**Question 1**

Based on your experience as a QA.

1. **Explain in about SDLC, STLC and their differences**

SDLC : Software Development Life Cycle, is a process used by the software industry to produce high-quality software system which help to meets the customer expectations. Phase of the SDLC are Requirement > Analysis and Design > Development > Testing > Deployment > Maintenance.

STLC : Software Testing Life Cycle is mainly related to software testing. The activities performed by the testing team to ensure the quality of the software. STLC is part of Software Development Life Cycle. The phase of Software Testing are Test Planning > Test Case Development > Test Environment Setup > Test Execution > Test Closure.

Difference :

SDLC is related to software development whereas STLC is related to software testing.

SDLC defines all the standard phases which are involved during the software development process, whereas the STLC process defines various activities to improve the quality of the product.

The SDLC life cycle helps a team to complete successful development of the software while STLC phases only cover software testing.

SDLC helps developing good quality software whereas STLC helps in making the software defect free.

1. **Explain the steps for Bug Cycle?**

- New, when tester found new bug and it posted for the first time.

- Assigned, the bug that posted by tester and approves by the lead tester will be assigne to the developer team.

- Open, The developer team starts to analizing the bug and works on to be fix. If the defect is not appropriate, the developer team can transfer the defect to these four states based on specific reasons Duplicate, Rejected, Deferred and not a bug.

- Fixed, When the developer team finish the task of fixing a defect and verifies the change then can mark the status of the defect as “Fixed”.

- Pending Retest, After defect fixed, developer team will assigne back to the tester to retest the defect. Until the tester working on retesting the defect, the status of the defect still remain in ‘Pending Retest”.

- Retest, The tester start the task to retesting the defect to verify if the defect is solved as the requirement or not.

- Reopen, if the defect still persist and the tester will assigne back to the developer team to do the fixing and mark the defect status as “Reopen”.

- Verified, If there is no issue in the defect and it was fixed properly by the developer then the state of the defect is changed and marked as verified.

- Closed, When a defect does not exist and is well tested and verified properly then it ends at a closed state. Tester changes the state of the defect.

**c. Give us Real Case about Severity and Priority**

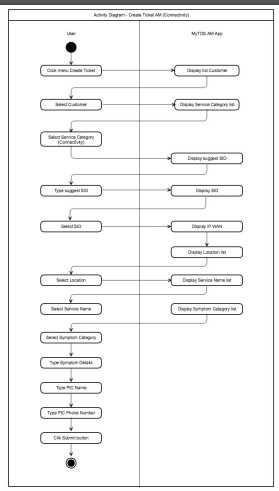
Priority is when the issue is impact on bussiness side. There are 4 priority categories critical, high, medium and low priority. Real case for critical priority issue is when the application is crash or user can’t login to the application because the server is down and it’s impacted to all user.

Severity is when the issue is impacted to application functionality. There are also 4 severity categories critical, high, medium, and low. Real case for critical severity issue is when user input correct username and password to login but after click button login, the application didn’t response or stuck in the same page.

**d. Give us explanation about Equivalence Partitioning & Boundary Value Analysis testing techniques**

Equivalence Partiotioning is a black box testing technique, input data units are divided into equivalend partition that can reduces time required for testing. Pick only one value from each partition for testing.

Boundary Value Analysis is the process of testing between extreme ends or boundaries between partition of the input value. Check the boundary values of each partition.

**Question 2**

as a QA, how to make a test scenario (Negative and positive cases) based on the activity diagram on the side?

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case : Create Ticket AM (Connectivity) | | |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| No. | Scenario | Test Case | Step Number | Test Step | Expected Result | Type |
| 1. | Create Ticket AM (Connectivity) | Create Ticket AM for Registered Customer | 1 | User Open My TDS AM App | Display dashboard My TDS App | Positive |
| 2 | User Click Menu Create Ticket | Display List Customer |
| 3 | User Select Customer | Display Service Category List |
| 4 | User Select Service Category (Connectivity) | Display Suggest SID |
| 5 | User Type Suggest SID | Display SID |
| 6 | User Select SID | Display IP WAN and Display Location List |
| 7 | User Select Location | Display Service Name List and Symptom Category List |
| 8 | User Select Service Name | Service Name selected |
| 9 | User Select Symptom Category | Symptom Category Selected |
| 10 | User Type Symptom Details | Details Symptom updated |
| 11 | User Type PIC Name | PIC Name updated |
| 12 | User Type Phone Number | Phone Number Updated |
| 13 | User Click Submit Button | Ticket successfully created |
| 2. | Create Ticket AM (Connectivity) | Create Ticket AM for Non-Registered Customer or Customer Data Not Found | 1 | User Open My TDS AM App | Display dashboard My TDS App | Negative |
| 2 | User Click Menu Create Ticket | Display List Customer |
| 3 | User Type Customer Name | Display Message "Data Customer Not Found, Please Check Again" |
| 3. | Create Ticket AM (Connectivity) | Create Ticket AM and SID Not Found | 1 | User Open My TDS AM App | Display dashboard My TDS App |
| 2 | User Click Menu Create Ticket | Display List Customer |
| 3 | User Select Customer | Display Service Category List |
| 4 | User Select Service Category (Connectivity) | Display Suggest SID |
| 5 | User Type Suggest SID | Display Message "SID Not Found, Please Check Again" |
| 4. | Create Ticket AM (Connectivity) | Create Ticket AM and User Not Input Symptom Details | 1 | User Open My TDS AM App | Display dashboard My TDS App | Negative |
| 2 | User Click Menu Create Ticket | Display List Customer |
| 3 | User Select Customer | Display Service Category List |
| 4 | User Select Service Category (Connectivity) | Display Suggest SID |
| 5 | User Type Suggest SID | Display SID |
| 6 | User Select SID | Display IP WAN and Display Location List |
| 7 | User Select Location | Display Service Name List and Symptom Category List |
| 8 | User Select Service Name | Service Name selected |
| 9 | User Select Symptom Category | Symptom Category Selected |
| 10 | User Not Input Symptom Details | Details Symptom Empty |
| 11 | User Type PIC Name | PIC Name updated |
| 12 | User Type Phone Number | Phone Number Updated |
| 13 | User Click Submit Button | Display Error Message "Symptom Details Can Not be Empty" |
| 5. | Create Ticket AM (Connectivity) | Create Ticket AM and User Not Input PIC Name | 1 | User Open My TDS AM App | Display dashboard My TDS App | Negative |
| 2 | User Click Menu Create Ticket | Display List Customer |
| 3 | User Select Customer | Display Service Category List |
| 4 | User Select Service Category (Connectivity) | Display Suggest SID |
| 5 | User Type Suggest SID | Display SID |
| 6 | User Select SID | Display IP WAN and Display Location List |
| 7 | User Select Location | Display Service Name List and Symptom Category List |
| 8 | User Select Service Name | Service Name selected |
| 9 | User Select Symptom Category | Symptom Category Selected |
| 10 | User Type Symptom Details | Details Symptom updated |
| 11 | User Not Input PIC Name | PIC Name Empty |
| 12 | User Type Phone Number | Phone Number Updated |
| 13 | User Click Submit Button | Display Error Message "PIC Name Can Not be Empty" |
| 6. | Create Ticket AM (Connectivity) | Create Ticket AM and User Not Input Phone Number | 1 | User Open My TDS AM App | Display dashboard My TDS App | Negative |
| 2 | User Click Menu Create Ticket | Display List Customer |
| 3 | User Select Customer | Display Service Category List |
| 4 | User Select Service Category (Connectivity) | Display Suggest SID |
| 5 | User Type Suggest SID | Display SID |
| 6 | User Select SID | Display IP WAN and Display Location List |
| 7 | User Select Location | Display Service Name List and Symptom Category List |
| 8 | User Select Service Name | Service Name selected |
| 9 | User Select Symptom Category | Symptom Category Selected |
| 10 | User Type Symptom Details | Details Symptom updated |
| 11 | User Type PIC Name | PIC Name updated |
| 12 | User Not Input Phone Number | Phone Number Empty |
| 13 | User Click Submit Button | Display Error Message "Phone Number Can Not be Empty" |

**Question 3**

You are testing a new version of software for a coffee machine. The machine can prepare different types of coffee based on four categories. i.e., coffee size, sugar, milk, and syrup.

The criteria are as follows:

- Coffee Temperature should be about 78%

- Coffee size (small, medium, large)

- Sugar (none, 1 unit, 2 units, 3 units, 4 units)

- Milk (yes or no)

- Coffee flavor syrup (no syrup, caramel, hazelnut, vanilla)

- Sugar (3 and 4 Units) cannot add Coffee flavor syrup

And now we have some issues:

- You can add Coffee flavor syrup when sugar is 3 Units

- The coffee temperature is below 78 %

QUESTION: As a QA:

1. Describe how a test scenario can ensure that all test coverage is covered

The Test Scenario can be defined as a metric, measure the amount of the testing perfomed by a set of test and reduce the product repetition and complexity.

Scenario 1 : Make a Coffee with Small Coffee Size, Sugar 1 Unit with Milk and Coffee Flavor Syrup Hazelnut

Scenario 2 : Make a Coffee with Medium Coffee Size, Sugar 3 Units without Milk, No Coffee Flavor Syrup Added

Scenario 3 : Make a Coffee with Large Coffee Size, none of Sugar with Milk, Coffee Flavor Syrup Vanilla.

B. How do you identification issue and write defect report from the following information?

Issue 1

Defect Number : Defect-1

Type : Bug

Summary : Can Add Coffee Flavour Syrup when Sugar is 3 Units

Description : The coffee machine can add coffee flavor syrup when the sugar already added 3 units

Steps:

* Make a coffee with Coffee Size (Small, Medium or Large)
* Add Sugar 3 Units
* Add Milk (Yes or No) \*optional
* Coffee Flavour Syrup added

Expected : With Sugar 3 and 4 units, Coffee Flavour Syrup should not be added

Severity : High,

Priority : Critical

Issue 2

Defect Number : Defect-2

Summary : The Coffee Temperature is below 78%

Description : The Coffee Machine produce the coffee and the time for preparing is too long so the temperature became less than 78%

Steps :

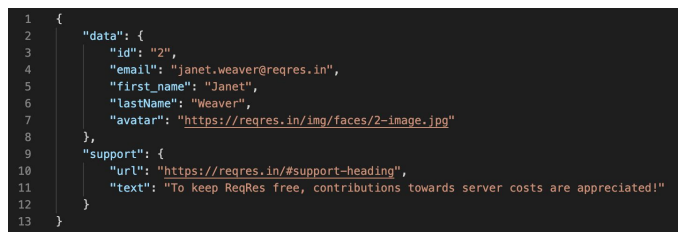
* Make a coffee with Coffee Size (Small, Medium or Large)
* Add Sugar 3 Units
* Add Milk (Yes or No) \*optional
* Coffe Temperature is below 78%

Expected : The Coffee Temperature should be about 78%

Severity : Medium

Priority : Medium

**Question 4**

Mock API:  Real API: https://reqres.in/api/users/2

QUESTION a. As a QA, Please compare between MockAPI and Real API. Is there any difference?explain it b. If there are differences, what defects will appear?

There are no different between MockAPI and Real API. The Data has the same output.

id:2,

email: janet.weaver@reqres.in,

first\_name: Janet,

last\_name: Weaver,

avatar":<https://reqres.in/img/faces/2-image.jpg>

url: https://reqres.in/#support-heading,

text: To keep ReqRes free, contributions towards server costs are appreciated!

**Question 5**

Create 5 Scenario Test from https://demo.midtrans.com/ with Gherkin Language and attach the results of the test report

Feature Shopping Pillow

As a visitor of the ecommerce website

I want to buy Pillow

Scenario Buy the Product

Given the user is already choose pillow product

Given the user is already in pillow product page

Given the user does not registered

When the user click ‘Buy Now’ button

Then the user should be direct to Checkout page



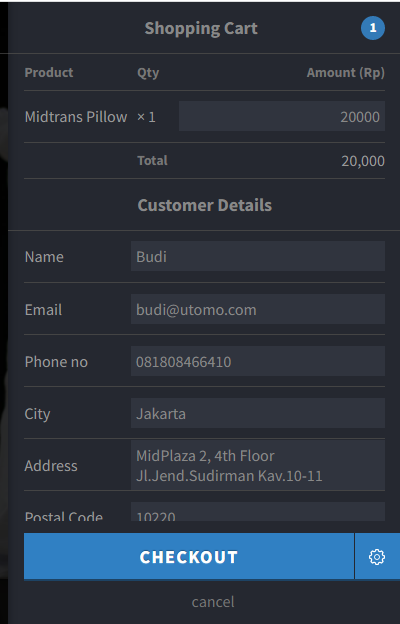
Scenario Checkout Product

Given the user is already in Checkout Page

When the user input all field correctly

And the user click ‘Checkout’ button

Then the user should be direct to Order Summary



Scenario Order Summary

Given the user already in the order summary page

When the user check ‘Amount’

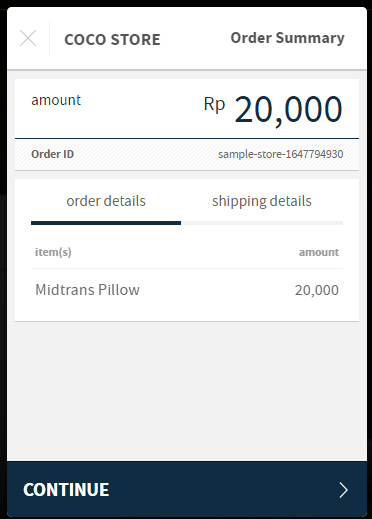
And the user check Order Details

And the user check shipping details

And the user click ‘Continue’ button

Then the user should be direct to Payment Method

And the user must see the list of Payment Method

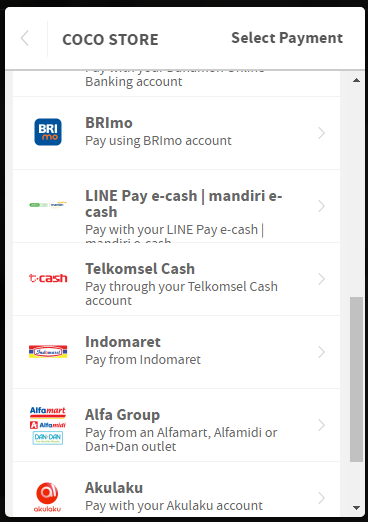


Scenario Select Payment Method

Given the user already in the Payment Method List

When the user choose one of the payment method

Then the user should be direct to Confirmation Payment Page



Scenario Payment Confirmation Method

Given the user already in the Payment Confirmation Method

When the user check Payment Method

And the user check ‘Amount’

And the user click ‘Pay Now’ button

And the user success to Pay

Then the user will direct to Successful Page

