Question 1

a. SDLC (Software development Life Cycle) is the application of standard business practices to building software applications that consist of six to eight steps: Requirement gathering, Design, Implementation, Testing, Deployment and Maintenance.

STLC (Software Testing Life Cycle) is a process used to test software and ensure that quality standards are met which consists of 6 phases: Requirement analysis, test planning, test case development, test environment setup, test execution, and test cycle closure.

Differences:

Parameter	**SDLC**	**STLC**
Origin	Development Life Cycle	Testing Life Cycle
Objective	To complete successful development of the software including testing and other phases.	The only objective is testing.
Requirement Gathering	The business analyst gathers the requirements and create a development plan.	The QA team analyse requirement documents like functional and non-functional documents and create System Test Plan
High & Low-Level Design	The development team creates the high and low-level design plans	The test analyst creates the Integration Test Plan
Coding	The real code is developed, and actual work takes place as per the design documents.	The testing team prepares the test environment and executes them
Maintenance	SDLC phase also includes post-deployment supports and updates.	Testers, execute regression suits, usually automation scripts to check maintenance code deployed.

b. Bug Cycle Steps:

- **New:** When a new defect is logged and posted for the first time. It is assigned a status as NEW.
- **Assigned:** Once the bug is posted by the tester, the lead of the tester approves the bug and assigns the bug to the developer team
- **Open**: The developer starts analysing and works on the defect fix
- **Fixed**: When a developer makes a necessary code change and verifies the change, he or she can make bug status as “Fixed.”
- **Pending retest**: Once the defect is fixed the developer gives a particular code for retesting the code to the tester. Since the software testing remains pending from the testers end, the status assigned is “pending retest.”
- **Retest**: Tester does the retesting of the code at this stage to check whether the defect is fixed by the developer or not and changes the status to “Re-test.”
- **Verified**: The tester re-tests the bug after it got fixed by the developer. If there is no bug detected in the software, then the bug is fixed and the status assigned is “verified.”
- **Reopen**: If the bug persists even after the developer has fixed the bug, the tester changes the status to “reopened”. Once again the bug goes through the life cycle.
- **Closed**: If the bug is no longer exists then tester assigns the status “Closed.”
- **Duplicate**: If the defect is repeated twice or the defect corresponds to the same concept of the bug, the status is changed to “duplicate.”
- **Rejected**: If the developer feels the defect is not a genuine defect then it changes the defect to “rejected.”

- **Deferred**: If the present bug is not of a prime priority and if it is expected to get fixed in the next release, then status "Deferred" is assigned to such bugs
- **Not a bug**: If it does not affect the functionality of the application then the status assigned to a bug is "Not a bug".

c. Real Case about Severity and Priority:

- **A very low severity with a high priority:** A logo error for any shipment website, can be of low severity as it not going to affect the functionality of the website but can be of high priority as you don't want any further shipment to proceed with the wrong logo.

- **A very high severity with a low priority:** Likewise, for flight operating website, a defect in reservation functionality may be of high severity but can be a low priority as it can be scheduled to release in a next cycle.

d. Equivalence Partitioning is a type of black box testing technique in which the input data units are divided into equivalent partitions that can be used to derive test cases to reduces the time required for testing because of the small number of test cases.

Boundary Value Analysis testing is used to identify errors at boundaries rather than finding those that exist in the centre of the input domain.

Question 2

Scenario: Create a Ticket

Positive Case

1. As a user, I want to create a ticket AM successful.
2. Check system behavior when the user cancels to book a ticket AM.
3. As a user, I want to check the service provided at a certain location.
4. Validate whether the symptom category list appears when selecting the service name.
5. Validate whether the location list appears after IP WAN.
6. As a user, I want to edit the previous selection data.

Negative Case

1. User enters an invalid character when inputting PIC name.
2. User inputs invalid phone number format.
3. User inputs the invalid suggest SID.
4. User skips selecting the service name.
5. User does not input the symptom detail.

Question 3

A. A test scenario can ensure that all test is covered by listing all possible positive and negative cases.
B. Defect Report:

1. Title: Issue on choosing coffee flavor syrup

- Description: The coffee flavor syrup can be added only when the sugar is 3 units.
- Actual result: User not able to add coffee flavor syrup when choosing other than 3 units of sugar.
- Expected result: User able to choose any coffee flavor syrup when choosing any units of sugar.

2. Title: Issue on coffee temperature

- Description: The coffee temperature does not the criteria; below than 78%.
- Actual result: User is able to make a coffee when the temperature is below than 78%.
- Expected result: The coffee machine will hold on a few second until reach temperature more than 78% before the user able to proceed to other step.

Question 4

a. Mock APIs provide predefined API responses for client applications which involves specifying endpoint routes (e.g., GET /api/status) as well as response behaviors (e.g., response payload and status code). It typically comes with a server implementation that allows them to easily run in a local environment. While, Real API's is using the dynamic data and already develop.

b. Some defects will appear such as functionality bugs, reliability bugs, performance bugs and security bugs.

Question 5

Scenario	TCID	Test Case	Step	Test Data	Expected Result	Actual Result	Status
Purchase Item	TC01	User wants to purchase midtrans pillow successfully	**Given** Launch the application and click buy now **When** Fill the shopping cart form **And** Select payment method **Then** Purchase successful	Email : nurul.qawork@gmail.com Card Number: 4811111111111114 Expire Date: 01/25 CVV:123 OTP:112233	The purchase successfully created	The purchase successfully created	Passed
Payment Method	TC02	User accidently enter invalid card detail	**Given** Launch the application and click buy now **When** Fill the shopping cart form **And** Enter invalid card credit data for payment **Then** Purchase unsuccessful	Email : nurul.qawork@gmail.com Card Number: 4921111111111114 Expire Date: 01/25 CVV:123 OTP:112233	The purchase successfully created and error message pop up	The purchase successfully created and error message pop up	Passed
Homepage Icon	TC03	User wants to look the pillow design photo	**Given** Launch the application **When** Click the transition button **Then** The photo slide to left or right	-	The photo is able to slide successfully.	The photo is able to slide successfully.	Passed
Sign Up	TC04	User want to register new account	**Given** Launch the application and click sign up button **When** Fill the register form **Then** Account successfully register	Business name: qa Full name: nurul Business Email: nurul.qawork@gmail.com Phone number: 8131111111 Password: Nurul123% Confirm Password: Nurul123%	The new account is successfully created.	The new account is successfully created.	Passed
Log In	TC05	User want to login successfully	**Given** Launch the application and click sign up button **When** Click log in hyperlink **And** Input username and password **Then** User go to dashboard	Username: nurul.qawork@gmail.com Password: Sarang3%	User login successfully and go to dashboard page.	User login successfully and go to dashboard page.	Passed