Guru99 Banking Project

Test Plan -

Revision History

Date	Description	Author	Comments
24.04.2023	v1	IonelaVasiliu	Initial drafts

Table of contents

1.Introduction

The Guru99 Bank project aims to provide net banking facility to its customers.

This document describes approaches and methodologies that will apply to the unit, integration and system testing of the application Guru99 Bank. It includes the objectives, test responsabilities, entry and exit criteria, scope, test process, test deliverables.

This test plan is designed to access and verify the functionalities of existing modules in the role of Manager, such as: the option of adding a new customer, but also to edit details of a customer and delete a customer if there isn't any exiting account on his name, the option of adding new account for an existing customer, the option of editing an account and deleting an account of an existing customer, by using the account number, the options that allow to deposit, withdraw and also transfer funds, in the Guru99 bank application.

This document includes all the steps that are made to fulfill the requirements given by the client.

1.1.Project objective

The project objective is to verify the functionalities of the modules in the account Manager, such that allows the entry of valid data in its fields, so that Manager should be able to add, edit or delete customers or accounts of existing customers, Manager should be able to perform actions as follows: deposit, withdraw, fund transfer, balance enquiry, mini statement, customized statement.

1.2.Functionalities in scope

The scope of this project is limited to the testing of the features in the succeeding sections of this document.

Functional testing are in scope and needed to be tested.

Only web applications will be tested.

1.3.Functionalities and tests out of scope

Non-functional testing like performance, security testing is beyond scope of this project.

No QA support for mobile application developed.

Automation testing is beyond scope.

2.Test process

A primary objective of testing is to assure that the system meets the full requirements, including quality requirements, and fit metrics for each quality requiremet and satisfies the use of test case scenarios and maintain the quality of the product.

At the end of the project development cycle, the client should find that the project has met all of their expectations as detailed in the requirements.

Any changes, additions or deletions to the requirements document will be documented and tested.

Another objective of testing will be: identify and expose all issues and associated risks, communicate all known issues to the project team and ensure that all issues are addressed in an appropriate matter before release.

2.1.Test planning

Roles and responsabilities:

Role	Responsabilities
Project manager	● Acts as a primary contact for development and QA team. ● Responsible for the project schedule and the overall success of the project.
QA Lead	● Participation in the project plan creation. ● Planning and organization of the test process. ● Coordinate with QA analysts on any issues encountered during testing. ● Report progress on work asignements to the PM.
QA	● Understanding requirements. ● Writing and executing test cases. ● Reviewing test cases. ● Reports to do. ● Defects reporting. ● Confirmation and regression testing. ● Bug reports. ● Preparation of test data. ● Coordinate with QA Lead for any issues or problems encountered during test preparation or test execution.

Entry criteria:

- roles needed for the project are allocated.
- Initial project risks were detected and mitigated.
- functional specifications are defined.
- verify if the test environment is available and ready for use.
- verify if test tools installed in the environment are ready for use.

Exit criteria:

- A certain level of requirements coverage has been achieved.
- No high priority or severe bugs are left outstanding.
- All high-risk areas have been fully tested, with only minor residual risks left outstanding.

- The schedule has been achieved.

Risks:

Project risks:

- the development team won't have the necessary training for these tasks.
- thick bureaucracy.
- the database won't support such a high volume of dates.
- tight deadline.
- not-compliance with the rules regulated in the banking field.
- sick personnel.
- a few team members.
- not having the necessary hardware equipment.
- misunderstanding of the requirements.

Product risks:

- Security risk-data leak.
- Low performance of the Guru99 bank application.
- Not having customer's requests.
- Application will crash.

RISK ID	RISK	Probability	Impact	Risk level=Impact x Probability
	Riscuri de proiect			
R1	The development team won't have the necessary training for these tasks.	3	4	12
R2	Thick bureaucracy.	2	3	6
R3	The database won't support such a high volume of dates	2	5	10
R4	Tight deadline.	3	3	9
R5	Not-compliance with the rules regulated in the banking field.	1	5	5
R6	Sick personnel.	2	2	4
R7	A few team members.	2	3	6
R8	Not having the necessary harware equipment.	1	5	5
R9	Misunderstanding of the requirements.	1	5	5
	Riscuri de produs			
R10	Security risk-data leak.	1	5	5
R11	Low performance of the Guru99 bank application.	2	4	8
R12	Not having customer's requests.	1	5	5

R13	Application will crash.	1	5	5

		Impact				
		Insignificant	**Minbor**	**Moderate**	**Major**	**Catastrophic**
	Probability	1	2	3	4	5
5	Almost certain					
4	Likely					
3	Possible			R4	R1	
2	Unlikley		R6	R2, R7	R11	R3
1	Rare					R5, R8, R9, R10, R12, R13

RISK MATRIX					
	Consequences/Impact				
	Insignificant	Minor	Moderate	Major	Catastrophic
Likelihood	1	2	3	4	5
Almost Certain	High	High	Extreme	Extreme	Extreme
Likely	Moderate	High	High	Extreme	Extreme
Possible	Low	Moderate	High	Extreme	Extreme
Unlikely	Low	Low	Moderate	High	Extreme
Rare	Low	Low	Moderate	High	High

2.2.Test analysis

The testing process will be executed, based on the requirements sent by the client, for the role of the Manager modules, in order to create test objectives, so that test conditions could be established. The team must fully understand the requirements. If appropriate, possible suggestions for the client can be made.

2.3.Test design

The tests conditions that were established in the test analysis phase will be elaborated into test cases and that will serve as a base for evaluating the behaviour of the Guru99 Bank Application, when the role of Manager is beeing used and if it is useful properly, in terms of the business requirements sent by the client.

2.4.Test implementation

The following elements are needed to be ready before the test execution phase begins:

- Testing environment is up and ready to be running.
- QA testers have completely understood the requirements.
- Create and prioritize test cases.
- Access to the testing environment is given.
- Cycle summary was created.
- Test cases were added to the cycle summary.
- Preparing test data and ensuring it is properly loaded in the test environment.

2.5.Test execution

In this phase, the execution of all test cases must be done, and also report the results in the used tools(Passes/Failed/Blocked).

Report Bugs must be created based on the failed tests.

Full regression testing is needed after the bugs are fixed.

2.5.1. Test status report

A test status report provides information about the status of the test effort, including overall software quality and test execution progress against the planned progress.

There are some steps that a status report should contain, such as:

- A test status report is used for tracking how much work is complete, how much work is left to be done.
- Using these status reports, we can even track the team performance. From a test status report, we can prepare future actionable items according to the priorities and make a list of the next week's priorities.
- It is also used to report issues that management should be aware of, in order to help solving them.
- Another purpose of the test status report is to track the total number of Test Cases that had been written, Test Cases passed, Test Cases failed, Test Cases to be executed.

In other words, a test status report is a report which is made weekly for tracking the actual progress, in order to permit solving any issues and/or defects and to establish tasks and priorities for the next week.

2.6.Test closure

At this stage, the analysis of the exit criteria defined in the test planning must be evaluated, in order to be as expected. Also, creating a test summary report to be communicated to the stakeholders is a major activity.

Another important activity in this stage is the product risk identification and analysis.

The following **product risks** were identified:

- Security risk-data leak.
- Low performance of the Guru99 bank application.
- Not having customer's requests.
- Application will crash.

The analysis of these product risks was made using the risk matrix model. This was described in the test planning section. Further actions will be taken in order to prevent these events.

2.7.Test monitoring and control

The actual progress will be periodic compared with the testing plan, and therefore, periodic reports will be made.

If there are any signs of not fulfilling the objective of the test plan, control measures will be executed.

Control measures are methods or actions that aim to eliminate, prevent, reduce, or mitigate the risks or hazards that exist in different contexts. Control measures may vary depending on the type and source of the risk or hazard, and may include techniques, practices, procedures, systems, devices, or other means of control. Control measures should be consistent with label requirements, health and safety responsibilities, and the hierarchy of controls.

3.Test deliverables

Test Deliverables are the test artifacts which are given to the stakeholders of a software project during the Software Development Life Cycle. In this process, there will be some deliverables in every phase. Some of the deliverables are provided before the testing phase begins and some are provided during the testing phase and rest after the testing phase is completed.

The deliverables will be described as follows:

3.1.Test conditions

Test conditions represent an item or event of a component or system that could be verified by one or more test cases. These are the titles established for test cases to be executed.

3.2.Test cases

A test case is a set of actions performed to determine if the system satisfies software requirements and functions correctly. The purpose of a test case is to determine if different features

within the system are performing as expected and to confirm that the system satisfies all related standards, guidelines and customer requirements.

3.3. Traceability matrix

3.4.Test case results

The typical test case format should detail the expected outcome and actual outcome, which the test itself validates. Most test case results fall into these categories: pass, fail, not executed, blocked.

Passing and failing tests indicate that the system either accomplishes what it is supposed to or fails in that attempt.

Test results that get marked as not executed are as the name suggests: tests that have not yet run, or will not run as part of this round of testing.

Blocked tests result from an external circumstance or precondition inhibiting the test from running. For example, a system failure that prevents functionality from being available will cause a blocked test, as will an improperly configured test environment.

When a failure is met, a bug report is necessary to make.

3.5.Bugs report

A good bug report covers all the crucial information about the bug, which can be used in the debugging process:

- It helps with a detailed bug analysis.
- Gives better visibility about the bug and helps find the right direction and approach towards debugging.
- Saves cost and time by helping debug at an earlier stage.
- Prevents bugs from going into production and disrupting end-user experience.
- Acts as a guide to help avoid the same bug in future releases.
- Keeps all the stakeholders informed about the bug, helping them take corrective measures.

An effective bug report should contain the following:

Title/Bug ID

Environment

Steps to reproduce a Bug

Expected Result

Actual Result

Visual Proof (screenshots, videos, text) of Bug

Severity/Priority

3.6.Test completion report

One of the critical outcomes of the Test Completion phase is the test completion report. This report is the summary of all the testing efforts which execute during the testing process. The completion report is a crucial input to the stakeholders to determine the amount of testing accomplished. In addition to that, it also analyzes the unattended risks and issues. It helps them to make informed decisions about the software.