ISTQB Advanced Agile Technical Tester
Exercise 2.2

- **ATT-2.2.1-1 (K4)**
    - **Analyze the creation of a test approach using test automation, experience-based tests and back-box tests, created using other approaches (including risk-based testing) for a given scenario in an Agile project**

You are a test analyst working at a bank on an agile project. Releases are implemented every two weeks. The features, represented by user stories, are selected and prioritized the week prior to the start of the sprint in a sprint planning meeting.

Typically, five to seven features are selected for the upcoming sprint. These features range from very simple to moderately complex. In this sprint, a new security feature will be implemented, along with a different way to display account balances to customers.

The user stories are typically very brief and lack much detail. Acceptance criteria are associated with each story, but there are always more points of acceptance than get documented.

The test team is fairly experienced with the business processes and the application, but the new features can sometimes be confusing.

Past history shows that, on some occasions, a feature may not be implemented if it is failing. But, those are the exceptions rather than the rule.

There is a test tool in place which is based on a scripting-first approach (as opposed to record/playback).

With this information, describe how you would create a test approach, especially how you would address each of these areas:

Test Automation (e.g., which tests should be automated and by whom, how often should the automated tests be performed, etc.)

Experience-Based Tests (e.g., what would be good candidates? Who would perform tests? Etc.)

Black-box Tests (Tests designed with techniques such as BVA, EP, Decision Tables and State-Transition test design techniques) (e.g., what would be good candidates? Who would analyze, design and perform the tests? Etc.)

Which major test approaches (risk-based, consultative, etc.) would be appropriate and how would the approach(es) support better release quality?