

Tik-76.115 Software Project

Test Plan

Project: XSteel

Phase: delivery

Identifier:

XSTEEL-TESTPLAN-02

Version	Date	Author	Note
0.1	8 th , Dec, 2000	Yanling,Xiao	1 st Draft
0.2	1 th , Feb, 2001	Yanling,Xiao	2 st Draft
0.3	2 th , Feb, 2001	Yongjun,Zhang	Revised
0.4	19 th , Mar, 2001	Yongjun Zhang	Revised
0.5	22 th ,Apr,2001	Yanling , Xiao	Revised

Change Log:

1.

1.1 Purpose and Scope: draft document is changed into final document.

The sentence has been deleted: This document will be updated in later phases of the project when more detailed information about software itself will be available.

2.

1.3 Definition, terms and abbreviations: status of test plan of T4 and LU has been changed from Not yet available to revised.

1.Introduction

1.1 Purpose and scope

This is *final document* for the test plan of Xsteel Model-base Object Editor, organized by the course Tik-76.115 Software Project at Helsinki University of Technology. During this one-year's course, this document will be used as the guide to control and manage the whole process of product development and document all testing activities related to Xsteel project.

1.2 Product and environment

The project is to provide an add-on Model-based Object Editor subcomponent to enhance the productivity of the development of Xsteel application. The Xsteel developer can visually manipulate the UI components, define the component property and create desired file.

1.3 Definition, terms and abbreviations

The specific testing procedures to be carried out at each phase are described as following:

- Test plan for phase T2
- Test plan for phase T3 (revised)
- *Test plan for phase T4 (revised)*
- *Test plan for phase LU (revised)*

Please add all the abbreviations which could be found in the technical specification.

The following table shows the abbreviations to be used in this document, for all the abbreviations used throughout the whole project, please refer to the Functional Specification of Xsteel Object Based Editor.

UI	User Interface
GUI	Graphical User Interface
UOG	Xsteel User Object Generator

1.4 Reference

1. Tik-76.115 Project Plan of Xsteel
2. Tik-76.115 Requirement specification of Xsteel
3. Tik-76.115 Functional specification of Xsteel

2. Environment Requirements

During the whole project, programming and related tasks will be done at the following places:

- 1) team members' homes,
- 2) A218 in Computer Science building.

Xsteel Model-based Object Editor does not communicate with other Xsteel component in process level, nor it will load any existing Xsteel library. Thus the software environment is rather individual and standalone. The software is expected to work in Windows NT/2000. Windows 98 will be supported during the development stage, however, in typical Xsteel application environment, Windows 98 is not supported. As a result, the application will not be tested in Windows 98 or below system.

2.1 Hardware

The end user of this product will be developers of Xsteel Applications. The typical running environment of Xsteel is PC with Windows NT/2000, 256MB RAM and over 700MHz CPU. The programming Operating system is Windows NT/2000, and the general hardware environment is desktop PC. Network connection, LAN etc.

2.2 Software

Tool Visual C++ and Xsteel application platforms are applied in the testing this project.

2.3 Security

Because testing development workstation and environment are only used by this development project, there won't be problems with other people using the same machine. Also there won't be any special security requirements for the testing project.

2.4 Tools

Since CTC++ could only give the coverage of testing, which is not informative for this project. BlackBox testing is used instead of it. BlackBox testing tools, such as Winrunner or Rational Testing tool. TestDirector is also use for test reporting.

Other Special testing software tools are not needed in the project.

3. Staffing and training needs

The group with six people will be responsible for whole test process, no other persons are needed or available. In order to complete test with required criteria, the whole group will be familiar with tools and environment.

4. Overview of Xsteel testing

4.1 Goals and methods

The purpose of this document is to validate functionality, and make sure that it fulfills the requirement what customer needs.

Project member, Guo Heng and Xiao Yanling are responsible of planning the test cases and executing. Xsteel team will test the Xsteel system, Zhang Yongjun, Shi Hao and Lu Tao are responsible for implementing the project tasks in Xsteel. Project members are responsible for using Xsteel to carry out tasks ?

4.2 Responsibilities for test tasks

Quality Manager and Web Manager are responsible for planning and preparing of testing.

Responsibility for test execution shall be assigned between project members.

All the test cases must be executed at least twice, each execution must be done by different person. Each test case must be executed by at least two persons, one is the tester, and the other one or two are the witness. If the test results are the same, the test results are accepted, otherwise the test cases must be executed again.

Project Quality Manager is responsible for ensuring that tests are carried out according to the test plan. Xsteel Project Manager and Quality Manager also monitor the testing practices.

4.3 Test schedule

During phase T3, the project group will start to test the project tasks into the system and make test case for it. The customer will participate in the testing process during the delivering phase, that is, the last phase. The T4 phase will be focused on the Beta functional testing of Graphical User Interface Subsystem and Alpha Testing of User Object Generator, and the customer could not be asked to take part in the product testing. During the delivering phase, the customer will be invited to assist us in setting up the Xsteel testing environment and performing the acceptance testing.

4.4 Test platform

Customer tests will be carried out in the PC platform provided by customer.

5. Overview of test methods

5.1 Internal tests

Internal testing will be performed during phase T2.

5.2 Regression tests

The goal of this phase is to verify that all defects found have been corrected and produce a final report on the testing activities.

5.3 Integration tests

The system will be installed into a customer-specified platform in the phase T4, integration testing will be carried as well in this phase, this is to say, component integration and interoperability will by necessity is integration-tested. The system prototype will be implemented and available for customer testing.

Before customer starts to test it, Xsteel prototype will be demonstrated, and a draft of User Guide will be provided as well. Other necessary testing material will also be provided.

5.4 Functionality tests

System functionality of Xsteel will be tested according to test cases , which Xsteel users must be able to execute with the system, and finding any possible conflicts or errors.

5.5 Security tests

Security tests are not existed in this project.

6. Testing methods

The results of each test case will be described as the following:

- Fatal errors. Fatal errors are which prevent important system function working out.
- Serious errors. System errors are not working according to requirement specification.
- Minor errors. System functions are not working completely according to project plan and requirement specification, but it works around.

7.Risk management

Risk Definition	Probabilty (0-10)	Impact (0-10)	Weight (0-10)	Contingency
Lack of Time	6	8	48	Weekend working and better personal time management.
Lack of Personnel	4	9	36	Commitment to this project should be stressed.
Lack of critical Information.	7	7	49	Consulting the course assistant and teacher, surfing course's web page and news group constantly.
Misunderstanding of customer's requirements.	5	10	50	Meeting the customer frequently, asking the customer to participate in the requirement definition and project planning.
Language Barrier	8	5	40	Consulting Fins for the important information of this course.
Lack of programming experience.	7	8	56	Arranging time for studying, and discussing programming problems with other group members.
Scope of project is too large.	6	7	42	Negotiating with the customer, and defining the requirements clearly.
Project is badly segmented.	8	7	56	Designing the product architecture carefully, reviewing it with the supervisor.
Wrong tools	6	7	42	Studying the available tools, comparing their characteristics and consulting experts.
Unfamiliar technology used	4	7	28	Arranging fixed time to study VC++ technology
Delivery passed the deadline.	4	5	20	Reading the news group and web page every day. Keeping the projects schedule.

In order to avoid or reduce risks, the following methods are used:

Each member of the group should put test in priority position among their other things, ensure each phase to be completed within schedule, plan next phase tasks in plenty of time.
Being familiar with test tools and environment as early as possible.

With limited resource, the member of group should start test work as early as possible, if necessary, increasing night work, excluding some tests of minor importance.

8. Acceptance and rejection criteria for the tests

The test will be suspended in case of hardware failure or operating system crash. The test case, which is not completed, will be repeated when the test is resumed.

In case that incomplete test case needs some data from the other test cases, those related test cases will be repeated as well.

The test memo will be used to record the test cases execution, and the suspended test case execution will be recorded in test memo

When the system is finally delivered to the customer, it should not contain unsolved fatal and serious errors.

The fatal and serious errors, which the project group estimates to be too cumbersome to correct, should be presented to the customer and project instructor, together with a suggestion of an alternative interpretation of the requirements, or an implementation change.

Testing can be temporarily suspended, if a major functional deficiency or problem with the implementation or test environment is found. This should always be immediately reported to the customer and the project supervisor, and a schedule for solving the reason for the suspension and re-continuing testing provided.

Continuing testing after correcting an environment problem needs no special procedure. However, after correcting a major functional fault which was the cause for the suspension, the impact of the correction to the system should be assessed, and decided, which of the previously carried out and passed test cases need to be re-executed.

Test phase can be considered finished, when all test cases that were scheduled are listed as following:

1. passed without any Fatal or Serious errors,
2. The functionality to be tested in the next test phase is implemented and ready to be tested.

In the latter case, all unsolved Fatal or Serious errors should be re-scheduled to be tested within this later test phase.

There is no special procedure related to test case acceptance. Every member who is testing the software is able to determine whether test case is properly executed and accepted. Quality manager will have overall responsibility of all test cases and he will check and accept all executed test cases. According to produced testing document, entire testing is accepted by the client. Since this is not a real testing project, acceptance of testing is not as critical as it might be in normal development projects.

This document was created with Win2PDF available at <http://www.daneprairie.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.