**Testing**

Testing of the System is critical for the delivery of a successful project. The key purpose of testing is to check whether that the developed system is fulfilled the user requirements and confirms to the test results expected under a wide range of conditions which could have been tested. To accomplish a successful system implementation, all aspects of the testing segment should be reviewed with the contribution of the developer as well as users. In this chapter it is illustrated about the types of testing used for developed software product and test plans for used in the process. As the last main topic in this chapter it is discussed about the implementation plan, which includes the factors need to be considered when system is implemented in the real environment.

**System Testing Objectives**

System testing enables the developed solution to be analyzed entirely in order to ensure that the specifications and business functions which it was intended are being met. The components of the system are being tested at the development of those components. But it is necessary to test those components to confirm the combinations of the system components were properly done.

System testing is conducted by checking the fallowing checkpoints are tailored with the developed system.

* Performance –This testing is performed to verify the system operate under peak and continuous loads of processes at a significant speed.
* Accuracy – Inaccurate information leads the whole system failure if the information is not stable. The test is performed to verify the outputs of the system are accurate in various operational environments.
* Functionality – Test is performed to make sure the system meet requirement specifications and hence supports business requirements of the company. Functionalities should address the problems existed in current system and its processes.
* Interfaces – New solution components are developed to provide more information using less number of interfaces and users should be able identify them easily and separately. Testing is performed to make sure the interfaces are done according to the specifications and the interfaces are linked each other in well-organized manner.
* Security – Since the system handles sensitive information of the organization’s and it’s users security of the system should be well-ensured, this testing is performed to verify the feature of the system ensure access, integrity and recovery features operate as expected.

**5.3.2 Testing Strategy**

A proper testing strategy is needed for the finishing of a good product. The developers intend to give the best finished product to the buyers. Testing is done in different methods and such methods are elaborated below. It is important that the testing is done in a step by step basis and that the different types of tests are related to each other.

**Unit testing**

Unit testing which also known as module is testing is a kind of procedure used to verify that each unit of source code is functioning correctly. In an application, the smallest testable part is the unit. Units of a system are differentiated from modules and those modules are made up of units. Unit testing will only test the components of the system and it will not help to recognize every fault in that system such as integration errors, performance errors and any system related issues. Unit testing will only be effective when it used along with other testing techniques. Typically this testing is done only by the system developers and not by the users who are going to use the system finally at the real operational environment.

Integration testing

Integration testing which is also known as Integration and Testing is a kind of testing in which the individual modules in the software are combined and tested those as a cluster. Purpose of conducting the integration testing is verifying the functional and performance requirements defined on project design specifications. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing. There can be many types of integration testing such as back bone, bottom-up, top-down and big bang. Integration tests can not include system-wide change testing.

System Testing

Overall functionalities of the system including the modules integrated modules and interfaces will be tested out of the system testing. System testing will not receive many errors since the input which is taken to the system testing is the system and modules which are tested under unit testing and integration testing. But the errors which identify during this testing process will be critical since it will be affected to the integration of the solutions various modules with each other and leads to changes of the system as a whole.

Acceptance Testing

Acceptance testing is a black-box testing which will be the next to be conducted. It is designed to ensure that all the changes made come across with the original system specifications and user requirements of the solution during the design, development and other initial stages. Basically this will decide the how the extend solution has completed the user requirements. Acceptance phase may also act as the final quality gateway, where any quality defects not previously detected may be uncovered

**Test Plan**

Test plan defines a systematic approach to test a system. This will discuss the process of conducting the above mentioned testing strategies. Under test plan, test strategies which are identified as to be carried, will be separated into to testing plans as unit testing and system testing. Components of the system and integration of components will be carried under unit testing. System testing and acceptance testing will be carried under System testing. Concerns made when designing the test plan are,

Identifying the components and features to be tested and not tested

Ensuring all required elements are in place for testing

Who conducts the testing for particular component or feature

Plan for make necessary changes for issues arise on testing

|  |  |
| --- | --- |
| Test Scenario | Description |
| Test and validate login module for web application | Check whether the login interface is aligned with the user experience guides, results when the credentials are correct and wrong, session control and the logout function from the system. |
| Test and validate the functions of policy module | Check whether the policy module is loading properly. Check whether it is possible add a new policy for a client filling out the necessary details. Once it is submitted record should be visible in the data grid. Grant the permission to do necessary modifications to certain details of a policy. |
| Test and validate the claim request handling module | Check whether the requests sent by the clients are received by the systems in a quick manner. And view the request and respond to the request and change the state of the request. |
| Test and validate the functions of reviewing and approving the claim | Check whether manager is being able to view the claim report submitted by the claim agent and review it and approve it for money transfer. |
| Test and validate the functions of garage services module. | Check whether existing data are retrieved from database and displayed on the data grid. Check whether the searching is working providing indicated parameters. Check the ability to add, edit or delete data from/to database. |
| Test and validate the functions of tow truck services module. | Check whether existing data are retrieved from database and displayed on the data grid. Check whether the searching is working providing indicated parameters. Check the ability to add, edit or delete data from/to database. |
| Test and validate the functions of vehicle templates module. | Check whether existing data are retrieved from database and displayed on the data grid. Check whether the searching is working providing indicated parameters. Check the ability to add, edit or delete data from/to database. |
| Test and validate the functions of spare parts module. | Check whether existing data are retrieved from database and displayed on the data grid. Check whether the searching is working providing indicated parameters. Check the ability to add, edit or delete data from/to database. |
| Test and validate the functions of manufacturers’ module. | Check whether existing data are retrieved from database and displayed on the data grid. Check whether the searching is working providing indicated parameters. Check the ability to add, edit or delete data from/to database. |
| Test and validate the functions of spare categories module. | Check whether existing data are retrieved from database and displayed on the data grid. Check whether the searching is working providing indicated parameters. Check the ability to add, edit or delete data from/to database. |
| Test and validate the functions of employees and user accounts modules. | Check whether existing data are retrieved from database and displayed on the data grid. Check whether the searching is working providing indicated parameters. Check the ability to add, edit or delete data from/to database. |

|  |  |
| --- | --- |
| Test Scenario | Description |
| Test and validate the WCF service hosted by looking in to exposed methods | Check whether all the necessary methods are exposed and check in the WCF test client whether they all response with correct set of data. |

|  |  |
| --- | --- |
| Test Scenario | Description |
| Test and validate the smartphone applications report accident for claiming. | Run the smartphone application and check whether the report function is working. Check with various policy number inputs. |

|  |  |
| --- | --- |
| Test Scenario | Description |
| Test and validate login module for tab application | Check whether the login interface is aligned with the user experience guides, results when the credentials are correct and wrong, and the logout function from the system. |
| Testing and validating Policy information module. | Test whether the system identifies the customer’s policy number and allows the agent to retrieve the relevant data of the relevant module from the company database through the web service. |
| Test and validate the claim form filling procedure | Make sure that only after giving the policy number agent can go to this screen. All the required fields must be checked before submitting the response. Finally agent must be able to submit the data and get the success message. |
| Testing and validating Vehicle component information module. | Test whether the system chooses the particular parts list after the part category is selected from the pick list, and whether the right price is given, of the damaged spare part chosen. Test whether the system can add /remove the damaged spare parts to claim and whether it displays on screen. When each item is added, the value and details of the item should be shown in a table. |
| Test and validate garage services module in tab application | Check whether the data are retrieved through the web service and also check whether filtering works. |
| Test and validate tow truck services module in tab application | Check whether the data are retrieved through the web service and also check whether filtering works. |