SFA Modernization Program
United States Department of Education
Student Financial Assistance

Campus-Based Programs
Testing Requirements

Task Order #49
Deliverable #49.1.3

Table of Contents

1 Introduction

1.1 Purpose

The purpose of this Testing Requirements document is to provide an outline of the testing efforts that need to be completed to fully test the redesign of the Campus Based System (CB System). The testing requirements are designed to integrate with the Rational Unified Process, an iterative development process that will be used to develop the web-based portions of the CB System. A different approach to testing the back-end database and application conversions is also discussed. These requirements will serve as guidelines for the creation and execution of detailed testing plans during the development phase of the project.

1.2 Scope

The testing requirements encompass all phases in which testing of the CB System needs to be conducted. The tests range from unit tests conducted by developers when the modules are coded, to the final user acceptance tests. Other tests that will be conducted are the integration tests, which will test the interface between the web front-end application and the Oracle database and the interfaces between the database and any external system (GAPS, etc.). When the architecture of the parallel environment is finalized, there will be other interfaces with the legacy system that will also have to be tested. According to the Rational Unified Process, several iterations will occur to produce the final software product. As iterations are completed, a program executable is produced that has additional functionality from the previous iteration. For this reason, regression testing will also be performed to assure that as new functionality is added, old functionality is not lost. Performance testing is also required to ensure that the architecture can support the expected level of concurrent use.

Participants in the testing of the CB System will include web front end and application developers, individuals converting the current VSAM data storage to an Oracle database, the Campus Based Programs staff, SFA's Quality Assurance (formerly IV&V) testers, and users from select institutions.

2 Approach

There are three main efforts in the CB System Redesign: 1) development of a web-based FISAP process; 2) development of a web-based solution for the SFA staff; and 3) migration of the Campus Based data and application to a relational database and application server environment.

The testing efforts for both the web-based FISAP process and the web-based solution for SFA staff will employ the same process because of the iterative nature in which they are being developed. Testing of the web-based FISAP will be separate from the testing of the web-based SFA Staff solution because of differences in user groups and timing of development. The testing of the data migration and application redesign will vary because these are invisible to the user and a truly iterative process does not apply well.

2.1 Web-based FISAP Process

The web front end for the FISAP process will be developed, tested, and deployed to the production environment by August 31, 2001 so the institutions can begin submitting FISAP forms for the next year using the web-based FISAP. The web-based FISAP is being developed iteratively, therefore, after each iteration of development formal testing and institutional user focus groups will be held to evaluate the progress of the application.

2.2 Web-based Solution for SFA staff

The web-based solution for SFA staff is targeted to be developed, tested, and deployed to the production environment by November 30, 2001. This effort will also be using iterative development and formal testing and CB staff focus groups will be held after each iteration.

2.3 Database and Application Conversion

The entire database and allocation application is targeted to be converted, tested, and deployed to the production environment by November 30, 2001. The approach to testing the application and database conversion centers around running different samples of data through the allocation process on the legacy system and the redesigned CB application. When the final allocation numbers from the redesigned system match the legacy allocation numbers, the legacy system will be retired and the redesigned application and database will be used. While the complete conversion of the application and database will not be available until November 30, 2001, a subset of the database that is necessary for the FISAP process will be in production by August 31, 2001.

3 Assumptions

- The legacy system will exist in parallel for a period of time following the redesign effort and will be phased out by December 31, 2001.
- Rational Clearcase and Rational Clearquest tools will be available and configured for the development environment and will be used for Configuration Management.
- Rational Robot and Rational Test Manager will be available and configured for use in automating tests.
- Test scripts will be created to test the documented requirements of the CB System as outlined in the System Requirements Document deliverable submitted to SFA Management.
- Developers and testers will have connectivity to the system.
- Testing environments will be properly configured when testing commences.
- Four separate and identical environments exist for the application: development, test, staging, and production.
- There is a Configuration Management plan that outlines migration of code between development, testing, staging, and production environments.
- Development of the web-based FISAP and web-based solution for SFA staff are conducted following the Rational Unified Process.
- Performance / Stress testing will be outsourced to a Modernization Partner subcontractor.

4 Phases of Testing

This section describes the different steps in the testing process and the intent of each of the testing "phases". Each of these test phases described also specify the environment in which these tests will be conducted, the participant or person responsible for conducting the test, tools that will be used to conduct the test, resolution to unexpected results that occur, the criteria that will be used to evaluate when that phase of testing is complete, and the iteration(s) of development in which the testing effort will be conducted. Because the testing approach for the web-based development and the database and application conversion are different, separate sections exist for each.

4.1 Testing of Web-based FISAP and Web-based Solution for SFA Staff

The procedures for testing each of the web components are listed in this section. The approach to the testing of the web-based FISAP and the web-based solution for SFA staff is the same, however, they will be tested at different times, with each of the front ends following the procedure below.

4.1.1 Developer Unit Test

The Developer Unit Test is a test constructed and conducted by developers to validate that the functionality of modules meet requirements.

Environment: Beacon local development environment
Participant: Developers
Tools: Manually conducted with web browser
Error Resolution: If errors are encountered, the developer will modify the code to resolve the error.
Exit Criteria: The unit test for each module is considered successful when the developer verifies that coding requirements are satisfied.
Iteration: each iteration

4.1.2 Component Test

The Component Test is a more structured set of tests that is conducted to test each of the functional requirements developed in the current iteration release. The component tests run for the current iteration could be used as regression tests for the following iterations. By creating automated component test scripts, the task of regression testing will execute more quickly. The task of creating the automated tests will add approximately 15 minutes to each script, however, not having to manually re-run them for regression will produce greater future time savings. In addition to functional requirements, section 508 standards should also be applied to testing at the component level.

Environment: VDC test environment
Participant: Mod Partner Tester(s) and Developers
Tools: Rational Robot, Rational Clearquest

Error Resolution: If errors are encountered, the tester will log the error using Clearquest and the error will be assigned to a developer to be fixed.
Exit Criteria: The test scripts for each component test will be documented, reviewed, and approved by the testing lead prior to them being executed. The component test is considered successful when the approved test script is executed and all expected results occur.
Iteration: each iteration

4.1.3 Regression Test

The Regression Tests will verify that the new functionality added to the system in each iteration does not cause functionality from previous iterations to fail. Regression testing will consist of the previously recorded component test scripts being re-run.

Environment: VDC test environment
Participant: Mod Partner Tester(s) and Developers
Tools: Rational Robot, Rational Test Manager, Rational Clearquest
Error Resolution: If errors are encountered, the tester will log the error using Clearquest and the error will be assigned to a developer to be fixed.
Exit Criteria: The Regression test will be considered successful when all component test scripts re-run successfully.
Iteration: each iteration beginning with the second

4.1.4 Integration Test

The Integration test will be conducted to test the interaction between the web front-end application and the database back end.

Environment: VDC test environment
Participant: Mod Partner Tester(s) and Developers
Tools: Rational Clearquest
Error Resolution: If errors are encountered, the tester will log the error using Clearquest and the error will be assigned to the appropriate developer to be fixed.
Exit Criteria: The test scripts for each integration test will be reviewed and approved by the testing lead prior to them being executed. The integration test is considered successful when the approved test script is executed and all expected results occur.
Iteration: based on conversion of data from VSAM to Oracle database and implementation of functionality.

4.1.5 Performance Test

The Performance Test will be an automated test to simulate a pre-determined number of users using each of the web-based components concurrently. This procedure may be outsourced to a Modernization Partner subcontractor because of resource limitations that exist on the project.

Environment: outsourced
Participant: TBD contractor

Error Resolution: If errors are encountered in the stress test, they will be communicated to the project lead and appropriate actions will be taken.
Exit Criteria: The stress test will be considered successful when the requirement for concurrent users can be accurately simulated and system performance is within the specified requirements.
Iteration: after final iteration

4.1.6 Quality Assurance

The Quality Assurance (QA) tests are run by an independent group within the SFA CIO office (formerly IV&V) to assure quality in the software product. Access to the system documentation is required for the QA process to be completed.

Location: <*TBD*>
Participant: <*TBD subcontractor hired by SFA*>
Error Resolution: <*TBD*>
Exit Criteria: <*TBD*>
Iteration: <TBD>

4.1.7 User Test

The User Test is conducted by a select group of institutional and SFA staff users to simulate actual usage of the CB System. This test should verify that users can connect to the system and that functional requirements are met. These tests will also generate enhancement ideas that will either be implemented or gathered for the next release.

Location: Institution and / or SFA location
Participant: Campus Based steering committee or SFA staff steering committee
Tools: Email / Conference calls
Error Resolution: All errors and comments will be collected via email from the participants. Errors will be assigned to developers to fix and comments for improvement will either be implemented or collected for implementation in release 2.0.
Exit Criteria: N/A
Iteration: End of each iteration

4.1.8 User Acceptance Test

The User Acceptance Test (UAT) is a set of test scripts that will be run by individuals from SFA. When these steps execute correctly, SFA will agree to accept the application as ready for production. The UAT will not replace the VDC requirements or the Production Readiness Review process that is required for an application to go into production. The exact test scripts will be created and approved by SFA management prior to the UAT, based on all of the requirements of the system.

Location: <*TBD*>
Participant: individuals designated by SFA Management
Error Resolution: All errors will be reported to the Project Manager who will have the appropriate developer correct the error.

Exit Criteria: The test scripts, approved by SFA management, are executed without error. The appropriate SFA official has approved of and signed the Production Readiness Review document.
Iteration: End of final iteration

4.2 Database and Application Conversion Testing

This section primarily deals with the testing of the application conversion. The testing conducted for the database conversion as a standalone entity will entail verifying that all current and historical data stored in the VSAM files is ported over to the Oracle database. The Campus Based application conversion will be divided into functional components to be developed and tested during the development, however, official iterations will not exist.

4.2.1 Developer Unit Test

The Developer Unit Test is a test constructed and conducted by developers to validate that the functionality of modules meet design and functional requirements.

Environment: Beacon local development environment
Participant: Developers
Error Resolution: If errors are encountered, the developer will modify the code to resolve the error.
Exit Criteria: The unit test for each module is considered successful when the developer verifies that coding requirements are satisfied.
Execution: during the coding of each module

4.2.2 Component Test

The Component Test is a more structured set of tests that is conducted to test each of the functional requirements developed for a particular module.

Environment: VDC test environment
Participant: Mod Partner Tester(s) and Developers
Error Resolution: If errors are encountered, the tester will log the error using Clearquest and the error will be assigned to a developer to be fixed.
Exit Criteria: The test scripts for each component test will be documented, reviewed, and approved by the testing lead prior to them being executed. The component test is considered successful when the approved test script is executed and all expected results occur.
Execution: after each module is coded

4.2.3 Integration Test

The Integration test will be conducted to test the interaction between the allocation application and the database. Tests will also be conducted to verify the interfaces between the CB

System and the external systems such as GAPS function properly. Once the system architecture is finalized, all of the interfaces can be defined for testing. The interaction between the web front ends and the database will have already been tested, but it would be wise to re-test these.

Environment: VDC test environment
Participant: Mod Partner Tester(s) and Developers
Tools: Rational Clearquest
Error Resolution: If errors are encountered, the tester will log the error using Clearquest and the error will be assigned to the appropriate developer to be fixed.
Exit Criteria: The test scripts for each integration test will be reviewed and approved by the testing lead prior to them being executed. The integration test is considered successful when the approved test script is executed and all expected results occur.
Execution: after the application has been converted and each module has gone through component testing

4.2.4 Quality Assurance

The Quality Assurance (QA) tests are run by an independent group within the SFA CIO office (formerly IV&V) to assure quality in the software product. Access to the system documentation is required for the QA process to be completed.

Location: <*TBD*>
Participant: <*TBD subcontractor hired by SFA*>
Error Resolution: <*TBD*>
Exit Criteria: <*TBD*>
Execution: <TBD>

4.2.5 System Test

The system test will test the front to back functionality of the CB System. The entire allocation process will be simulated with the school user entering information, edits being kicked off, correction and explanation of edits, CB staff users using their web-based solution to modify data and run reports, and the allocations being made and communicated.

Location: VDC test environment
Participant: Mod Partner Tester(s) and Developers
Tools: Rational Clearquest
Error Resolution: If errors are encountered, the tester will log the error using Clearquest and the error will be assigned to the appropriate developer to be fixed.
Exit Criteria:
Execution: after integration testing for the application redesign is complete.

4.2.6 User Acceptance Test

The User Acceptance Test (UAT) is a set of test scripts that will be run by individuals from SFA. When these steps execute correctly, SFA will agree to accept the application as ready for production. The UAT will not replace the VDC requirements or the Production Readiness

Review process that is required for an application to go into production. The exact test scripts will be created and approved by SFA management prior to the UAT, based on all of the requirements of the system.

Location: <*TBD*>
Participant: individuals designated by SFA Management
Error Resolution: All errors will be reported to the Project Manager who will have the appropriate developer correct the error.
Exit Criteria: The test scripts, approved by SFA management, are executed without error. The appropriate SFA official has approved of and signed the Production Readiness Review document.
Execution: after system test has been completed

5 Level of Effort

The level of effort required to test the CB System will increase with each iteration. Functionality implemented in the current development iteration will be unit tested by the developers during the iteration and component tests will be performed at the end of the iteration. Regression tests of functionality implemented in previous iterations will also be performed after each iteration. Thus, as functionality increases with each iteration, so will the level of effort required to test.

It is expected that a testing lead and test analyst will be required to develop and execute a detailed test plan. Support from Beacon will also be necessary to make the testing effort successful.

6 Risks

- If an automated testing tool is not used for developing component and regression tests, as fixes are introduced to the system towards the end of development, there would be considerable time lost as regression tests are run manually.
- Without a Configuration Management plan, issues will arise with overwritten code, promotion of code to other environments, and resolution of errors.

7 Glossary

Iteration: Within the Rational Unified Process software development lifecycle, there are several iterations or steps toward the finished software product. Each iteration will result in an executable program with new functionality.

Rational Unified Process: Iterative software development process with a suite of integrated software tools to aid in development. The RUP has 4 phases of development: Inception, Elaboration, Construction, and Transition. This methodology is designed to mitigate risks by developing the riskiest parts of the application first. The majority of the development of the system is conducted in the elaboration and construction phases.

Section 508: Section 508 requires that Federal agencies' electronic and information technology is accessible to people with disabilities. Standards to meet these requirements have been published and any existing and new government computer applications must meet the standards. The standards for web based information and applications can be found at: http://www.section508.gov/docs/accessstandards.htm