

J2ME 기반 모바일 GUI TS(Test Supporter)

황선명*, 김상수*

*대전대학교 컴퓨터공학과

sunhwang@dju.ac.kr, bboys10@naver.com

TS(Test Supporter) for GUI of Mobile in J2ME Platform

Sun-Myung Hwang*, Sang-Su Kim*

*Dept of Computer Engineering, Dae-Jeon University

요 약

1) 모바일 응용 S/W GUI를 기존의 매뉴얼을 보고 테스트 하는 단계적이고 수동적인 방법을 탈피하여 사용자 중심의 시나리오를 통하여 자동으로 GUI를 테스트 하는 것을 목적으로 한다. 제시한 테스트 방법은 사용자 중심의 UI를 설계하여 간편하고, 정확하게 테스트 할 뿐만 아니라 테스트케이스를 재사용하는 환경을 지원한다. 제안한 자동화 도구는 시나리오 기반의 정확한 테스트와 재사용성을 높인 테스트 도구로서 기존 테스트 도구와 기능적인 비교를 통하여 성능을 확인하였다.

으로 이루어져 있으며, GUI 테스트 지원기를 실제 구현하여 테스트 하는 방법과 결과에 대하여 설명하고, GUI 테스트 지원기를 적용하였을 때의 적용평가에 대하여 서술하였다.

[표 1] 모바일과 데스크탑 S/W 테스트 비교

요소	모바일	데스크탑
테스트 기록	GUI 이벤트 기록을 위한 기능을 제공하지 못해, 사용자 인터렉션에 대한 에뮬레이션 의 불가능	GUI 이벤트 기록을 위한 java.awt.Robot 제공
배치 자동화	디바이스 상에 배치와 테스트의 지루한 수작업	배치가 완전히 자동화되어 있고, 테스트도 비교적 자동화되어 용이함
테스트 환경 차이점	디바이스 상에 차이점이 있고, 모든 디바이스 상에서 테스트가 요구됨	개발과 배치/ 테스트가 동일 하고, 특정 OS에 의존하는 경우가 거의 없음
API	API 표준화가 아직 미성숙	API 표준이 성숙되어 있고, 다양한 벤더로부터 JVM 이식 가능

1. 서론

현재 모바일 어플리케이션 개발자들은 클라이언트-서버 및 웹 어플리케이션 개발자와 달리 매우 엄격한 제약사항(메모리, 화면크기, 입력장치 등), 짧은 생명주기, 심한 사용자 편의성에 대한 요구사항 등에 직면해 있다.[1] 신뢰성 있는 모바일 어플리케이션을 개발하기 위해서 개발 초기에 S/W 결함을 최대한 많이 발견하여 S/W 품질을 대폭 향상할 수 있는 방법이 필요하다. 또한 모바일 어플리케이션의 다양한 GUI에 대한 테스트 방법과 구체적인 테스트 케이스 생성 기법이 요구된다.[2][3][4][5]

모바일 S/W는 데스크탑 S/W보다 단순하고, 크기가 작다. 모바일 환경은 단순한 JVM과 H/W 계층인 디스플레이 장치, 네트워크 포트 등에 접근하기 위한 API를 제공한다. 이러한 환경에서 프로그래밍과 테스트는 서버 환경보다 더욱더 어렵다. 휴대폰은 각자 다른 H/W 환경, 화면크기와 칼라수, 메모리 크기와 전력 등을 가지고 있다. 또한 같은 어플리케이션이라도 설치되는 타겟 디바이스에 따라 조금씩 다르게 동작 할 수도 있다. 모바일과 데스크탑 S/W 테스트의 차이점은 [표 1]과 같이 요약된다.[6][7]

따라서 본 논문의 구성은 GUI 테스트 지원기의 내부 구조와 각각의 컴포넌트들에 대한 설계 및 설명

2. 관련연구 J2ME Platform

J2ME 비록 휴대폰이나 임베디드 기기를 위한 플랫폼이지만, 기본적으로 자바가 가지고 있는 특징인, 객체지향방식의 프로그래밍, 코드의 높은 이식성, 안

1) 이 논문은 2008년도 정부(교육과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임(No.R01-2008-000-20607-0)

전한 네트워크 보안 지원 및 J2SE와 J2EE와의 상호 호환성은 그대로 유지하고 있다. 여기에 더하여 J2ME는 다음과 같은 특징을 지니고 있다. [4][5][8]

가. 다중플랫폼호환성(cross-platform compatibility)

100% 순수한 J2ME API를 사용하는 애플리케이션은 쉽게 다른 업체가 만든 모델과 광범위하게 호환될 수 있다. 단말기가 다르다고 해서 프로그램 개발자 측에서 같은 프로그램을 다시 개발하는 일은 상당히 번거로운 일이다. 하지만 J2ME 플랫폼은 어느 단말기에서든지 시간과 장소에 관계없는 강력한 호환성을 가진다.

나. 보안성

무선 인터넷 환경에서의 보안은 아직도 해결되지 않는 문제점들이 많다. J2ME는 기존 자바의 보안 모델을 무선통신에 적용시킴으로써, 이를 상당부분 해결하고 있다. J2ME로 작성된 애플리케이션은 기본적으로 디바이스의 하드웨어나 다른 리소스에 직접 접근할 수 없기 때문에 바이러스나 다른 악성 프로그램을 만들어낼 수 없다.

다. 동적 애플리케이션 다운로드

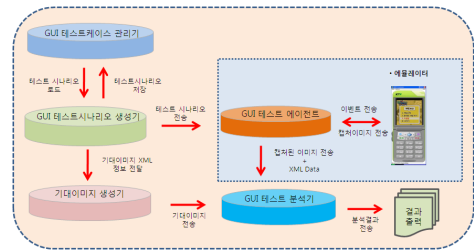
무선 서비스를 통한 자바 애플리케이션들은 실시간적으로 동적으로 다운로드 되므로 사용자들은 A/S 센터를 방문하고나, 업그레이드를 쉽게 받을 수 있다.

고 실행 결과를 입력받는다.

라. GUI 테스트 에이전트 : 모바일 S/W를 GUI 테스트케이스에 따라 실행하도록 한다. 모바일 S/W와 함께 빌드하여 에뮬레이터 또는 모바일 디바이스 상에서 실행된다. GUI 테스트 케이스 관리기에서 보내는 이벤트를 받아서 모바일 S/W를 동작시킨다.

마. 기대이미지 생성기 : GUI 테스트케이스에 포함되는 기대이미지를 만든다. 기대이미지는 응용 S/W에서 이벤트를 발생시킬시 출력되는 화면을 미리 정의해 놓은 것이다.

3.2 GUI TS(Test Supporter) 구조



[그림 1] GUI TS(Test Supporter) 구조

[그림 1]은 GUI TS의 전체적인 구조와 각 컴포넌들이 주고받는 이벤트들을 보여주고 있다.

3. GUI TS(Test Supporter) 기능 및 구조

3.1 GUI TS(Test Supporter) 기능

GUI TS(Test Supporter)를 구성하는 컴포넌트의 기능은 다음과 같다.

가. GUI 테스트 시나리오 생성기 : GUI를 테스트하기 위한 시나리오를 작성한다. 테스트 시나리오에는 테스트 개요, 테스트 목적, 주의 사항과 함께 GUI 테스트케이스로 구성된다.

나. GUI 테스트 분석기 : GUI테스트케이스의 기대 이미지와 실행 결과화면을 분석한다.

다. GUI 테스트케이스 관리기 : GUI테스트케이스를 관한다. 에뮬레이터 EH는 모바일 디바이스에서 동작하는 모바일 S/W와 연결하여 S/W를 실행시키

4. GUI TS 테스트 주요 방법

4.1 화면 상태정보 생성방법

GUI를 자동으로 테스트하기 위해서는 각각의 이미지마다 정보를 가지고 있어야 하는데 이 정보는 XML 형식으로 저장한다. [표 2]은 각 이미지에 대한 XML 스키마이다. 한 화면에 대한 이벤트 타입과 이벤트 값, 그리고 전,후의 이미지 값을 가진다.

[표 2] 한 화면의 XML 스키마

```
<item>
  <itemid="S001001000">
    <event          eventType="KeyClick"
eventValue
      ="down", nextItem="S001002000"/>
    <event          eventType="KeyClick"
eventValue
      ="up", nextItem="S001003000"/>
    <event          eventType="KeyClick"
eventValue
      ="enter", nextItem="S002000000"/>
  </item>
```

[표 2]에서 보면 화면 ID가 “S001001000”인 화면이 키 이벤트 ‘down’ 값이 들어오면 화면 ID값이 ‘S001002000’인 다음 화면을 출력하고 ‘up’키를 누르면 ‘S001003000’인 화면을 출력하게 된다. 화면에서 동작하는 모든 키 이벤트 값을 정의해 주고 해당 이벤트 값에 따른 다음 화면 ID를 지정해 준다. 이벤트가 들어왔을 경우 원본 이미지 서버에서 해당 이미지를 자동으로 불러오게 된다.

4.2 GUI 테스트 방법

GUI 테스트 시나리오 생성기에서 시나리오를 생성하면 시나리오에 포함된 각 화면의 정보들이 GUI 테스트 분석기를 통하여 테스트를 하게 된다. GUI 테스트 분석기는 각 화면에 포함된 XML 정보를 비교·분석하여 테스트를 하게 된다.

테스터가 테스트 시나리오를 작성한 후 테스트를 실행하면 GUI 테스트 지원기는 테스트 시나리오 흐름대로 이벤트를 전송하여 테스트가 자동으로 시작된다. 에뮬레이터는 내부 실행으로 사용자의 눈에 보이지 않게 된다. 테스트 시나리오의 화면 상태, 이미지, 이벤트 값 등의 정보가 테스트 에이전트를 통해 에뮬레이터로 들어온 후, 에뮬레이터는 정해진 시나리오의 순서에 따라 구동된다. 이때 에뮬레이터는 S/W를 동작시키는 동시에 테스트 에이전트를 통하여 테스트 이미지 서버에 수행된 이미지를 캡처하여 저장한다.

그리고 기대이미지 서버는 이미지와 XML 정보들을 바탕으로 분석을 하게된다. 먼저 화면 상태를 비교한 후, 화면 상태가 정보가 일치하면 이미지를 비교한다. 이미지는 픽셀비교를 통해 정확하게 오류를 검출해 낼 수 있다.

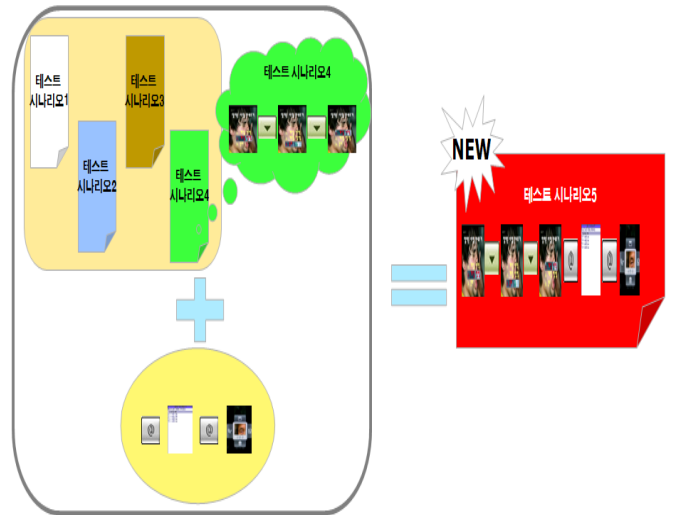
[그림 2]에서처럼 우선 캡처 이미지의 XML정보와 기대이미지의 XML정보를 비교한 뒤 두 이미지의 픽셀값을 비교하여 오류를 검출한다.



[그림 2] 오류검출 시 비교화면

4.3 테스트 시나리오 재사용 방법

사용자가 테스트를 하는데 있어서 기존의 테스트 시나리오를 재사용한다면 많은 시간과 자원이 절약될 수 있을 것이다. 이와 같이 GUI 테스트 지원기도 사용자가 기존의 테스트 시나리오를 이용하여 새로운 시나리오를 작성할 수 있다. 테스트 시나리오 재사용 방법은 아래 [그림 3]과 같다.

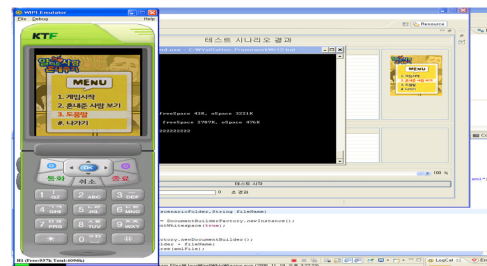


[그림 3] 시나리오 재사용 방법

이 방법은 저장해 놓은 시나리오에 사용자 직접 입력 방법을 병행하는 것이다. 저장해 놓은 시나리오를 불러온 후 사용자가 새로 작성하고자 하는 테스트 시나리오에 맞게 이벤트와 이미지를 끼워 넣음으로써 재사용성을 확보하고 반복적인 작업을 탈피하는 장점이 있다.

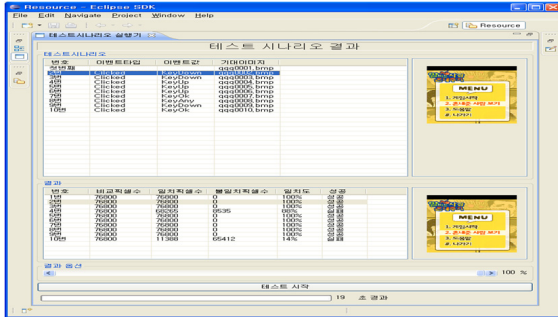
5. 테스트 수행

테스트 프로그램은 이번 실험을 위하여 제작한 간단한 프로그램을 대상으로 한다. ‘알미운 사람 혼내주기’라는 모바일 응용 S/W로 핸드폰으로 찍은 사진이나 다운로드 받은 사진을 타겟으로 지정하여 간단한 키패드 조작으로 알미운 사람을 혼내주는 프로그램이다.



[그림 4] 테스트 수행화면

[그림 4]는 TS를 실행한 뒤 정해진 시나리오 순서대로 에뮬레이터를 실행시키면서 이미지를 캡처하고, 그 이미지와 기대이미지를 비교하고 있는 그림이다.



[그림 5] 결과화면

[그림 5]는 테스트 결과화면인데 각 단계마다 기대 이미지 리스트가 나오고, 프로그램이 실행되면서 나온 캡처화면과 비교하여 픽셀을 비교하여 일치도를 백분율로 표시하고 테스터가 정한 기준 퍼센트를 넘으면 테스트성공으로 그렇지 않으면 실패로 표시되는 그림이다.

5. 결론

본 논문에서는 "모바일 응용 S/W GUI 자동화 테스트 방법 및 도구"를 제안하였다. 모바일 응용 S/W GUI를 간단한 방법으로 GUI의 오류 유무를 판별할 수 있으면 테스터 입장이 아닌 개발자 입장에서 테스트함으로써 구현 단계에서부터 오류를 줄일 수 있다. 또한 공개S/W인 이클립스(Eclipse)를 플랫폼으로 채택함으로써 대기업이 아닌 중소기업 및 벤처기업 등에서 부담 없이 사용할 수 있다는 장점이 있다.

현재 GUI 테스트 지원기는 SKVM, KTF_AROMA 등의 에뮬레이터에만 한정되어있고, 또한 터치폰 등의 GUI는 지원을 하지 못한다. 앞으로 모바일 응용 S/W GUI는 사용자와 교감할 수 있는 시스템으로 변환해 갈 것이다. 이것의 대표적인 예가 오늘의 터치 폰들을 들 수 있다. 사용자의 행동에 즉각적으로 반응하여 진동 등으로 표출함으로써 사용자가 교감을 느낄 수 있다.

향후에는 이러한 차세대 기능 및 다양한 플랫폼 등을 적용할 수 있도록 향후 연구가 필요할 것이라 사료된다.

참고문헌

- [1] "MOPAD-I-MASTT 상세설계서" 한국전자통신연구원(2007-S032-01, 다중 플랫폼 지원 모바일 응용 S/W 개발환경 기술 개발)
- [2] 채현철, 이정주, 황선명, 정양재 "모바일 응용 S.W GUI의 자동화 테스트를 위한 도구 구현", 한국정보처리학회 제15권 제1호, 2008. 11.
- [3] 채현철, 황선명 "모바일 응용 S.W GUI의 자동화 테스트 및 관리를 위한 도구 설계 및 구현", 한국정보처리학회 제15권 제2호, 2008. 5.
- [4] 정일재, "시나리오기반의 모바일 응용 소프트웨어 GUI 테스트 방법에 관한 연구", 대전대학교 일반대학원 석사학위논문, 2008. 2.
- [5] 박상필, "이미지 플로우 기반의 모바일 GUI 테스트 도구에 관한 연구", 대전대학교 일반대학원 석사학위논문, 2008. 2.
- [6] 윤석진, 김철홍, 신규상, "J2ME 기반 모바일 응용 S/W의 단위 테스트 도구 개발", 한국정보처리학회 춘계학술발표대회 논문집 제14권 제2호, 1033-1035, 2007. 11.
- [7] 채현철, 황선명, 김철홍 "TDD기반의 모바일 단위 테스트 방법 및 개발에 관한 연구", 한국정보처리학회 춘계발표대회 논문집 제14권 제2호, 2007. 11.
- [8] Sun-Myung Hwang, Hyeon-Cheol Chae, "Design & Implementation of Mibile GUI Testing Tool", ICHIT2008 International Conference, 2008. 7.
- [9] 이상윤, 김선자, 김홍남, "한국 무선 인터넷 표준 플랫폼(WIPI)의 표준화 현황 및 발전 방향", 정보과학회지 제22권 21호 통권 제176호, pages 16-23, 2004. 01.
- [10] 김문겸, "모바일 게임GUI 디자인에 관한 연구", 청주대학교 대학원 석사학위논문, 2006. 12.
- [11] 홍준성, "모바일 플랫폼의 기술현황 및 발전방향", 정보과학회지 제22권 21호 통권 제176호, pages 8-14, 2004. 01.