</a>	1. Enter <a href="https://open.spotify.com/">https://open.spotify.com/</a> in the browser		Https protocol is active	As expected	Pass	Pass	Pass	
TC02	BR02	High	Verify that the user can sign up into the site using a correct mail and password	1. Enter <a href="https://open.spotify.com/">https://open.spotify.com/</a> in the browser 2. Click on the "Sign up" icon in right header menu 3. Find menu "Sign up with your email address" 4. Click on the field "Enter your email." 5. Enter email data from test data 1 6. Click on the field "Enter your email again." 7. Enter email data from test data 1 again 8. Click on the field "Create a password." 9. Enter password data from test data 2 10. Click on the field "Enter a profile name." 11. Enter profile name from test data 3 12. Enter in menu "What's your date of birth?" data from test data 4 (Month, Day, Year) 13. Choose in radio buttons "What's your gender?" button from test data 5 14. Click on the captcha checkboxes "I'm not a robot" 15. Click on "Sign up" button	<b>Test data 1:</b> rasacet733@charav.com <b>Test data 2:</b> rasacet733! <b>Test data 3:</b> rasacet733 <b>Test data 4:</b> May 20 1999 <b>Test data 5:</b> Male	The user successfully registered on the sites	The user successfully registered on the site	Pass	Pass	Pass	
TC03	BR03	High	Verify that the user can Log In into the site using an existing mail and password	1. Enter <a href="https://open.spotify.com/">https://open.spotify.com/</a> in the browser 2. Click on the "Log in" icon in right header menu 3. Enter email data from test data 1 4. Click on the field "Password" 5. Enter password data from test data 2 6. Click on the button "Log In"	<b>Test data 1:</b> rasacet733@charav.com <b>Test data 2:</b> rasacet733!	The user successfully entered on the sites	The user successfully entered on the sites	Pass	Pass	Pass	
TC04	BR04	High	Validate that menu links lead to the right pages ("Home", "Search", "Your Library", "Liked Songs")	1. Enter <a href="https://open.spotify.com/">https://open.spotify.com/</a> in the browser 2. Find menu links in the left side ("Home", "Search", "Your Library", "Liked Songs") 3. Click on the "Home" 4. Compare the link was opened with the test data 1 5. Hover over the tab and compare the title with Test data 1.1 6. Click on the "Search" 7. Compare the link was opened with the test data 2 8. Hover over the tab and compare the title with Test data 2.1 9. Click on the "Your Library" 10. Compare the link was opened with the test data 3 11. Hover over the tab and compare the title with Test data 3.1 12. Click on the "Your Library" 13. Compare the link was opened with the Test data 4 14. Hover over the tab and compare the title with Test data 4.1 15. Click on the "Home" 16. Compare the link was opened with the test data 1 17. Hover over the tab and compare the title with Test data 1.1	<b>Test data 1:</b> <a href="https://open.spotify.com/">https://open.spotify.com/</a> <b>Test data 1.1:</b> Spotify – Web Player <b>Test data 2:</b> <a href="https://open.spotify.com/search">https://open.spotify.com/search</a> <b>Test data 2.1:</b> Spotify – Search <b>Test data 3:</b> <a href="https://open.spotify.com/playlist">https://open.spotify.com/playlist</a> <b>Test data 3.1:</b> Spotify – Your Library <b>Test data 4:</b> <a href="https://open.spotify.com/track">https://open.spotify.com/track</a> <b>Test data 4.1:</b> Spotify – Liked Songs	All header menu links lead to the right pages	As expected	Pass	Pass	Pass	
TC05	BR05	Medium	Verify that the user can find a specific query (artist, song, playlist, album)	1. Enter <a href="https://open.spotify.com/">https://open.spotify.com/</a> in the browser 2. Find menu links "Search" in the left side 3. Click on the "Search" 4. Click on Search Field ("What do you want to listen to?") 5. Enter Test data 1 in Search Field 6. Click on "Artist" in meue under Search Field 7. Check that the first value found matches the Test date 1 8. Click on X in right side of the Search Field 9. Enter Test data 2 in Search Field 10. Click on "Songs" in meue under Search Field 11. Check that the first value found matches the Test date 2 12. Click on X in right side of the Search Field 13. Enter Test data 1 in Search Field 14. Click on "Playlists" in meue under Search Field 15. Check that the found playlists match the Test date 1 16. Click on X in right side of the Search Field 17. Enter Test data 3 in Search Field 18. Click on "Albums" in meue under Search Field 19. Check that the first value found matches the Test date 3	<b>Test data 1:</b> The Beatles <b>Test data 2:</b> Let It Be <b>Test data 3:</b> Abbey Road	Valid request found	Valid request found	Pass	Pass	Pass	

TC06	BR06	Medium	Check that the user can create new "Playlist"	1. Enter https://open.spotify.com/ in the browser 2. Find menu links "Create Playlist" in the left side 3. Click on the "Create Playlist" 4. Check that the Playlist "My Playlist #1" added under menu in left side	New playlist successfully created	New playlist successfully created	Pass	Pass	Pass	
TC07	BR07	Medium	Check that the user can change name of "Playlist"	1. Enter https://open.spotify.com/ in the browser 2. Find and click Playlist "My Playlist #1" under menu in left side 3. Click on the "My Playlist #1" in the top right side of page 4. In opened Modal Window (pop-up) "Edit details" click on field "name" 5. Enter Test data in Search Field "name" 6. Click button "Save" in right footer in Modal Window (pop-up) "Edit details"	Test data: My Beatles Playlist	Playlist name changed successfully	Playlist name changed successfully	Pass	Pass	Pass
TC08	BR08	Medium	Check that the user can add song to a user's playlist	1. Enter https://open.spotify.com/ in the browser 2. Find Playlist "My Beatles Playlist" under menu in left side 3. Click on the "My Beatles Playlist" in the top right side of page 4. Click on Search Field "Search for songs or episodes" 5. Enter Test data in Search Field "Search for songs or episodes" 6. Check that the correct song is found (Test data ) under the Search Field 7. Click on the "add" button on the left side of the found first song menu 8. Check that the song successfully added to the "My Beatles Playlist"	Test data: Let It Be - Remastered 2009	Item added to playlist successfully	Item added to playlist successfully	Pass	Pass	Pass
TC09	BR09	Medium	Check that the user can remove item from a user's playlist.	1. Enter https://open.spotify.com/ in the browser 2. Find Playlist "My Beatles Playlist" under menu in left side 3. Click on the "My Beatles Playlist" under menu in left side 4. Find and click to add song under the playlist name (# TITLE) in center part of the page 5. Find and click on the three dots ("More options for...") in the right side of current song. 6. In opened menu find "Remove from this playlist" and click		Item deleted to playlist successfully	Item deleted to playlist successfully	Pass	Pass	Pass
TC10	BR10	Medium	Check that the user can follow to an existing playlist	1. Enter https://open.spotify.com/ in the browser 2. Find menu links "Search" in the left side 3. Click on the "Search" 4. Click on Search Field ("What do you want to listen to?") 5. Enter Test data in Search Field 6. Click on "Playlists" in menu under Search Field 7. Click on the first playlist found 8. Find and click on the three dots ("More options for...") under picture, name and descriptions of the current playlist 9. In the menu that opens, select "Add to Your Library" 10. Click on "Your Library" menu links in the left side 11. Check that the added playlist present in "Playlists"	Test data: The Beatles	Playlist successfully added in "Your Library"	Playlist successfully added in "Your Library"	Pass	Pass	Pass
TC11	BR11	Medium	Check that the user can unfollow an existing playlist	1. Enter https://open.spotify.com/ in the browser 2. Click on "Your Library" menu links in the left side 3. Find and click on the playlist link that was added to the library in the TC10 4. Find and click on the three dots ("More options for...") under picture, name and descriptions of the current playlist 5. In the menu that opens, select "Remove from Your Library" 6. Click on "Your Library" menu links in the left side 7. Check that the playlist has been successfully deleted		Playlist successfully deleted from "Your Library"	Playlist successfully deleted from "Your Library"	Pass	Pass	Pass
Functional Negative Testing										
TC12	BR12	High	Verify that the Log in with invalid email address is not accepted	1. Enter https://open.spotify.com/ in the browser 2. Click on the "Log in" icon in header menu 3. Click on the field "Email address or username" 4. Enter invalid email data from test data 5. Click on the field "Password" 6. Enter correct password, use test data 2 7. Click on the button "Log In"	Test data 1: mail@.com Test data 2: rasacet733!	Got the error message: "Incorrect username or password."	As expected	Pass	Pass	Pass

TC13	BR13	High	Verify that the Log in with invalid password is not accepted	1. Enter https://open.spotify.com/ in the browser 2. Click on the "Log in" icon in header menu 3. Click on the field "Email address or username" 4. Enter correct email data from test data 5. Click on the field "Password" 6. Enter invalid password, use test data 2 7. Click on the button "Log In"	<b>Test data 1:</b> rasacet733@charav.com <b>Test data 2:</b> 123456	Got the error message: "Incorrect username or password.	As expected	Pass	Pass	Pass
TC14	BR14	High	Verify that the Log in with empty email field is not accepted	1. Enter https://open.spotify.com/ in the browser 2. Click on the "Log in" icon in header menu 3. Leave blank field "Email address or username" 4. Click on the field "Password" 5. Enter correct password, use test data 6. Click on the button "Log In"	<b>Test data:</b> rasacet733!	Got the error message: "Incorrect username or password.	As expected	Pass	Pass	Pass
TC15	BR15	High	Verify that the Log in with empty password field is not accepted	1. Enter https://open.spotify.com/ in the browser 2. Click on the "Log in" icon in header menu 3. Click on the field "Email address or username" 4. Enter email data from test data 5. Leave blank field "Password" 6. Click on the button "Log In"	<b>Test data:</b> rasacet733@charav.com	Got the error message: "Incorrect username or password.	As expected	Pass	Pass	Pass
AD HOC TESTS										
TC16	BR16	High	Verify that the Log in with invalid password is not accepted	1. Enter https://open.spotify.com/ in the browser 2. Click on the "Log in" icon in header menu 3. Click on the field "Email address or username" 4. Enter email data from test data 1 5. Click on the field "Password" 6. Enter correct password, use test data 2 7. Click on the button "Log In"	<b>Test data 1:</b> test@mail.comtest@mail.com <b>Test data 2:</b> rasacet733!	Got the error message: "Incorrect username or password.	As expected	Pass	Pass	Pass

### Cross-browser Compatibility Matrix

Testing source: <https://open.spotify.com/>

Device: PC

OS: Windows 11/64

Browsers: Google Chrome, Mozilla Firefox, Microsoft Edge

Requirement	Test Case	Chrome 107.0.5304.107	Mozilla Firefox 106.0.5	Microsoft Edge 107.0.1418.42
BR01	TC01	P	P	P
BR02	TC02	P	P	P
BR03	TC03	P	P	P
BR04	TC04	P	P	P
BR05	TC05	P	P	P
BR06	TC06	P	P	P
BR07	TC07	P	P	P
BR08	TC08	P	P	P
BR09	TC09	P	P	P
BR10	TC10	P	P	P
BR11	TC11	P	P	P
BR12	TC12	P	P	P
BR13	TC13	P	P	P
BR14	TC14	P	P	P
BR15	TC15	P	P	P
BR16	TC16	P	P	P
Total passed cases		16	16	16
Total failed cases		0	0	0

Testing source: <https://open.spotify.com/>  
Device: PC  
OS: Windows 10/64  
Type: performance, security, API testing

Test Case ID	Business Requirement ID	Test Priority	Test Title (REQ)	Step description	Test Data	Expected Result	Actual Result	Status (Pass/Fail)	Notes
Performance Testing									
TC17	BR17	Highest	Implement Performance testing for <a href="https://open.spotify.com/">https://open.spotify.com/</a>	1. Use <a href="https://gtmetrix.com/">https://gtmetrix.com/</a> 2. Use <a href="https://www.browerstack.com/speedlab">https://www.browerstack.com/speedlab</a> 3. Use Lighthouse extension		Overall performance score should be more than 75 (Desktop)	The score is less than 75 Gtmetrix - 54 Speedlab - 28	Fail	Bug report
Security Testing									
TC18	BR18	Highest	Implement Security testing for <a href="https://open.spotify.com/">https://open.spotify.com/</a>	1. Use Snky <a href="https://snky.io/test/website-scanner/">https://snky.io/test/website-scanner/</a> 2. Use Mozilla Observatory <a href="https://observatory.mozilla.org/">https://observatory.mozilla.org/</a>		Overall security score should be more than A	The score is less than A Webpage security score - B	Fail	Bug report
API Testing									
TC19	BR19	Highest	Implement POST Request (Create a Playlist)	1. Create environment in Postman app 2. Create Pre-request Scripts (unset "Playlist name", random "Playlist name") 3. Get New Access Token ( <a href="https://accounts.spotify.com/api/token">https://accounts.spotify.com/api/token</a> ) 3. Set keys and values for the environment 4. Create Tests for checking the response 5. Send Post Request with Postman API	Test data1 Playlist name: 'My Playlist', 'My New Playlist', 'Best Music', 'Rock playlist', 'My Songs', 'Greatest Hits'	1. Status code is 201 Created 2. Response time is less than 800ms 3. Response size is less than 2048Kb 4. Response must be ok to be processed 5. Header is correct 6. The response body contains the correct data 7. The response body must be in the correct type	As expected	Pass	
TC20	BR20	Highest	Implement PUT Request (Change Playlist Details)	1. Create Pre-request Scripts (unset "Playlist name", unset "description"; Using the "Dynamic variables" to create a playlist name and description) 2. Create Tests for checking the response 3. Send Post request with Postman API		1. Status code is 200 OK 2. Response time is less than 800ms 3. Response size is less than 2048Kb 4. Response must be ok to be processed 5. Header is correct	As expected	Pass	
TC21	BR21	Highest	Implement POST Request (Add Items to Playlist)	1. Create Tests for checking the response 2. Send Get request with Postman API		1. Status code is 201 Created 2. Response time is less than 800ms 3. Response size is less than 2048Kb 4. Response must be ok to be processed 5. Header is correct 6. The response body contains the correct data 7. The response body must be in the correct type	As expected	Pass	
TC22	BR22	Highest	Implement DEL Request (Remove Playlist Items)	1. Create Tests for checking the response 2. Send Get request with Postman API		1. Status code is 201 Created 2. Response time is less than 800ms 3. Response size is less than 2048Kb 4. Response must be ok to be processed 5. Header is correct 6. The response body must be in the correct type	As expected	Pass	
TC23	BR23	Highest	Implement PUT Request (Follow Playlist)	1. Create Tests for checking the response 2. Send Get request with Postman API		1. Status code is 200 OK 2. Response time is less than 800ms 3. Response size is less than 2048Kb 4. Response must be ok to be processed 5. Header is correct	As expected	Pass	
TC24	BR24	Highest	Implement DEL Request (Unfollow Playlist)	1. Create Tests for checking the response 2. Send Get request with Postman API		1. Status code is 200 OK 2. Response time is less than 800ms 3. Response size is less than 2048Kb 4. Response must be ok to be processed 5. Header is correct	As expected	Pass	
TC25	BR25	Highest	Implement GET Request (Get Specific Playlists)	1. Create Tests for checking the response 2. Send Delete request with Postman API		1. Status code is 201 Created 2. Response time is less than 800ms 3. Response size is less than 2048Kb 4. Response must be ok to be processed 5. Header is correct 6. The response body contains the correct data	As expected	Pass	
API Testing (negative Post Requests)									
TC26.1	BR26	Low	Implement Post Request (Create Playlist with invalid name)	1. Create Tests for checking the response 2. Enter test data in the body in the value "name" 3. Send Get request with Postman API	Test data: one	1. Status code is 400 2. Response time is less than 800ms 3. Response size is less than 2048Kb 4. Response must be ok to be processed 5. Header is correct 6. The response body contains the correct data 7. The response body must be in the correct type	The Status code is 400 Bad Request "message": "Error parsing JSON."	Pass	

TC26.2		Low	Implement Post Request (Create Playlist with invalid "public" data)	1. Create Tests for checking the response 2. Enter test data in the body in the value "public" 3. Send Get request with Postman API	Test data: "0"	1. Status code is 400 2. Response time is less than 800ms 3. Response size is less than 2048Kb 4. Response must be ok to be processed 5. Header is correct 6. The response body contains the correct data 7. The response body must be in the correct type	The Status code is 400 Bad Request "message": "Error parsing JSON."	Pass	
TC27.1		Low	Implement Post Request (Change Playlist details with invalid name)	1. Create Tests for checking the response 2. Enter test data in the body in the value "name" 3. Send Get request with Postman API	Test data: -	1. Status code is 400 2. Response time is less than 800ms 3. Response size is less than 2048Kb 4. Response must be ok to be processed 5. Header is correct 6. The response body contains the correct data 7. The response body must be in the correct type	The Status code is 400 Bad Request "message": "Error parsing JSON."	Pass	
TC27.2	BR27	Low	Implement Post Request (Create Playlist with invalid "description" data)	1. Create Tests for checking the response 2. Enter test data in the body in the value "descdescription" 3. Send Get request with Postman API	Test data: 0	1. Status code is 400 2. Response time is less than 800ms 3. Response size is less than 2048Kb 4. Response must be ok to be processed 5. Header is correct 6. The response body contains the correct data 7. The response body must be in the correct type	The Status code is 500 Internal Server Error "message": "Server error."	Fail	Bug report
TC27.3		Low	Implement Post Request (Create Playlist with invalid "public" data)	1. Create Tests for checking the response 2. Enter test data in the body in the value "public" 3. Send Get request with Postman API	Test data: 0	1. Status code is 400 2. Response time is less than 800ms 3. Response size is less than 2048Kb 4. Response must be ok to be processed 5. Header is correct 6. The response body contains the correct data 7. The response body must be in the correct type	The Status code is 400 Bad Request "message": "Boolean value must be either true or false"	Pass	
TC28.1		Low	Implement Post Request (Add items to playlist in wrong format)	1. Create Tests for checking the response 2. Enter test data in the body 3. Send Get request with Postman API	Test data: one	1. Status code is 400 2. Response time is less than 800ms 3. Response size is less than 2048Kb 4. Response must be ok to be processed 5. Header is correct 6. The response body contains the correct data 7. The response body must be in the correct type	The Status code is 400 Bad Request "message": "Error parsing JSON."	Pass	
TC28.2	BR28	Low	Implement Post Request (Add items to playlist with wrong uris data)	1. Create Tests for checking the response 2. Enter test data in the body in the value "uris" 3. Send Get request with Postman API	Test data from environment: {{(wrong_uris)}}	1. Status code is not 200 2. Status code is one of 400s 3. Response time is less than 500ms 4. Response size is less than 1024B 6. Response body doesn't include "id" 7. Response includes "message" as a key equals to "Unexpected value for field firstName" 7. Response is an error and has JSON body	The Status code is 400 Bad Request "message": "Invalid track uri: spotify:track:{{(wrong_uris)}}"	Pass	
TC28.3		Low	Implement Post Request (Add items to playlist with negative number in "position" data)	1. Create Tests for checking the response 2. Enter test data in the body in the value "position" 3. Send Get request with Postman API	Test data: -1	1. Status code is not 200 2. Status code is one of 400s 3. Response time is less than 500ms 4. Response size is less than 1024B 6. Response body doesn't include "id" 7. Response includes "message" as a key equals to "Unexpected value for field firstName" 7. Response is an error and has JSON body	The Status code is 400 Bad Request "message": "Invalid position, must be positive."	Pass	
TC29.1		Low	Implement Post Request (Delete items from playlist with wrong uris data)	1. Create Tests for checking the response 2. Enter test data in the body in the value "uris" 3. Send Get request with Postman API	Test data from environment: {{(wrong_uris)}}	1. Status code is 400 2. Response time is less than 800ms 3. Response size is less than 2048Kb 4. Response must be ok to be processed 5. Header is correct 6. The response body contains the correct data 7. The response body must be in the correct type	The Status code is 400 Bad Request "message": "JSON body contains an invalid track uri: spotify:track:{{(wrong_uris)}}"	Pass	
TC29.2	BR29	Low	Implement Post Request (Delete items from playlist in wrong format)	1. Create Tests for checking the response 2. Enter test data in the body 3. Send Get request with Postman API	Test data: one	1. Status code is not 200 2. Status code is one of 400s 3. Response time is less than 500ms 4. Response size is less than 1024B 6. Response body doesn't include "id" 7. Response includes "message" as a key equals to "Unexpected value for field firstName" 7. Response is an error and has JSON body	The Status code is 400 Bad Request "message": "JSON body doesn't conform to specification"	Pass	
TC29.3		Low	Implement Post Request (Delete items from playlist with empty body data)	1. Create Tests for checking the response 2. Leave the request body empty 3. Send Get request with Postman API		1. Status code is not 200 2. Status code is one of 400s 3. Response time is less than 500ms 4. Response size is less than 1024B 6. Response body doesn't include "id" 7. Response includes "message" as a key equals to "Unexpected value for field firstName" 7. Response is an error and has JSON body	The Status code is 400 Bad Request "message": "JSON body contains a track without URI."	Pass	

TC30.1	Low	Implement Post Request (Follow to playlist with empty body request)	1. Create Tests for checking the response 2. Leave the request body empty 3. Send Get request with Postman API		1. Status code is 400 2. Response time is less than 800ms 3. Response size is less than 2048Kb 4. Response must be ok to be processed 5. Header is correct 6. The response body contains the correct data 7. The response body must be in the correct type	The Status code is 400 Bad Request "message": "Insufficient scope"	Pass
		BR30					
TC30.2	Low	Implement Post Request (Follow to playlist with invalid public data)	1. Create Tests for checking the response 2. Enter test data in the body in the value "public" 3. Send Get request with Postman API	Test data from environment: {{wrong_uris}}	1. Status code is not 200 2. Status code is one of 400s 3. Response time is less than 500ms 4. Response size is less than 1024B 6. Response body doesn't include "id" 7. Response includes "message" as a key equals to "Unexpected value for field firstName" 7. Response is an error and has JSON body	The Status code is 400 Bad Request "message": "JSON body does not adhere to the defined endpoint parameters"	Pass