

Developing an Automated Fuzz Testing Framework for Windows Embedded Handheld Applications

By

Nizam Abdallah

B.AppSci in Mathematics and Computing
Swinburne University of Technology, Melbourne, Australia

GradCert Eng in Simulation Technology
Royal Melbourne Institute of Technology, Melbourne, Australia



Submitted in fulfilment of the requirements for the degree of

Master of Information Technology

March 2011

Supervisor: Dr. Sita Ramakrishnan

Clayton School of Information Technology

Monash University

Abstract

Mobile devices have continuously had an increasing positive impact on our lives by providing not only the ability to communicate, but also work, watch movies, listen to music, play games and keep in touch with family and friends. As a result software developers need to ensure that the mobile software developed is robust, reliable and secure to use by a variety of users operating a variety of different handheld devices. How users interact with mobile GUI applications is different to how they may interact with desktop GUI applications. On the desktop the main ways to interact with applications is via a keyboard and mouse. On a device, much of the interaction is performed via a touch screen, hardware buttons and gestures including multi-touch and accelerometer functionality.

Software test automation has proven to be a useful technique in making the testing process more cost efficient and has a positive impact on the quality and robustness of the software being developed. Furthermore, fuzz testing is another testing technique that has increased in popularity over the last 20 years and has proven to be successful in finding defects by sending random input data to an application and determining if the application will crash. In this research, an automated fuzz testing framework was developed that can be used to test applications targeting the Windows Embedded Handheld devices. The main contribution is *Torqueo*, an automated fuzz testing framework that is capable of running on any device running the .NET Compact Framework. *Torqueo* is a novel idea, as it allows software teams to perform various types of fuzz testing, including GUI fuzz testing on a number of mobile devices. GUI fuzz testing is performed using 1) the Win32 API to simulate keyboard and stylus input, or 2) .NET reflection to detect and invoke GUI controls directly. In addition to the development of the fuzz testing framework, two test tools were developed that make use of the functionality provided by the framework. The first tool was a standalone client that runs on the device, while the second tool was a client / server agent that also runs on the device and receives test commands over a TCP/IP connection from a desktop client.

Test tools are usually intrusive when performing tests on an application and negatively impacts the performance on the hardware running the application under test; this is especially true on an already constrained mobile device. Therefore, in this research, the two different GUI invocation techniques were compared to determine which technique has a greater performance impact on the memory usage on the device. It was discovered that using .NET reflection consumes more memory when used to randomly invoke controls on GUI forms that perform more CPU intensive operations, such as file IO and database IO. Furthermore, the memory used by the two test tools to execute tests was also compared. It was discovered that the client / server agent used less memory than the standalone client.

Finally, a common way of reproducing defects in random testing is to log the actions and data used during a random tests session and replay the same actions using the same data in an attempt to reproduce defects. However in this research, an algorithm is proposed that assists in the reproduction of defects on various GUI forms by generating test cases based on the sequence of GUI controls randomly invoked during the original test session.

Vita

Abdallah, N. (2010, 12). *Performance Impact of Using .NET Reflection in .NET Compact Framework Applications*. Retrieved 12 22, 2010, from Monash University - Clayton School of Information Technology Publications: <http://www.csse.monash.edu.au/publications/2010/tr-2010-260-full.pdf>

Abdallah, N., & Ramakrishnan, S. (2009). Automated Stress Testing of Windows Mobile GUI Applications. *International Symposium on Software Reliability Engineering (ISSRE)*. Mysore, India: IEEE, ISSRE.

Abdallah, N., & Ramakrishnan, S. (2011). Torqueo: Automated Fuzz Testing for Windows Embedded GUI Applications. *11th International Conference on Quality Software*. Madrid, Spain, Submitted for review.