Contents

Task-1

Task description

As a tester, you're assigned to test Sign Up / Log In flow on XYZ.com platform (https://www.XYZ.com). Please list out all the possibilities after careful review the web site functionality.

Explanation of task execution

Below is my personal investigation of the ***XYZ.com*** platform.

I found the Sign Up / Log In flow and described it on the diagram.

Based on created diagram I discovered different test scenarios and then created a list of test cases based on discovered test scenarios.

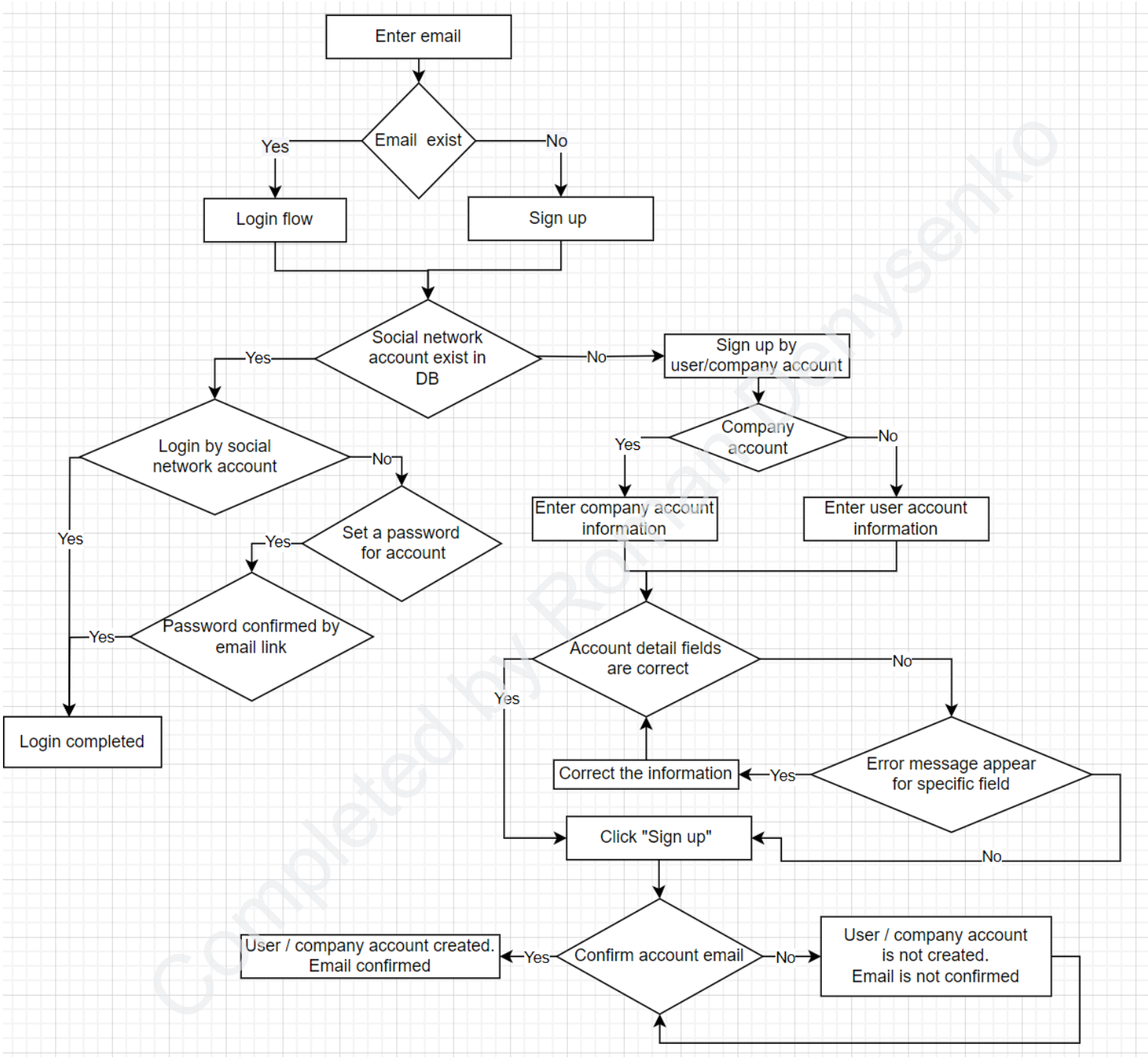
Below is a list of high priority tests that I would like to use during app development phases and regression testing. Also, an additional test that can be written based on approved approach for the QA team. Those test cases can have medium or low priority for the test runs.

I added some notes for test cases and want to clarify that I skipped some descriptions and expected result. I agree that they should have more precise description, but I decided to describe my point of view and focus on specific parts or elements that will show how and why I'm working with test documentation in this way.

In case when we are talking about end-to-end tests, I ignored minor actions and did not describe or maybe missed some important things that I did not understand properly during emu exploratory testing.

End-to-end scenario

Below is a list of end-to-end scenarios for checking **Sign-Up flow**.



1. **Sign-Up flow for users with an email address.**

   A. Valid Sign-Up flow.

   The user fills all fields on the Sign-Up form, using valid data, provides a valid password that will correspond to describe security requirements, and accepts and confirms the account by the activation link that will be sent to the user email box. More information regarding the tests you will find in the table of test cases below.

   B. Invalid Sign-Up flow.

   The user will not fill all fields on the Sign-Up form, or skip something or using invalid data, or an invalid password that will not correspond to security requirements or confirm the account by the activation link that will be sent.
   More information regarding the tests you will find in the table of test cases below.

NOTES

I expect that the activation link user will receive only when the Sign-Up form will be filled out properly and the app will send the request for account creation to the email that was used for account creation.

2. **Sign-Up flow for users with a social network account (existing accounts).**

   a. **Valid Sign-Up flow.**
   The user selects a social network account (existing accounts) on the Sign-Up form, then confirms the access to the social network account data.

   Note -1.
   I expected that 3rd party services granting access to the web app, then redirects the user to the app home page or whatever)
   More information regarding the tests you will find

   b. **Invalid Sign-Up flow.**
   The user selects a social network account (existing accounts) on the Sign-Up form, then terminates the registration process, and will not complete the required actions for platform account creation.
   More information regarding the tests you will find in the table - 1.

3. If a user has an account and should complete the authentication procedure by providing a user login and password. It should work when user account created without any available social network identification options. If a user was created by social network options

Table - 1

Test case name	Flow name	TC index	TC steps	Comments
User registered a new **user account** with valid data by email.	Sign-Up (by email)	TC-1	1. Enter valid email (email should not exist in the web app DB) on the Sign-Up / Login form. 2. Click the “Next” button. 3. Don’t check the “Company account” checkbox. 4. Enter valid user details and password. 5. Click the “Sign-Up” button. 6. Confirm account creation by email link. 7. Login by using email and provided password.	This TC is possible to replicate and create several different test cases with more precise actions when the user can cancel some actions like declining the activation email link. Some TCs I added below, some are possible to discuss with the team and implement more specific test scenarios.
User registered a new **company account** with valid data by email.	Sign-Up (by email)	TC-2	1. Enter valid email (email should not exist in the web app DB) on the Sign-Up / Login form. 2. Click the “Next” button. 3. Check up the “Company account” checkbox. 4. Enter valid company details and password. 5. Click the “Sign-Up” button. 6. Confirm account creation by email link. 7. Login by used email and provided password.	

User registered a new **user account** with invalid data by email.	Sign-Up (by email)	TC-3	1. Enter valid email (email should not exist in the web app DB) on the Sign-Up / Login form. 2. Click the “Next” button. 3. Don’t check the “Company account” checkbox. 4. Enter **invalid** user account details, names and password. 5. Click the “Sign-Up” button. 6. Check that the web form is showing error notifications. 7. Login by used email and provided password and be sure that user was not logged to the platform	This TC is also split on a few different TC that will check specific error gathering via entering invalid data. Here it is possible to use different test techniques for checking behavior of the fields, e.g., boundary value analysis and equivalence partitioning. **Note.** 7th step I added for this test because the web app simultaneously can send and save user credentials and will allow me to create an account. When this action happens, we will have a system bug.
User registered a new **company account** with invalid data by email.	Sign-Up (by email)	TC-4	1. Enter valid email (email should not exist in the web app DB) on the Sign-Up / Login form. 2. Click the “Next” button. 3. Check up the “Company account” checkbox. 4. Enter invalid account details, names and password. 5. Click “Sign-Up” button. 6. Check that web form is showing error notifications. 7. Login by used email and provided password and be sure that user was not logged to the platform.	
User declined email confirmation after registering new **user account**	Sign-Up (by email)	TC-5	1. Enter valid email (email should not be exist in the web app DB) on the Sign-Up / Login form. 2. Click “Next” button. 3. Don’t check up the “Company account” checkbox. 4. Enter valid user details, names and	

			password. 5. Click the “Sign-Up” button. 6. Do not confirm email registration. Do not confirm account creation email. 7. Login by using email and provided password and be sure that the user was not logged to the platform.	
User declined email confirmation after registering new **company account**	Sign-Up (by email)	TC-6	1. Enter valid email (email should not exist in the web app DB) on the Sign-Up / Login form. 2. Click the “Next” button. 3. Check up the “Company account” checkbox. 4. Enter valid company details, names and password. 5. Click the “Sign-Up” button. 6. Do not confirm email registration. Do not confirm account creation email. 7. Login by using email and provided password and be sure that the user was not logged to the platform.	
User registered new **user account** by social network account.	Sign-Up (by social network account)	TC-7	1. Enter valid email (email should not exist in the web app DB) on the Sign-Up / Login form. 2. Click the “Next” button. 3. Don’t check the “Company account” checkbox. 4. Click the “Facebook” button and grant access for the web app. 5. Logout from the platform. 6. Login using the same option. 7. Repeat steps 4-6 using LinkedIn and Google options	In case when a user has linked email with a social network account this TC should be cloned and edited according to existing Sign-Up/Login flow of the web app. Also it can be possible to create additional TC when a user registered a web app account directly on the Welcome popup. Auth flow for the 3d party services can be tested independently also, and it’s a source for the additional TC. I found that Google auth option was integrated with some issues. Please check the “Discovered Issues” section. Some TC I added below, some is possible to discuss

				with the team and implement more specific tests as well later.
User completed the reset of password for existing **user account** / **company account**	Sign-Up (by email)	TC-8	1. Enter valid email (email should exist in the web app DB) on the Sign-Up / Login form. 2. Click “Next” button. 3. Click on the “Forgot password” link. 4. Complete actions for resetting the password that described in the received email. 5. Login by new credentials. 6. Repeat steps 1-5 for company account	In case when “Reset password” feature should be tested independently and more deeply this TC can be used as a reference and all steps for resetting the password should be defined as unique steps for the verification.
Login on the platform by existing (not blocked) user / company account		TC-9	TBD	Login tests for the user or company accounts should test the flow when user is using valid credentials for log in on the platform and logged out from it. Those tests can be a part of test cases that created for Sing in flow or created as independent flow.
Log out /Sign out tests		TC-10		I did not write any tests for the log / sign out on the platform. Those tests will check the behavior of the web site when user would like to logging /signing out from the platform and can be added as an additional steps for the test cases from this list

Here is a short description that can help to understand exactly what I was trying to test and what should be tested deeper.

4. More detailed testing expect to test the Sign-Up form, all fields on it and all related features that are involved in the process of account creation, e.g. “Forgot password”, “Set a password for your account”, account activation by email link, etc.
5. I found that the web app has several text fields and logic for filling the password form. In this case I can implement an approach for creating tables with test data for using it during testing this form independently, as part of security testing as well.

Task-2

Task description

1. We are asking your opinion for what kind of testing you will recommend for Sign Up / Log In flow on the platform. Please provide the list of various testing types which you will recommend and justify them. Explain with examples about why did you pick those testing types?

Explanation of task execution

First of all, I would like to note that when I start to test the platform I'm using a basic approach, when the tester has no requirements, or it's not completed yet.

I will divide all my tests into two groups, functional and non-functional. For each approach, I will use different browsers, e.g. Chrome, Firefox, Safari, devices: mobile phones, tablets, and laptops that are launched on different OS, e.g. windows, macOS, Android and iOS (for mobile testing).
If I have no limits and comprehensive documentation for testing, I will start from functional testing of the platform using a defined list of devices.

Below you will find the list of test types and short description that I would like to implement.

1. **Functional tests**. Please check the list of test cases in the section -1
2. **End-to-end tests**. Please check the list of test cases in the section -1
3. **Integration tests**. Here is possible to test the platform integration with 3rd party services, e.g., Google or Facebook. Please check the related section below.
4. **UI / UX testing**. Please check the related section below.
5. **Accessibility testing**. Please check the related section below.
6. **Security testing**. Please check the related section below.
7. **Performance testing**. Please check the related section below.

UI / UX testing

Checking the web site components, UI, navigation elements, controls, etc. on different devices, browsers, OS's, screen resolutions and screen orientations. Here is possible use different test techniques and approaches.

Pairwise testing - for checking app behavior on different browsers, OS's, devices and implement these tests using any available resources. Executing this test manually or using online services.

Decision table testing - for checking behavior of the forms.

Cross-browser testing - for checking behavior of the Sign Up / Log In flow in different browsers.

Main scenarios for UI / UX tests

1. Verify that Sign Up / Log In page opens.
2. Verify that all UI elements and components are rendered without any glitches or visual artifacts. Check and compare the behavior on accepted devices. All devices are showing the same design and all elements work as designed during testing Sign Up / Log In flows.
3. Verify that the user can use navigation options that are available the for current device.
4. There is possible to navigate or access the different controls by pressing the 'Tab' key on the keyboard.
5. Check if the password is in masked form when typed in the password field.
6. Show / hide button is working (This option is not available on the web site)
7. Web app behavior when username and/or password are invalid.
8. Copy-pasting option for the password should not work or work, if it was approved.
9. Verify that the user is able to login by entering valid credentials and clicking on the "Sign in" or "Login with…" button.
10. Validation message (error notifications) are displaying for all mandatory fields. It should work for all mandatory fields. For some reason they are not marked as mandatory (asterisk are missing)
11. Check that the validation message is displayed when user exceeds the character limit of the characters. Right now, web platform forms have no limits. It can confuse a little when you are entering long password, for instance, and can't handle how many symbols you are entered already.

12. Verify if there is a checkbox with the label “Keep me logged in” on the Welcome page (Sign-Up flow).
13. Checking that closing the browser should not log-out an authenticated user. Launching the application should lead the user to login state only.

Accessibility testing

Web site should be tested for accessing for the people with disabilities. Main focus should be done using WCAG documentation.

- Checking the web site behavior using any available online services or tool.
- Complete basic accessibility testing by manual testing the Sign-Up / Login form:
    - Check color contrast.
    - Check keyboard friendly behavior (not only TAB button).
    - Zoom option without missing the context.
    - Voice assistance. Ability to “read” the web site content. Using the screen readers for more simple testing.

Security Test Cases for Sign-Up / Login Page

1. Check the login session timeout.
2. Limit on the total number of unsuccessful login attempts
3. Check that web app shows incorrect credentials message.
4. Check XSS vulnerability vectors.

There is possible to use Postman for playing with an API request, Fiddler (Charles) for intercepting API request manually or OWASP ZAP for more precise actions in automation mode.

Performance test cases for Sign-Up / Login Page

1. Create a script for sending multiple Sign-Up / Login requests simultaneously. There is possible to use any available services or applications, e.g., JMeter, K6 or something different.
2. Checking the web app behavior by interrupting/ blocking API request can show weak places of the web app. Connection throttling is also very productive in this case. Connection throttling is also very productive in this case. I found that the web app does not handle blocked API requests and did not show any service notifications.
3. Measuring the authentication time for Sign-Up / Login (Sign out / Log out) flows.
4. Checking Sign-Up / Login (Sign out / Log out) app behavior from different locations. In some cases, app can work very slowly in one location and without any delays for another. For this I will use any available online services like VPN or any cloud solutions, like a deployed virtual machine on servers that are in referent countries.

Regarding to test techniques I will use several reliable approaches for decreasing number of tests and volume of test data. It should be:

- “Decision table testing”.
- “Boundary values testing” - testing the behavior of web forms using set of defined values and decrease number of tests.
- “Pairwise testing”. It can help to check the web form mixing the filing different fields in different orders that can help decrease number of tests.

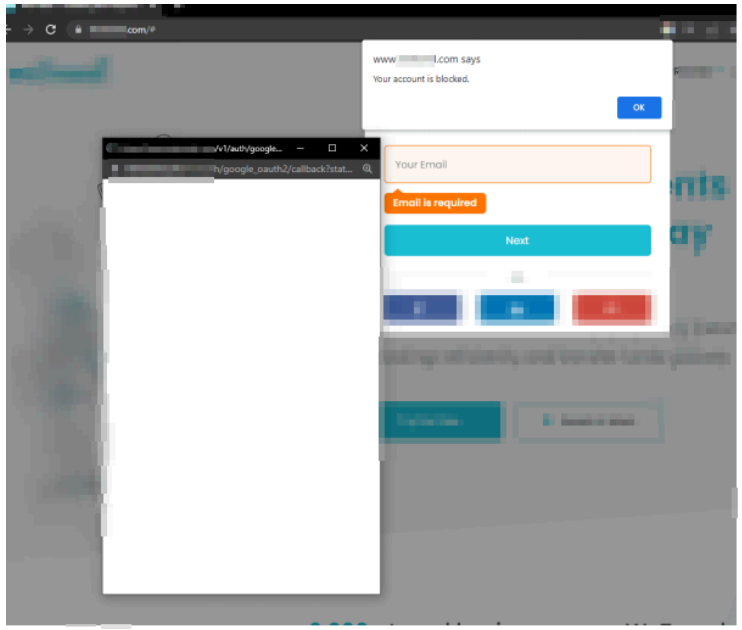
Discovered issues.

Google Sign-Up. Your account is blocked popup is hidden.

After choosing the Google sign in option, the user can’t see ‘Your account blocked’ popup on the same browser page. For viewing the popup user should change the focus on different web page.

Browser - Chrome

OS: MacOS v12.6



Account is blocked after completing the reset password process.

When I created new user account via Sign up form and decided to reset password via “forgot password” link I discovered that after successful actions my account was blocked.

I assume that it can be related to the app security or issues on DB level because if service side can’t handle the reset request properly or reset link was not stored properly or auth key was missing it can make some magic and user account can be blocked.

Below is a picture that I was seeing during my test.