GAM150 Team Exercise

Team Exercise: Testing, Bug Reporting and Debugging

Introduction

The objective of this team exercise is to familiarize teams with several software development and testing processes. During the exercise teams are required to identify, communicate, fix and retest a variety of issues within a given project. Additionally, teams are required to use Trac (or other approved software) to communicate issues between members in a manner similar to that used by development and test teams.

Instructions

Setup

- Divide your team into two groups:
    - Dev Team – Technical Director plus 1 or 2 others.
    - Test Team – Test Manager, if available, plus 1 or 2 others.
    - Dev team should be the same size or larger than the test team
- Physically separate the two groups, if possible.
    - Separate rows or separate ends of a row.
- Use Trac, SVN (or Git/Mercurial), and Visual Studio as the primary tools for this exercise.
    - The use of other Version Control System (VCS) or Project Management (PM) software is permitted with instructor approval.
- Verify that all members of the team have access to the team VCS repository.
- ***Read all instructions thoroughly.***

Dev Team

- Download the team exercise files from Moodle.
- Unzip the project to a clean location.
- Add and commit the project to the team's VCS.
    - In SVN, the recommended location for projects is in the "trunk" folder.
    - **Do *not* commit files generated by Visual Studio.**
- No changes may be made to the project unless a ticket exists in Trac that identifies the bug or the change that is required.
- Investigate any issues reported by the test team (using Trac), address any valid defects, and submit a new build to SVN. If you have questions regarding a ticket, please use Trac to obtain additional information. Any changes made to the project without a supporting Trac ticket may result in a penalty to the assignment grade.
- Read the instructions in Main.c for additional objectives of the exercise. Any tasks required to meet these additional objectives should be entered into a Trac ticket before any changes are made.

GAM150 Team Exercise

Test Team

- Download the project from SVN onto one, or more, computers.
- Use Visual Studio to build the project and begin testing for any defects.
- Report any defects found, including error or warning messages, to the Dev Team using Trac.
  - How much information is required to effectively communicate the issue?
  - What is the type of defect? What is its severity/priority?
  - Please avoid verbal discussions with the dev team regarding defects. Simple statements between the two teams, like "build's up" or "new bug submitted" are fine.
  - Respond to any requests for information via Trac
- When a defect is reported as fixed by the Dev Team.
  - Download, build and test the new build.
  - When a defect is verified as fixed mark it "closed" in Trac.

Debugging Techniques

- Single Step (hotkeys: <F10>, <F11>)
  - Useful for watching what code is *actually* doing
- Breakpoints
  - Useful for quickly running through code, such as loops
- Breakpoints on memory writes
  - Useful for detecting memory corruption and buffer overruns
  - In Visual Studio: "DEBUG/New Breakpoint/New Data Breakpoint"
- View Memory
  - In Visual Studio: "DEBUG/Windows/Memory"

Submission Requirements

Make sure that the following steps have been performed before submitting:

- Clean the project using the appropriate option in Visual Studio (VS).
- Verify that any executables and intermediate files (such as .o, .sdf, .filters, etc.) have been deleted
- Include a file (.doc, .odt or .txt) containing a list of the tickets created in Trac showing, at a minimum, each ticket's description and comments. The contents of this file will be used to determine whether or not the team properly used the software for communicating issues, as outlined in the exercise instructions.
- Zip the project files at the project folder level (.zip format only)
- Name the .zip file according to the following naming convention
  - (team ID)_GAM15x_(login)_TeamExercise.zip
    - Example: A05_GAM15x_john.doe_TeamExercise.zip
- Submit the .zip file to the Moodle page for your individual section.
- Download the .zip file to a clean location and verify that the submitted project builds and executes properly.