Daniel Panturas
BSCS601

Answer the following questions. Provide adequate explanation and support for your answers.

1. Role of an End-User: As a tester, we always do not test the application product but play the role of an end-user. As an end-user, how would you like to modify the ATM (Automated Teller Machine) so that it can be more beneficial?

   - As a person who always use atm it would be beneficial if their features are up to date with current technology. Also, it would be great if it has strong security and ease of access. For example, having a touchscreen technology with a simple and user-friendly interface would help users navigate easily, especially to those who aren't that great at using any digital devices. Based on Aratek (2023), integrating biometrics into ATMs significantly enhances security in today's banking landscape. The traditional pin method is prone to fraud. These biometric identifiers are inherently difficult to replicate or steal, making unauthorized access exceedingly challenging. Since biometrics has become more recent especially in e-wallet it would also be great to provide additional line of security to prevent any malicious thing from happening. Lastly, based on my experience it would be beneficial if I can add an option about my withdrawal like having to choose the bill I want (100, 500, or 1000).

2. Testing a Login Page: You are tasked with testing the login functionality of a web application. What are the critical test scenarios you would cover?

   - Login functionality features varied depending on what system it would be use, but were going to mention critical test scenario for general login function here. Things like user-friendly ui, security, efficiency, and usability are needed here

   - **Valid Credentials Test –** If the user credentials are valid like correct password and email, then a notification will pop up about their successful login and will be redirect to the page only login user have access.
   - **Invalid Credentials Test –** If user credentials is not valid an error would pop up either their email or password is invalid or wrong. A try again word will also pop up to notify them properly.
   - **Blank Fields Test –** If upon logging in one or more required field are blank it would show in the interface that, that certain is blank and is required to have a value before proceeding to log in.
   - **Session Management Scenario –** If the user is already login, a test for session will be conducted. User will only logout if the session has ended or the user intended to log out by clicking the logout button. Tester needs to verify that both the button and session time is working properly.

- **Security Test –** One of the most important test is this, this where the tester ensure that various security measurement is working properly. Like for example preventing SQL injection from happening, limiting log in attempts If the user log in too much and still wrong and also ensuring that password is hidden by making the visible password as dot.
- **Forgot and Remember Scenario –** This two ensure convenience and hope for the user, this scenario test involves when user forgot their password and keep trying on accessing it, forgot password would help user to recover it by sending email to their gmail and changing their password. Remember me on the other hand help in remembering the credentials of user for eases of log in.

3. Managing Changing Requirements: How do you handle frequent changes in requirements during the testing phase?

   - First thing is to prioritize the collaboration with your stakeholders, having a great communication with them would help lessen ambiguous requirement making the testing and developing phase much clearer. After that I would ensure that I have a develop a flexible and modular cases this ensure a dynamic approach in testing the change of requirements which could help lessening the workload.
   According to Rachlin (2024), having a flexible test case could build a five-star product, even if something changes on the requirements it would be easy to fix just by using the develop test again and to only change the affected part of program. Having this is beneficial to lessen the cost of production too, additionally we need to have a proper tracing wherein I would link the test into the requirement so I can easily locate which part I should update once the requirements has been changes. Lastly, I prioritize re-testing and regression testing to ensure that changes didn't break existing features.

4. Cross-Browser Testing: You need to ensure that a web application works seamlessly across multiple browsers. How would you approach this?

   - To test that we can open our web application across multiple browsers especially to this browser that has wide range of user. However, we need to consider various things first like Identifying which browser is popular and what devices your target devices. Additionally, to have a proper test we need to use the analytics of data of which browser is the most use and what version. After that we can proceed on checking the contents of your program to your chosen browser, checking the if the layout of your interface is not distorted and working properly, does the design still work and no issue occur like incompatibility with java script or any given technology you use. Additionally, to speed things up we can use tools for cross browser testing

5. Testing a Payment Gateway Integration: How would you test a payment gateway integration for an e-commerce application?

   - Testing a payment gateway is important to make sure customers can pay safely and easily. First, you need to use a sandbox or test mode provided by the payment company. This lets you try different payment scenarios without using real money.

   Next, test if payments work correctly:

   - Try a successful payment using test card details.
   - Try a failed payment with wrong or expired card info.
   - Test what happens if a user cancels or loses connection during payment.
   - Check if the system prevents duplicate payments (like when someone clicks "Pay" twice).

   - Also, make sure the order updates only when the payment goes through. The app should send a receipt or message after payment, and all payment info should be saved correctly. Security is also very important. Finally, test how your app handles messages from the payment company to update payment status. Testing all these steps helps protect customers and keeps your online store running smoothly.

6. Testing APIs for Third-Party Integrations: How do you ensure the reliability of APIs when integrating third-party services?

   - To make sure third-party APIs work well with your app, start by reading their documentation to understand how they work. Use a test environment or sandbox if available. Then, test different scenarios like:

   - Successful responses
   - Errors (like wrong data or no internet)
   - Slow or delayed responses

   - Make sure your app handles these situations properly it shouldn't crash or show confusing errors. Also, check if the API is secure and if you're using the right keys and permissions. Finally, keep an eye on the API's limits and have fallbacks or backups in case the API goes down.

7. Testing a Mobile App: You are responsible for testing a mobile app that integrates with GPS and payment systems. What key areas would you focus on?

   - When testing a mobile app that uses GPS and payment systems, focus on:

   - **Location Accuracy**: Make sure the GPS gives correct location data in different places and conditions (like indoors or with poor signal).
   - **Permissions**: The app should ask for location and payment permissions properly and work correctly if the user says “no.”
   - **Payment Testing**: Try different payment scenarios (successful, failed, cancelled, slow, etc.) using test cards or a sandbox.
   - **Network Conditions**: Test how the app works with slow or no internet.
   - **Battery and Performance**: GPS and payments can drain battery—make sure the app doesn’t overuse them.
   - **Security**: Make sure all data is sent safely and sensitive info (like card details) is protected.

References:

Aratek ((2023) *Biometric ATM: The future of secure Financial Transactions*. https://www.aratek.co/news/biometric-atm-the-future-of-secure-financial-transactions

Rachlin, R. (2024, May 27). *How flexibility in software testing can build a Five-Star product*. Ubertesters. https://ubertesters.com/blog/how-flexibility-in-software-testing-can-build-a-five-star-product/