TEST PLAN (TP)
FOR

Phunctional UML Editor
(pUML)

Version 1.0
April 3, 2012

Prepared for:
Dr. Clint Jeffery

Prepared by:
Josh Armstrong
Zach Curtis
Brian Bowles
Logan Evans
Jeremy Klas
Nathan Krussel
Maxine Major
Morgan Weir
David Wells

University of Idaho
Moscow, ID 83844-1010

CS384 TPD
RECORD OF CHANGES (Change History)

Change Number	Date completed	Location of change (e.g., page or figure #)	A M D	Brief description of change	Approved by (initials)	Date approved
01	3/27/2012	Test Plan	A	Document Created	MM	3/27/2012

A - ADDED M - MODIFIED D – DELETED

pUML
TABLE OF CONTENTS

Section Page

1 TEST PLAN IDENTIFIER

This is test plan No. 007.

The pUML project team is not associated with an established company at this time, and this will be the only unique numerical identification number for any of our products.

2 REFERENCES

- Systems and Software Requirements Specification (SSRS) ver. 0.0
- System and Software Design Description (SSDD) ver. 0.0

3 INTRODUCTION

The purpose of this Test Plan is to ensure the integrity of the pUML software through a well-defined series of tests. The testing outlined in this plan will be applied to each component of the pUML software. Errors will be well-documented and each test will require a follow-up so all changes or recommended improvements to both functionality and features may be applied to this software.

The testing required will include manual testing, unit testing, and a combination of both and/or other specialized tests as necessary for each test item. results for each test item will be logged. action taken on each test item will be logged The purpose for this test is to ensure that all errors are found and handled.

4 TEST ITEMS

Items to be tested include:

- Installers
- Multiplatform portability
- Legality
- Functions and parameters
- Excessive code complexity
- Large program components, i.e. window and canvas class
-

5 SOFTWARE RISK ISSUES

Software risk areas include extremely complex functions and modifications made on components with a history of failure. These functions and components are itemized as follows:

- QPaint. This introduces a complex hierarchy of classes and functions. To ensure this is adequately functional, manual testing will be required.
- Compilation issues.
- Restoring object classes. This tends to be fragile.

6 FEATURES TO BE TESTED

Features to be tested include alll objects, connectors, and associated functionality, Open/Save/Restore functionality, and Diagram types load properly.

6.1 MENU/OPTIONS MENU/MAINWINDOW

- Options window appears at pUML startup.
- When options window is closed with no selections made, the pUML main window remains.
- When "New" is selected, options window appears.
- When "Open" is selected, an explorer window appears.
- When "Save" is selected ...
- When "Save As.." is selected...
- When "Print" is selected...
- When "Import/Export" is selected...
- When Exit is clicked...
-

6.2 OBJECTS

All objects should perform or have the following tasks performed on them in a consistent manner in the following cases:

- Objects can be created.
- Objects can be deleted.
  - Any associated connectors must also be deleted.
  - All associated text and descriptions must also be deleted.
- Objects can be selected.
- Objects can be de-selected.
- Objects can be moved after they have been placed
  - Objects moved retain the new location with the same integrity as the previous location.
  - All text and descriptions associated with the object move with the object to the same location in relation to the object.
- Objects can accept an indefinite number of connection lines.
- The object's properties may be modified.
  - Description or contents may be modified an indefinite number of times.
- The object's size may be modified...
  - Manually.
  - Auto-fit to contents.

6.3 CONNECTORS

All connectors should perform or have the following tasks performed on them in a consistent manner in the following cases:

- Connectors can be created.
  - Only one connector may exist between any two objects.
  - Only one self connector may exist per object.
- Connectors can be deleted.
- Connectors can be selected.
- Connectors can be de-selected.
- Connectors cannnot change the objects they connect other than the two selected once they have been placed.
- If an object a connector is attached to moves, the connector's endpoint moves with that object.
  - Connectors moved retain the new configuration with the same integrity as the previous configuration.
  - All text and descriptions associated with the connector move with the connector to the same location in relation to the connector as before.
- The connector's properties may be modified.
  - Line description may be modified an indefinite number of times.

6.4 OPEN

Once the pUML software has been launched, files may be opened if the following criteria are met:

- The file type is of pUML type.
- The file name is legitimate.
- The file is not already open in pUML.

Once a file has been opened:

- The diagram is loaded into a new tab.
- Toolbar is updated with that diagram's objects and connectors.
- All objects and connectors in the file are in the exact same state and configuration as when the file had been previously saved.
- All text descriptions are preserved from previous save.
- All features are modifiable
  - Objects may be added or removed, and retain all object functionality.
  - Connectors may be added or removed, and retain all connector functionality.
- The file may be saved.
- The file may be closed.

6.5 SAVE

The save function may be invoked if the following criteria are met:

-
-
-

6.6 RESTORE

6.7 DIAGRAM TYPES

6.8 TAB FUNCTIONALITY

6.9

6.10

7 FEATURES NOT TO BE TESTED

[Insert text here.]

8 APPROACH

[Insert text here.]

9 ITEM PASS/FAIL CRITERIA

[Insert text here.]

10 SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS

[Insert text here.]

11 TEST DELIVERABLES

[Insert text here.]

12 REMAINING TEST TASKS

[Insert text here.]

13 ENVIRONMENTAL NEEDS

[Insert text here.]

14 STAFFING AND TRAINING NEEDS

[Insert text here.]

15 RESPONSIBILITIES

Who is in charge? This issue includes all areas of the plan. Here are some examples:

- *Selecting features to be tested and not tested.*
- *Ensuring all required elements are in place for testing.*

[Insert text here.]

16 SCHEDULE

[Insert text here.]

17 PLANNING RISKS AND CONTINGENCIES

[Insert text here.]

18 APPROVALS

[Insert text here.]

19 GLOSSARY

[Insert text here.]

20 APPENDIX A. [insert name here]

Include copies of test examples, etc. supplied or derived from the customer. Appendices are labeled A, B, . . . n. Reference each appendix as appropriate in the text of the document.

[insert appendix A here]

21 APPENDIX B. [insert name here]

[insert appendix B here]