

## Testing Lotus Notes applications

Most of the Lotus Notes applications come under the technical domain of Rapid Application Development (RAD). Such applications span over a period of 2 to 3 months. Testing is primarily manual and as a project manager one always worries about the time that needs to be kept aside for unit testing, Functional testing and User acceptance testing. Most of the times UAT is last step before production. There may be various approaches to reduce the testing time of such applications. Many organizations use Lotus Domino as a RAD tool to develop functionally intensive groupware applications. They have a dedicated lotus development team that maintains and tests these applications. This includes enhancements, new releases, bug fixes, new applications. Normally all the testing (unit testing, integration testing, system testing etc..) is done by the development team except the UAT testing. The client representatives do the UAT.

Let us look at some of the common concerns that organizations have-

How can I cut down the cost of rework?

How can I cut down the cost of the testing cycle?

How can I be sure that the development vendor produces the right software in the estimated time?

Is the estimated time right?

Let us look at the second concern and address the same in this document. The methodology that we discuss would solve other concerns also to some extent.

This involves employing a third party vendor dedicated to testing these applications. Advantages –

1. 3<sup>rd</sup> party testing (unbiased); that would identify defects in the current release rather than they leaking into the future releases and hence increase rework and maintenance cost.
2. Well structured rigorous testing process
3. Quality control
4. Visibility of the application stability from a third party view

The idea is to cut down on the testing cycle by doing away with separate unit testing, integration testing, system testing and UAT plans. The exploratory approach is the best way to test a product quickly when starting from scratch.

You have only two stages in the test cycle Unit Testing and Exploratory functional testing. In this methodology we only have the following documents.

1. Unit test plans for the development team by the development team
2. Functional specifications prepared by the business analysts
3. Document on exploratory test cases that were tested by the testing team. This is prepared by the testing team.

After unit testing the development team sets the application up in the QA environment. Conducts a comprehensive knowledge transfer by demonstrating the application to the QA testing team.

The testing team conducts the exploratory functional testing. This approach covers all the aspects of integration and system testing.

The outcome of an exploratory testing session is a set of notes about the product, failures found, and a concise record of how the product was tested. When practiced by trained testers, it yields consistently valuable and auditable results. This document forms the base line of test cases for future releases.

The success of the approach depends on the efficiency of the knowledge transfer that takes place between the development team and the testing team. It is common knowledge in the testing circles that the development team hesitates to discuss some critical potential failure areas in the application during the knowledge transfer process. Thus the onus is on the testing team to ask the relevant questions during the demonstration to get the maximum knowledge on the application and if time permits get a demonstration on the same. The whole idea is that the application that is set up in the QA environment goes through a sort of smoke test during the demonstration. Any critical/major/setup issues that crop up, can be recorded and the application sent back to the development/set up team before the test cycle starts.

Listed below is a sort of checklist with questions that could be asked during a demonstration of a Lotus application-

### **Environment**

1. Is this a notes application?

*If yes?*

1.1. Is this a mail in database?

*If yes? If no, go to step 1.2.*

- Ensure that the mail in database has been created in the appropriate NAB and configured accordingly.

## 1.2. Is it hosted on the appropriate server?

*If yes?*

- Does it require RDBMS connectivity?

*If yes?*

Ensure that the same is set up appropriately.

- Is this a web enabled Intranet/Internet application?

*If yes? If no go to step 2.*

- Ensure that the http task is running on the host server.
- If this application incorporates search functionality then ensure that the Notes database is full text indexed.
- Please ensure that “Anonymous” is added to the database ACL with “No Access” for web authentication to function.

## 2. Does this application have any external interfaces?

*If yes? If no, go to step 3.*

- Get the list of all of the interfaces that the application interacts with.
- Get the appropriate access privileges from the owners of the interfaces.
- Relationship between the dependencies must be outlined for e.g.,
  - Field Mapping.
  - Agents (If Any)
  - DBA support requirements.
- Get time estimates to satisfy dependencies requirements. E.g., time involved in the data entry to external interfaces, running of Denorm packs, schedule for refreshing data, etc.

## **Application Security**

3. What is the structure of the ACL setup in terms of the database access level, roles and privileges?

**Note: Please ensure that you have all of the test ids required for the testing of the application. Always use test ids and not your production ids. Also, make sure that your current location document is pointing to a QA Mail server.**

- Are there any roles?

*If yes?*

- Ensure that the roles have been added to the database ACL with appropriate access and privileges.
- Are there any groups?

*If yes?*

- Ensure that the groups have been created in the NAB and added to the database ACL with appropriate access and privileges.

## **Agents**

4. Does this application have any scheduled agents?

*If yes? If no, go to step 5.*

- How are they scheduled to run?
- Ensure that the agents are scheduled to run on the appropriate servers and at appropriate intervals. Also, ensure that the agents have been signed with an id (by the Support team) with unrestricted access to the designated server.

5. Does this application have a configuration document?

*If yes? If no, go to step 6.*

- Determine what these settings are and ensure that the appropriate settings are configured.

## **Scope and Limitations**

6. If the application has been QA'd before, is an emergency fix or an enhancement release, please discuss the scope of testing with the developer in the presence of your Account Manager.

### **Audience**

7. Who is the audience of the application?

### **Performance**

8. Are any reports generated?

*If yes? If no, go to step 9.*

- What is the amount of time that it takes to generate the report/s?

9. What is the reasonable amount of time that it takes for processes such as import, export, and updates agents to execute?

### **Must Sees**

10. What are the critical features of the application?

### **Documentation Required**

11. If available please get the following documents:

- Workflow of the application accompanied by a flowchart diagram, outlining the database issues such as security constraints, access levels, roles & responsibilities, etc.
- Business rules involved, critical application features and functions and/or any underlying assumptions/limitations
- Basic unit test plans/scripts

A few Advantages of the this approach-

Testing team prepares the necessary test document rather than the development team. That implies that the development team can concentrate on more productive development work.

Reduced test cycle and hence reduced cost.

Disadvantages of this approach-

This methodology largely depends on the expertise of the testers and their experience.

### **Summary**

An exploratory approach in testing Lotus Notes applications is known to reduce testing time and provide better test coverage though it depends largely on the expertise of the testers concerned.