

Banking Domain Application Testing: Sample Test Cases

Banking Domain Testing

Banking Domain Testing is a software testing process of a banking application for functionality, performance, and security. The main purpose of testing banking application is to ensure that all the activities and functionalities of a banking software run smoothly with no errors and it remains protected.

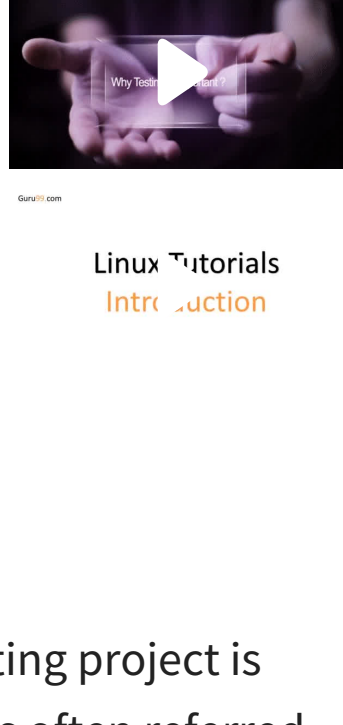
The BFSI (Banking, Financial services and Insurance) sector is the biggest consumer of IT services. Banking Applications directly deal with confidential financial data. It is mandatory that all the activities performed by banking software run smoothly and without any error. Banking software perform various functions like transferring and depositing fund, balance inquiry, transaction history, withdrawal and so on. Testing banking application assures that these activities are not only executed well but also remain protected from hackers.

In this tutorial, we will learn

- What is Domain in Testing?
- Why Domain Knowledge Matters?
- Introduction to Banking Domain
- Characteristics of a banking application
- Stages of testing banking applications
- Sample Test Case for Net Banking Login Application
- Challenges in testing banking domain & their Mitigation

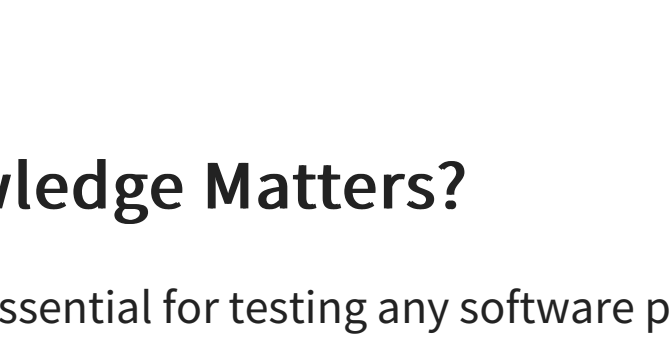
Join our Live Banking Testing Project for Free

FEATURED VIDEOS



What is Domain in Testing?

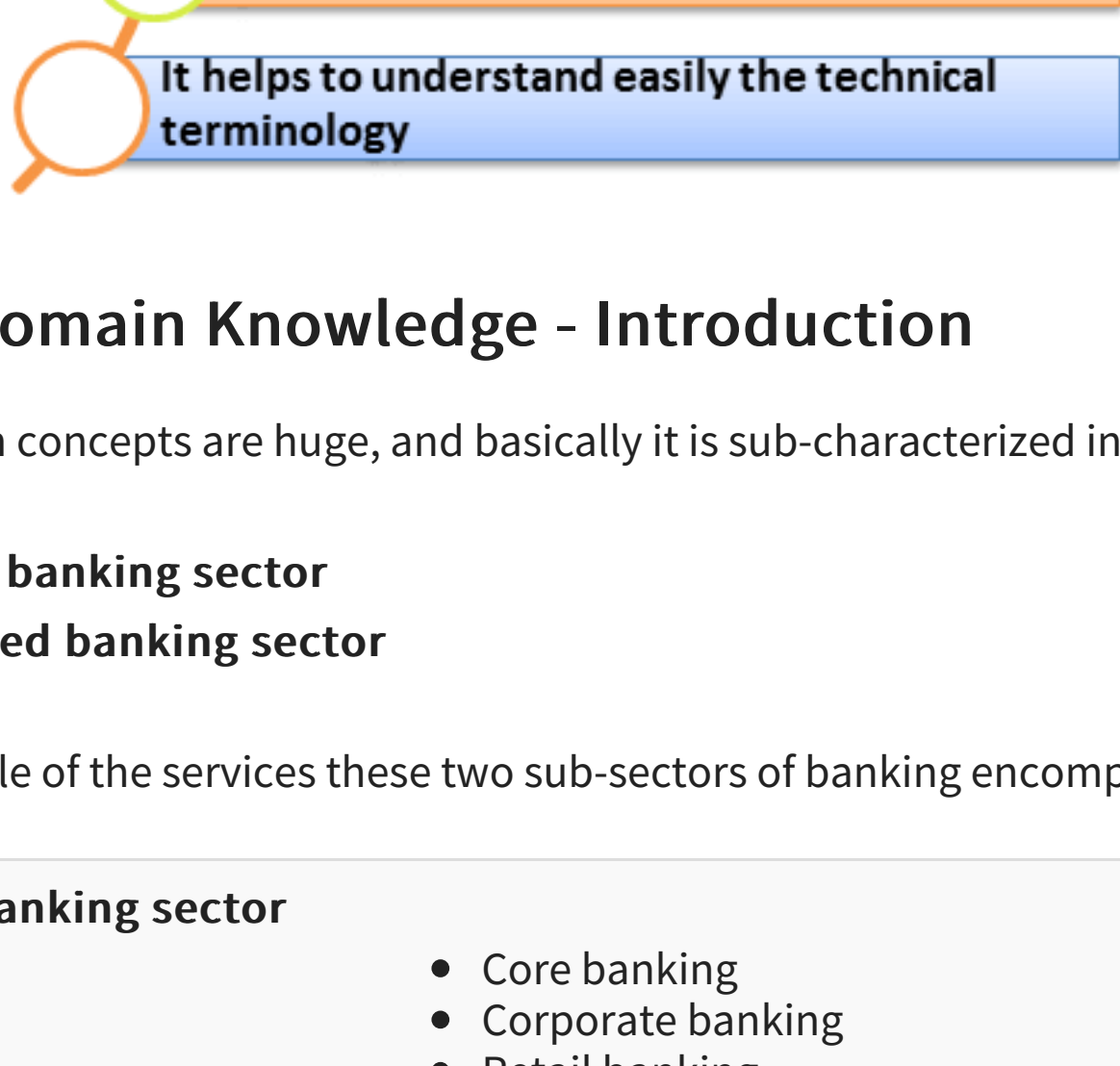
Domain in Testing is nothing but the industry for which the software testing project is created. When we talk about software projects or development, this term is often referred to. For example, Insurance domain, Banking domain, Retail Domain, Telecom Domain, etc.



Usually, while developing any specific domain project, domain expert help is sought out. Domain expert are master of the subject, and he may know the inside-out of the product or application.

Why Domain Knowledge Matters?

Domain knowledge is quintessential for testing any software product, and it has its own benefits like



Banking Domain Knowledge - Introduction

Banking domain concepts are huge, and basically it is sub-characterized into two sectors

- Traditional banking sector
- Service based banking sector

Below is the table of the services these two sub-sectors of banking encompass

Traditional banking sector	<ul style="list-style-type: none">Core bankingCorporate bankingRetail banking
Service based banking sector	<ul style="list-style-type: none">CoreCorporateRetailLoanTrade financePrivate bankingConsumer financeIslamic bankingCustomer delivery channels/Front end delivery

Based on the scope of your project you may need to test one or all of the above service offerings. Before you begin testing, ensure you have enough background on the service being tested.

Characteristics of a Banking Application

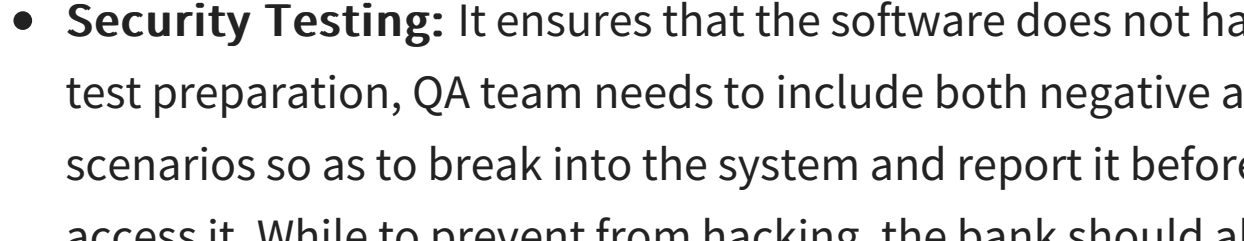
Before you begin testing, it's important to note the standard features expected of any banking application. So that, you can gear your test efforts to achieve these characteristics.

A standard banking application should meet all these characteristics as mentioned below.

- It should support thousands of concurrent user sessions
- A banking application should integrate with other numerous applications like trading accounts, Bill pay utility, credit cards, etc.
- It should process fast and secure transactions
- It should include massive storage system.
- To troubleshoot customer issues, it should have high auditing capability
- It should handle complex business workflows
- Need to support users on multiple platforms (Mac, Linux, Unix, Windows)
- It should support users from multiple locations
- It should support multi-lingual users
- It should support users on various payment systems (VISA, AMEX, MasterCard)
- It should support multiple service sectors (Loans, Retail banking etc.)
- Foolproof disaster management mechanism

Test Phases in Testing Banking Applications

For testing banking applications, different stages of testing include



- Requirement Analysis:** It is done by business analyst; requirements for a particular banking application are gathered and documented
- Requirement Review:** Quality analysts, business analysts, and development leads are involved in this task. The requirement gathering document is reviewed at this stage, and cross-checked to ensure that it does not affect the workflow
- Business Requirements Documentation:** Business requirements documents are prepared by quality analysts in which all reviewed business requirements are covered
- Database Testing:** It is the most important part of bank application testing. This testing is done to ensure data integrity, data loading, data migration, stored procedures, and functions validation, rules testing, etc.
- Integration Testing:** Under [Integration Testing](#) all components that are developed are integrated and validated
- Functional Testing:** The usual software testing activities like [Test Case](#) preparation, test case review and test case execution is done during this phase
- Security Testing:** It ensures that the software does not have any security flaws. During test preparation, QA team needs to include both negative as well as positive test scenarios so as to break into the system and report it before any unauthorized individual access it. While to prevent from hacking, the bank should also implement a multi-layer of access validation like a one-time password. For [Security Testing](#), automation tools like IBM AppScan and HPWebInspect are used while for [Manual Testing](#) tools like Proxy Sniffer, Paros proxy, HTTP watch, etc. are used
- Usability Testing:** It ensures that differently able people should be able to use the system as normal user. For example, ATM with hearing and Braille facility for the end users to ensure the compliance of the application with the real world scenario.
- User Acceptance Testing:** It is the final stage of testing done by the end users to ensure the compliance of the application with the real world scenario.

Sample Test Case for Net Banking Login Application

Security is prime for any banking application. Therefore, during test preparation, QA team should include both negative and positive test scenarios in order to sneak into the system and report for any vulnerabilities before any unauthorized individual get access to it. It not only involves writing negative test cases but may also include destructive testing.

Following are generic test cases to check any banking application

Sample test cases	
For Admin	<ul style="list-style-type: none">Verify Admin login with valid and Invalid dataVerify admin login without dataVerify all admin home linksVerify admin change password with valid and invalid dataVerify admin change password without dataVerify admin change password with existing dataVerify admin logout
For new Branch	<ul style="list-style-type: none">Create a new branch with valid and invalid dataCreate a new branch without dataCreate a new branch with existing branch dataVerify reset and cancel optionUpdate branch with valid and invalid dataUpdate branch without dataUpdate branch with existing branch dataVerify cancel optionVerify branch deletion with and without dependenciesVerify branch search option
For New Role	<ul style="list-style-type: none">Create a new role with valid and invalid dataCreate a new role without dataVerify new role with existing dataverify role description and role typesVerify cancel and reset optionVerify role deletion with and without dependencyverify links in role details page
For customer & Visitors	<ul style="list-style-type: none">Verify all visitor or customer linksVerify customers login with valid and invalid dataVerify customers login without dataVerify banker's login without dataVerify banker's login with valid or invalid data
For New users	<ul style="list-style-type: none">Create a new user with valid and invalid dataCreate a new user without dataCreate a new user with existing branch dataVerify cancel and reset optionUpdate user with valid and invalid dataUpdate user with existing dataVerify cancel optionVerify deletion of the user

Challenges in testing Banking domain & their Mitigation

Challenges tester might face during testing banking domain are

Challenge	Mitigation
<ul style="list-style-type: none">Getting access to production data and replicating it as test data, for testing is challenging	<ul style="list-style-type: none">Ensure that test data meets regulatory compliances requirements and guidelinesMaintain the data confidentiality by following techniques like data masking, synthetic test data, testing system integration, etc.
<ul style="list-style-type: none">The biggest challenge in testing banking system is during the migration of the system from the old system to the new system like testing of all the routines, procedures and plans. Also how the data will be fetched, uploaded and transferred to the new system after migration	<ul style="list-style-type: none">Ensure Data Migration Testing is completeEnsure Regression Test cases are executed on old and new systems, and the results match.
<ul style="list-style-type: none">There may be the cases where requirements are not documented well and may lead to functional gaps in test planMany non-functional requirements are not fully documented, and testers do not know whether to test it or not	<ul style="list-style-type: none">The test should participate in the project right from Requirement Analysis phases and should actively review the Business Requirements
<ul style="list-style-type: none">The most important point is to check whether the said system follows the desired policies and procedures	<ul style="list-style-type: none">Compliance or Regulatory Policies testing must be done
<ul style="list-style-type: none">The scope and the timelines increases as banking application are integrated with other application like internet or Mobile banking	<ul style="list-style-type: none">Ensure Time budget for Integration Testing is accounted if your banking application has many external interfaces

Summary

Banking domain is the most vulnerable area for cyber-theft, and safeguarding the software requires precise testing. This tutorial gives a clear idea of what it takes for banking domain testing and how important it is. One must understand that -

- Majority of banking software are developed on **Mainframe** and **Unix**
- Testing helps to lessen possible glitches encounter during software development
- Proper testing and compliance to industry standards, save companies from penalties
- Good practices help develop good results, reputation and more business for companies
- Both manual and automated testing have respective merits and usability

Join our [Live Banking Domain Testing Project](#)

YOU MIGHT LIKE:

SOFTWARE TESTING

What is Volume Testing? Learn with Examples

What is Volume Testing?
VOLUME TESTING is a type of Software Testing, where the software is...

[Read more >](#)

SOFTWARE TESTING

Mainframe Testing - Complete Tutorial

Before learning mainframe testing concepts, lets learn What is a Mainframe? The mainframe is a...

[Read more >](#)

SOFTWARE TESTING

Unit Test vs

Integration Test: What's the Difference?

What is the Unit Test? Unit Tests are conducted by developers and test the unit of code(aka...

[Read more >](#)

SOFTWARE TESTING

Web

Application Testing Checklist: Example Test Cases for Website

While testing the web applications, one should consider the below mentioned template. The below...

[Read more >](#)

COURSE

TEST

Management Tutorials: Complete Training Course

Course Summary Test Management is a series of planning, execution, monitoring and control...

[Read more >](#)

AGILE TESTING

Top 34 Agile Testing Interview Questions & Answers

Download PDF 1) As a tester what should be your approach when requirements change continuously? When...

[Read more >](#)