



Document management system

Test Plan

Table of contents

1. Project summary	4
2. Scope of Testing	4
2.3 Preconditions and outer dependencies	4
2.4 Acceptance Criteria	4
3. Test Items	5
3.1. Features to Be Tested	5
3.2. Features Not to Be Tested	6
4. Testing Approach	7
4.1. Prioritization	7
4.2. Entry/Exit Criteria	7
4.3. Item Pass/Fail Criteria	8
4.4. Classification of Defects	8
5. Testing Levels	9
5.1. Module Testing	9
5.1.1. Test Responsibilities	9
5.1.2. Test Methods	9
5.1.3. Entry and Exit Criteria	9
5.2. System Testing	9
5.2.1. Test Responsibilities	10
5.2.2. Test Methods	10
5.2.3. Entry and exit criteria	10
5.3. User Acceptance Testing	10
5.3.1. Test Responsibilities	10
5.3.2. Entry and Exit Criteria	10
5.4. Other Testing Level Types	11
5.4.1. Smoke Testing	11
5.4.2. Regression Testing	11
6. Test Schedule, Organization and Responsibilities	11
6.1. Testing Schedule	11
6.2. Test Project Organization & Resources	11
7. Tool for Test Design and Execution	12
8. Test Environments	12
8.1. Development Environment	12
8.2. Test Environment	12
8.3. Production Environment	12
9. Test Metrics	12

10. Deliverables	13
10.1. Test Items and Documents	13
11. Change history (document)	13

Document management system

1. Project summary

Document management system is a system to track, manage and store organization documents. Users are able to create, sign, review, download documents according to the rights of groups they belong.

The purpose of this document is to define the test approach for the project. In this document the project context, test approach, testing levels, test environments, metrics to be collected and initial risks from testing point of view will be defined.

2. Scope of Testing

2.1. In scope

All of the functional requirements will be covered with manual and automation tests. The functional testing is carried out by feeding the input and validating the output from the application.

Regression and smoke testing will be automated using Selenium IDE.

4 of 7 non-functional requirements will be tested:

- Passwords stored in database will be checked if they are hashed using SHA-256 or higher (NFR4).
- The system's event log will be checked if it's working according to requirements (NFR5).
- The system will be checked if every query is executed within 1 second (NFR7).

2.2 Out of scope

- System integration testing.
- Accessibility testing.
- Security testing.
- User Acceptance testing will be conducted by End-users.
- Non-functional requirements which describe development tools or technologies (NFR1, NFR2, NFR3, NFR6 - Check DMS system requirements for references).

2.3 Preconditions and outer dependencies

The external preconditions that must be available at given time in order to be able to start a specific work for testing the project:

1. OS with an internet connection.
2. Web browser.
3. First the software items should be installed to the production environment.

2.4 Acceptance Criteria

- Regression testing is completed and all the bugs are registered and fixed.

- All of the planned test cases are completed.
- Final test report is ready.
- System testing is completed and no less than 70% of first priority tests passed.
- User acceptance testing is 100% completed.
- No high priority or severe bugs are left outstanding.
- The schedule has been achieved.

3. Test Items

3.1. Features to Be Tested

Testing of the application will focus on all target requirements that can be traced directly to use cases (or business functions), and business rules. The goals of these tests are to verify proper data acceptance, processing, and retrieval, and the appropriate implementation of the business rules. This type of testing is based on black box techniques, that is, verifying the application (and its internal processes) by interacting with the application via the GUI and analyzing the output (results).

All of the features in the product backlog will be tested.

Subject	Comments	Reviewer
System users	Only identified users can use the system.	Test team
System user log in	User logs in using username and password.	Test team
Administrator log in	Administrator logs in using username and password.	Test team
Group creation	Administrator creates user groups.	Test team
User assignment to groups	Administrator assigns users to groups.	Test team
Types of documents	Administrator creates types of documents.	Test team
Assigning document type to a group	Administrator assigns document types to a group and sets group rights to create or approve documents of this type.	Test team
User registration	Administrator registers new system users filling username, First name, Last Name, password and role fields.	Test team
Password updation	Administrator is able to update user passwords.	Test team
Assigning users to multiple groups	Administrator is able to assign users to several groups.	Test team
Document creation	A user creates a document based on the groups he/she is assigned to.	Test team
Types of documents	It is possible to create different document types in the system.	Test team
Unlimited document creation	Groups should be able to create an unlimited number of assigned document types.	Test team
Document approval	A user is able to submit documents for approval.	Test team
Restricted permission to view documents	A user is the only one that can see his/hers created documents that were not yet submitted.	Test team

Document signing and rejection	A user is able to preview, sign or reject pending documents and attached files that were submitted to his/her groups.	Test team
Mandatory document properties	Each document must have(mandatory): unique id number; date of creation; author; type; name; Description.	Test team
Applicable document properties	Each document must have(if applicable):date of submission; date of approval; date of rejection; validating person; reason of rejection; one or more attached files (.pdf).	Test team
State of the document	Submitted documents should display a state: "Submitted", "Signed", "Rejected".	Test team
Document statistic	A user is able to see statistics about document type (count of submission, acceptance and rejection) assigned to his/her group.	Test team
Document filtration	A user is able to filter statistics of documents assigned to his/her group based on time frame or the count of submissions.	Test team
Document download	A user is able to download an archive (zip) which contains his/hers documents.	Test team
Administrator registration	An administrator is able to register new administrators.	Test team
Editing groups	An administrator able to edit groups by adding or removing users, changing group's permissions.	Test team
User documents preview	Users are able to view list of all his/her documents and are able to filter them.	Test team
User documents review	Users are able to review selected documents from the list.	Test team
Password security	User passwords should be encrypted using hash (security should be SHA-256 or higher)	Test team
Methods for security	"Salt" and "pepper" methods are used to increase security.	Test team
Logging the actions	Every action performed within the system should be logged.	Test team
Log properties	Event log should contain: time of the event, the name of the event and all relevant information.	Test team
Security of sensitive information	Event log should NOT contain sensitive information such as passwords or document content.	Test team
Performance requirements	Any action should not take more than 1 second to complete	Test team
File saving	Files attached to the documents should be saved within the system.	Test team

3.2. Features Not to Be Tested

Non functional requirements: are not included in the testing commitment, the development team is responsible for fulfilling these requirements.

Subject	Comments	Reviewer
System development	System should be developed using Java 8+, Spring 5+.	Test team

API development	API should be developed using React 16+.	Test team
Database type	System should use RDB (Relational database).	Test team

4. Testing Approach

The scope of testing includes Verification and Validation of the system under test.

Validation is the confirmation that the requirements for specific use have been fulfilled.

The material used as a base for creating test cases are requirements and technical architecture for the more technical testing levels as Module Testing. Use cases are used for more business like testing as Acceptance Testing and the testing levels in between use a combination of these.

Testing will be done from three different points of view. These are technical testing, integration testing and requirements fulfilment. The technical testing will verify that system functionalities are working internally, integration testing that functionalities are working together and requirements fulfilment that all items created successfully meet the requirements.

4.1. Prioritization

- Implemented functionalities.
- Common new interfaces.
- Basic use cases or end to end scenarios.
- Error scenarios and special cases .
- Old functionalities.
- Old interfaces.

4.2. Entry/Exit Criteria

Entry criteria define the level where we should be before the actual testing can be started. Exit criteria describe the level where we should be after the testing is completed.

Entry criteria for each testing level are:

- Environment in question is up and running.
- Environment in question is stable.
- Environment in question has been fully allocated for testing.
- Software items have been implemented and installed to the correct environment.
- All needed configurations have been completed.
- Required test data has been imported or created.
- Base documents are approved and no further changes are expected.
- Test cases are created based on correct documents.
- Test cases are reviewed and approved.
- Earlier testing level exit criteria has been fulfilled.

Exit criteria for each testing level are:

- Test Cases related to testing level are agreed to cover enough of the requirements.

- All urgent and high severity defects related are fixed and closed.
- All Test Cases related to testing level in question are executed.
- All entry and exit criteria are fulfilled.

4.3. Item Pass/Fail Criteria

Item passes if it is proved that:

- The material used as the base for testing is approved.
- The test case is correctly based on the correct test base, i.e. Use case or requirement.
- The test case is reviewed and approved.
- All test cases related to item are successfully run.
- The coverage of the test cases is seen and approved to cover enough.
- The item is seen as functional from integrated parts as well.

Item fails if there are:

- Misunderstandings in the test base material that cause test cases to fail from some point of view.
- Code is not correct.
- Human error may cause items to fail in the testing phase.
- Items can not be tested because of test environment issues.
- If there are any urgent or high severity defects open that are related to some other related issue.
- If the item is not created at all.
- Item is not functional because of some other integrated system.

4.4. Classification of Defects

The table below shows the classification of the defect during the implementation project.

Severity	Description	Impact to testing	Actions	Analysis & Fix schedule
Urgent Defect	Defect causes that important functionalities are not achievable, that data disappears, performance deteriorates or parts of the system crash	showstopper for testing	Needs actions immediately	Defect analysis should be started immediately, fix delivery time ASAP
High Defect	Defect means that a functionality does not work according to specifications or does not work at all	Essential function not functional	Needs actions soon	Defect analysis should be started latest in one to two days, fix delivery in one week
Medium Defect	With Medium severity defect it is meant that some part of the system does not work normally but needs some special actions in order to get all the normal functionalities	Some part of function is not working	Can be accepted, actions to be decided in the Steering group	Defect analysis should be started in one week, fix delivery in two weeks
Low Defect	With a Low severity defect it is meant a defect that is mainly cosmetic and does not	Cosmetic error, e.g. wrong error message	Can be accepted	Defect analysis should be started in one week if there is

	have any effect on functionality			time and higher severity defects are fixed, fix delivery in one month
--	----------------------------------	--	--	---

5. Testing Levels

5.1. Module Testing

Module testing will be performed by the development team.

After this testing level the items should be ready for System Testing after some Smoke Testing.

5.1.1. Test Responsibilities

Development team is responsible for this testing level..

5.1.2. Test Methods

Developers will cover their code with jUnit integration tests.

5.1.3. Entry and Exit Criteria

Entry criteria:

- Planning phase has been completed.
- System design, technical design and other relevant documents are properly reviewed, analysed and approved.
- Business and functional requirements are defined and approved.
- Testable codes or units are available.
- Availability of test environment.

Exit criteria:

- Successful execution of the unit tests.
- All the identified bugs have been fixed and closed.
- Project code is complete.

5.2. System Testing

System Testing is a technical testing type to verify that a system meets its specified requirements. Here the behaviour of the whole system is tested. Test cases should be based on requirements and risks.

This testing level will be executed by the test team. The actual testing will be completed in the Test environment that is fully allocated for testing at this point.

Between Module Testing and System Testing some Smoke Testing will be executed to approve the system to be ready for System Testing. The system will be ready for System Integration Testing when the exit criteria for this testing level are fulfilled.

5.2.1. Test Responsibilities

Test team is responsible for this testing level..

5.2.2. Test Methods

Test team will create test cases and execute these after approval according to an earlier decided schedule.

Test cases are created by the test team based on each item the product backlog. Each test case will be run by one of the test team members.

5.2.3. Entry and exit criteria

Entry criteria:

- Working functionality available.
- Test cases are prepared for execution.

Exit criteria:

- Successful execution of the system tests.
- All specified business and functional requirements have been met.
- Priority bugs have been fixed and closed.

5.3. User Acceptance Testing

Acceptance Testing is a formal testing type where the objective is to verify that a system meets its specified requirements from the user point of view. Most often the use cases or similar end to end business scenarios are tested here.

This testing level should be executed by customer and there should be an Acceptance Testing Coordinator available to plan and monitor the testing. Customers should include end users or other authorized entities to determine whether to accept a system or component. The actual testing should be completed in the environment as production like as possible but not in production if the production is already in use.

The system should be ready for production use when the exit criteria for this testing level are fulfilled.

5.3.1. Test Responsibilities

The customer will test this testing level. This means that usually the end users are the testers as mentioned above.

5.3.2. Entry and Exit Criteria

Entry criteria:

- Successful completion of system testing phase.
- Priority bugs found during previous testing activities have been fixed and closed.

- Functional and Business requirements have been met.
- Acceptance testing environment is ready.

Exit criteria:

- Successful execution of the user acceptance tests.
- Business requirements got fulfilled.
- No critical defects have been left out.
- Product Owner signs off acceptance testing.

5.4. Other Testing Level Types

There are a few other testing types that should be taken into consideration while planning, designing and executing tests. These testing types are Smoke Testing and Regression Testing.

5.4.1. Smoke Testing

Smoke Testing is testing basic operations in the system without planned test cases to show that the system is functional. This should be executed every time a new version of code is delivered to an environment to verify that the quality of the code is high enough. Smoke Testing is one of the main entry criteria for System Testing.

In Module Testing level the person doing the installations is responsible for doing Smoke Testing. In the System Testing level, that happens in the Test environment the responsible person or team is the test team. If Acceptance Testing is done in another environment, then the new version in the environment should also be smoke tested before starting the actual testing and that should be carried out by the customer.

5.4.2. Regression Testing

Regression Testing is testing some certain set of test cases to verify that a system has not been affected by updates as new version or defect fixed. Regression test set should be executed every time a new version is installed to a system or related defect fixed.

Difference to Smoke Testing is that here test cases are designed and approved and form a ready-made set of test cases based on normal use cases. The regression test set can be executed as a whole or only a part of the test set may be chosen in special cases where smaller updates are done to the system.

6. Test Schedule, Organization and Responsibilities

6.1. Testing Schedule

Test Level	Task	Planned schedule
Module Testing	Module Testing completed	01/23/20 – 03/27/20
System Testing	System Testing completed	01/23/20 – 03/27/20
Acceptance Testing	Acceptance Testing completed	02/07/20 – 03/27/20

6.2. Test Project Organization & Resources

Role	Responsibilities
QA team	1. Participation in the project plan creation/update process. 2. Planning and organization of test process for the release. 3. Understand requirements. 4. Writing and executing Test cases. 5. Reviewing Test cases. 6. Preparation of Test Data 7. Defect reporting and tracking. 8. Retesting and regression testing. 9. Bug Review meeting.

7. Tool for Test Design and Execution

Process	Tool
Test case creation	Microsoft Excel
Test case tracking	Microsoft Excel
Test case execution	Manual, Selenium, Eclipse IDE
Test case session management	Jira
Defect management	Jira
Test reporting	PDF

8. Test Environments

8.1. Development Environment

Component	Environment
Operating system	Linux
Project Code Storage	Github Repository
Software	Git, Visual Studio Code, Eclipse IDE

8.2. Test Environment

Component	
Operating System	Windows 10
Browser	Google Chrome Version 79.0.3945.130 (Official Build) (64-bit)

8.3. Production Environment

Component	
Operating System	Any operating system

9. Test Metrics

Test execution status report is delivered after each sprint, containing a number of executed test cases and functional requirements coverage.

Final test status report is delivered at the end of system development.

10. Deliverables

10.1. Test Items and Documents

- Master Test Plan.
- Sprint Test report.
- Final Test Report.
- Test Cases.

11. Change history (document)

Version	Date	Author	Reviewed by	Approved by	Change history
v1.0	01/28/20	Test team	Test team	Test team	-