

다양한 모바일 플랫폼을 위한 모바일 애플리케이션

이식성 테스트 평가 모델 설계*

박해윤^o, 김성철, 최병하, 최종무, 유해영
 단국대학교 컴퓨터학부

{park8312^o, sungchel, notanything, choijm, yoohy}@dankook.ac.kr

Design of Portability Test Model

for Evaluating Mobile Application on various Mobile Platform

Haeyoon Park S.C. Choi B.H. Kim J.M. Choi Haeyoung Yoo

Department of Computer Science and Engineering, College of Engineering, Dankook University

요 약

최근 스마트폰과 같은 다양한 애플리케이션을 사용할 수 있는 모바일폰의 유행에 힘입어 모바일 애플리케이션 수요가 폭발적으로 증가하고 있다. 이에 따라 모바일 애플리케이션에서의 품질의 중요성도 점차 커지고 있다. 하지만 모바일 애플리케이션은 다양한 플랫폼이나 모바일폰의 제약사항에 따라 호환성이 떨어지는 특성을 가짐으로써, 다른 플랫폼에서 애플리케이션을 재사용시 해당 플랫폼이나 모바일폰의 제약사항에 따라 다시 개발해야하는 제약을 가지게 한다. 이런 재개발은 플랫폼의 특성이 플랫폼 별로 상이함에 따라 개발에 어려움을 겪게 하고, 이는 애플리케이션의 품질을 저하시키는 요인이 될 수 있다.

따라서, 본 논문에서는 모바일 애플리케이션을 개발함에 있어서, 플랫폼에 비종속적인 범용 애플리케이션을 개발하기 위해 고려되어야 하는 이식성 척도들과 기존 애플리케이션의 이식성 품질 평가 방법을 제안한다. 이를 위해, 먼저 애플리케이션을 다른 플랫폼으로 이식할 때 고려되어야 하는 모바일 애플리케이션의 특성들을 확인하고, ISO/IEC 9126에서 정의한 소프트웨어 품질 특성 모델을 참조하는 이식성 척도를 도출함으로써 기존 애플리케이션에서 이식성을 평가하기 위한 방법을 설계한다. 이를 통해 모바일 애플리케이션의 타 플랫폼 이식 가능성을 확인함으로써 다른 플랫폼에서의 재개발 필요성과 노력 정도를 확인할 수 있을 것으로 기대된다. 향후 이를 기반으로 모바일 애플리케이션의 타 플랫폼 이식 시, 수정되어야 하는 부분을 확인할 수 있도록 하는 자동화된 이식성 테스트 툴을 연구하고자 한다.

1. 서 론

모바일 애플리케이션은 모바일 환경에서 사용되는 휴대폰이나 휴대용 개인정보단말기(PDA)에서 구현되는 응용프로그램을 의미한다[1]. 최근 모바일 애플리케이션은 스마트폰의 인기에 힘입어 폭발적으로 증가하는 추세를 보이고 있다. 이 같은 현상은 현재 가장 많은 모바일 애플리케이션을 등록하고, 판매하고 있는 애플 앱스토어에 등록된 모바일 애플리케이션의 숫자를 통해 확인할 수 있다.

앱스토어의 통계치를 제공하는 148apps¹⁾에 따르면, 앱스토어에 한달 간 등록된 애플리케이션은 2008년 5월 2건에서 2010년 2월에는 2만여건 이상이 등록되고 있는 폭발적인 증가치를 보이고 있다.

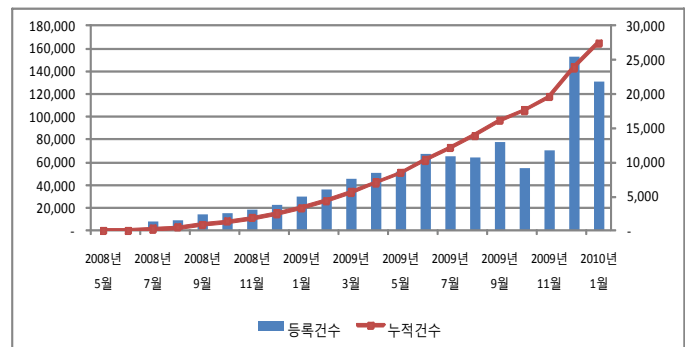


그림 1 애플 앱스토어의 애플리케이션 월별 등록건수 및 누적등록 건수 (기준 : 2010년 1월)

표 1 애플 앱스토어 애플리케이션 월별 추이

	08.5	08.12	09.6	09.12	10.2
등록건수	2	3,865	11,224	25,474	20,579
누적건수	2	14,915	61,839	142,767	185,084

* 본 과제는 정보통신산업진흥원의 SW공학 요소기술 연구개발사업의 결과물임을 밝힙니다.

1) <http://www.148apps.com/10000/>

하지만, 모바일 애플리케이션은 플랫폼에 따라 가지게 되는 상이한 여러 가지 특성때문에 각 플랫폼에 종속적이다[5][2]. 이는 모바일 애플리케이션이 플랫폼에 따라 등록 건수에 차이를 보이는 것에서 확인할 수 있다. MWC 2010에서 Distimo²⁾는 그림 2에서 보이는 것처럼 각 애플리케이션 마켓에 따른 애플리케이션 등록건수를 비교하여 발표하였다. 물론 아이폰의 앱스토어 외에도 애플리케이션 마켓의 성장세가 두드러짐에 따라 다른 애플리케이션 마켓의 애플리케이션 수도 증가하고 있으나 아직도 플랫폼별로 애플리케이션 등록건수는 큰 불균형을 보이고 있다.

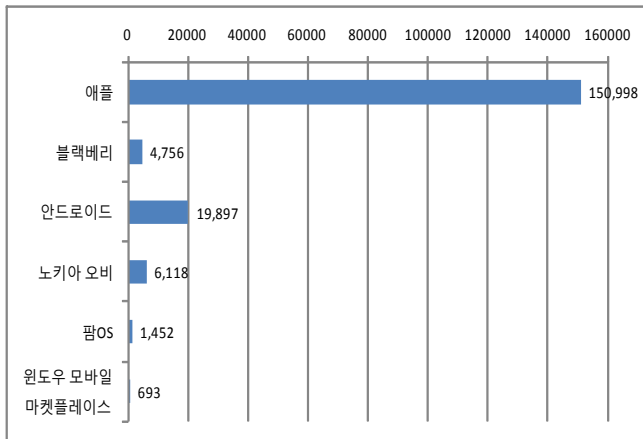


그림 2 애플리케이션 마켓 별 등록건수 비교
(기준 : 2010년 1월)

애플 앱스토어에 등록된 애플리케이션은 아이폰 플랫폼을 기반으로 하고, 안드로이드 마켓에 안드로이드 애플리케이션은 안드로이드 플랫폼을 기반으로 한다. 이때문에, 애플 앱스토어의 애플리케이션은 안드로이드 OS를 사용하는 안드로이드 기반에 스마트폰에서 사용할 수 없다. 이를 가능하게 하기 위해서는 안드로이드 플랫폼에 맞도록 아이폰 플랫폼에 기반한 애플리케이션을 다시 구현해야 한다.

하지만, 모바일 애플리케이션은 3~4개월 안에 개발되어 시장에 출시되어야 하는 짧은 개발주기를 가진다[6]. 이 짧은 개발 주기에 맞춰 여러 플랫폼에서 서로 다른 언어로 개발하는 것은 개발자나 개발사에게 있어 큰 문제가 된다. 예를 들자면, A 플랫폼을 이용하여 개발된 애플리케이션을 B 플랫폼에서 사용하기 위해서는 다시 B 플랫폼에 환경에서 B 플랫폼을 기반으로 재개발해야 한다. 이는 A 플랫폼에서 개발된 소스코드를 재사용하는 것이 아니라, 설계에서부터 플랫폼 환경에 맞는 재설계를 통해 새로 개발하여야 하는 것이다. 이러한 문제는 개발한 애플리케이션에 대해 테스트할 수 있는 시간을 줄이게 되고, 이는 곧 재개발된 모바일 애플리케이션에 품질을 저하시키는 문제를 야기한다. 또한 다양한 플랫폼에서 테스트되어야 하기 때문에 각 플랫폼의 특성을

이해하는 것을 필요로 하게하고, 이는 개발에 들어가는 노력을 가중시킬 수 있다. 이와 같은 현상은 개발자나 개발사로 하여금 각 모바일 플랫폼에 균형적인 애플리케이션 개발을 포기하거나 어렵게 한다.

이에 따라, 본 논문에서는 모바일 애플리케이션을 타 플랫폼에 이식하기 위해 고려해야만 하는 특성들을 확인하고, 이 고려사항을 ISO/IEC 9126 품질 특성의 하나인 이식성과 매핑함으로써 모바일 애플리케이션 이식을 위한 이식성 메트릭을 제안한다. 또한 제안된 메트릭을 정량적으로 평가하기 위해 이식성 세부 테스트 모듈을 제안한다. 이를 통해, 모바일 애플리케이션을 다른 플랫폼에서 구현 할 때, 발생할 수 있는 문제나 필요한 노력을 확인할 수 있으며, 나아가 재개발을 효율적으로 할 수 있도록 애플리케이션 개발 시 제안된 메트릭들을 고려함으로써 좀 더 범용적인 애플리케이션을 개발할 수 있을 것으로 기대된다.

본 논문은 다음과 같이 구성된다. 2장에서는 현존하는 모바일 애플리케이션을 평가하는 방법과 이식성 평가들에 대해 조사하고 모바일 애플리케이션의 품질 향상을 위한 연구들을 보이며, 3장에서는 모바일 애플리케이션을 다른 플랫폼에 이식할 때 고려해야 하는 특성을 확인하고, AHP기법을 응용하여 이 고려사항들을 ISO/IEC 9126에 품질특성 중 하나인 이식성에 부특성들과 매핑한다. 이를 기반으로 4장에서 매핑된 이식성 메트릭들을 정량적으로 평가하기 위한 이식성 세부 테스트 모듈을 제안한다. 그리고 마지막으로 결론과 향후 연구를 소개한다.

2. 관련연구

ISO 8402에 따르면, 품질은 “명시되고 의미하는 니즈를 만족시키는 그것의 능력에 관련된 구성요소의 전체 특성” 이라고 정의되며, ISO/IEC 9126은 그림 3과 같이 소프트웨어 품질 특성의 6개 주특성들과 27개의 부특성으로 구성되는 품질모델을 정의한다. 이 품질 모델은 일반적이기 때문에 어떤 소프트웨어 제품에 특정한 용도에 따라 맞춤화하여 적용할 수 있다[7][9].

가능성	신뢰성	사용성	효율성	유지보수성	이식성
적합성	성숙성	이해가능성	시간반응성	분석성	적용성
정확성	결함허용성	학습성	자원효율성	변경성	설치가능성
상호운용성	회복성	운영성	준수성	안정성	대체성
보안성	준수성	선호도		시험가능성	공존성
준수성		준수성		준수성	준수성

그림 3 ISO/IEC 9126-1에 정의된 주특성과 부특성

이런 특성들을 통해 모바일 애플리케이션을 테스트하는 방법은 기존에 많은 연구가 있었다. Sakura She[8]는 J2ME(Java 2 Platform, Micro Edition) 애플리케이션들을 자동으로 테스트하기 위한 툴인 Hermes를 제안했다. Hermes는 자동화된 테스트 실행과 리포팅, 다방면 블럭

2) Distmo : 네덜란드의 앱스토어 분석업체,
http://www.readwriteweb.com/archives/the_truth_about_mobile_application_stores.php

박스 테스트를 지원하며, 독립된 애플리케이션이고, 확장 가능성을 지원하는 오픈 인터페이스를 포함한다. Hermes는 외적으로 식별될 수 있는 행위를 기반으로 하여, 다음 표 2와 같은 요인들을 테스트한다.

표 2 Hermes에서 수행하는 다방면 테스트

Facet type	Test concern
Aesthetics	Layout, Color, Font
Enviroment	Response time, Memory usage, Network availability and selection, Power consumption
Functional	Content, Navigability

특히, 최근 스마트폰의 애플리케이션을 테스트하기 위한 연구로써, Antti[10]는 스마트폰 애플리케이션을 위한 GUI 테스트를 자동화하기 위해, 상태 머신 기법을 이용한 테스트 방법을 제안하고, 스마트폰의 제한된 자원이라는 제약조건에 따라 단순하게 GUI를 구성하는 방법을 기술했다. 또 여기에 소프트웨어 프로덕트 라인의 개념을 도입하여 테스트 산출물과 결과들을 재사용할 수 있는 방법도 제안했다. 이런 제안들을 검증하기 위해, 기존에 시장 출시된 스마트폰 S60에 내장된 애플리케이션과 제안들의 상호작용을 테스트하였고, 이 테스트에서 기존에 발견되지 못한 몇몇 결함을 찾아낼 수 있었다.

하지만, 이런 모바일 애플리케이션의 테스트를 위한 연구는 최근의 다양한 플랫폼에서의 애플리케이션 이식성을 확인하는데에는 적합하지 않다.

또한 ISO/IEC 9126 특성을 직접적으로 참조하는 S/W의 이식성 시험 방법에 대한 연구로써, 양해술[3]은 각종 디지털 콘텐츠를 불법복제로부터 보호하는 디지털 저작권 관리 S/W의 이식성을 평가하기 위한 시험 메트릭을 제안했다. 이를 위해 ISO/IEC 9126 이식성의 부특성인 적응성에 권리 표현 지원, 설치가능성에 설치정보 제공, 공존성에 공존 가능 정보 제공, 준수성에 이식 표준 준수 정보 제공을 메트릭으로 정의하고 평가 방법을 제안했다. 이 연구를 통해 기존에 ISO/IEC 9126이나 ISO/IEC 12119기반의 평가에 비해 디지털 저작권 관리 S/W의 이식성에 초점을 맞춘 핵심적이고 최적화된 평가를 수행할 수 있다. 표 3은 이 연구에서 제안한 이식성 중 하나인 적응성에 권리 표현 지원 메트릭의 시험 모듈이다.

표 3 권리 표현 지원 메트릭의 시험 모듈

적용성	메트릭명	권리 표현 지원	Right Expression에 관해 콘텐츠의 형태에 따른 다양한 형태의 권한을 제공하는가?
	측정항목	A	콘텐츠 형태에 다른 권리 표현 형태의 수 문서인 경우 View, Print, 동영상의 경우 Play, S/W의 경우 Execute 등
	B	S/W에서 제공하는 권리 표현 형태의 수	
	계산식	권리 표현 지원 = B/A	
	결과영역	$0 \leq \text{권리 표현 지원} \leq 1$	
	적용대상	공통	

이런 모바일 테스트 연구를 위해 우선 확인되어야 하는 테스트 요구사항들에 대한 연구도 활발하다. Valeria[11]은 테스트 요구사항을 2가지 카테고리로 분류하여 전문가와 문헌조사를 비교함으로써 모바일 테스트의 요구사항을 도출하였다. 2가지 카테고리 중 테스트 프로세스를 위한 테스트 요구사항은 다음과 같다.

- ① 개발 프로세스 모델은 테스트 프로세스에 초점을 맞춰야만 한다.
- ② 모바일 애플리케이션은 에뮬레이터와 모바일 디바이스에서 모두 테스트되어야만 한다.
- ③ 테스트 리포트는 애플리케이션 이름과 버전, 에뮬레이터의 버전, 모바일 디바이스와 테스트 환경에 대해 명시해야만 한다.
- ④ 확인되어진 각 에러에 대해, 에러 리포트는 그것의 서술, 발생 빈도, 애플리케이션에서의 위치와 그것을 반복하기 위한 단계별 가이드를 알려야만 한다.
- ⑤ 모바일 애플리케이션은 모바일 디바이스에서 이미 설치된 애플리케이션에 해를 끼쳐서는 안 된다.
- ⑥ 모바일 애플리케이션은 모바일 제약사항에 따라 테스트되어야 한다.
- ⑦ 테스트는 연구실에서 테스트될 수 있는 것과 실제 모바일 환경(필드 테스트)에서 테스트하는 것 중 어떤 것이 올바른 것인지 알아야만 한다.
- ⑧ 사용성 테스트는 모바일 애플리케이션의 개발 사이클동안 포함되어야만 한다.

이와 같이 모바일 애플리케이션은 최근 증가추세에 발맞추어 많은 연구가 이루어지고 있다. 하지만, 최근에 모바일 플랫폼의 다양성을 고려한 모바일 애플리케이션 테스트에 대한 연구는 부족하다.

3. 모바일 애플리케이션을 위한 이식성 품질 평가

3.1 모바일 애플리케이션 이식에 고려사항

모바일 애플리케이션을 위한 이식성 품질 평가 모델을 개발하기에 앞서, 먼저 모바일 애플리케이션을 다른 모바일 플랫폼에 이식하기 위해 고려해야하는 특징을 확인해야 한다. 이에 이절에서는 다음과 같은 고려사항을 도출하였다.

① 플랫폼별 개발언어 차이

모바일 애플리케이션들의 개발언어는 플랫폼에 종속적이다. 이런 특성 때문에 플랫폼에 따라 개발언어가 상이할 수밖에 없다. 이는 다양한 플랫폼에서 기존 애플리케이션의 모듈을 재사용할 수 없게 만들며, 재개발을 어렵게 하는 요인이 된다. 예를 들어, 애플리케이션을 가장 많이 보유하고 있는 애플의 아이폰 애플리케이션은 Objective-C로 개발된다. 하지만 구글의 안드로이드 플랫폼은 Java로, 윈도우 모바일 플랫폼은 .NET관련 언어로 개발된다. 이러한 차이는 다른 플랫폼에 기존 모바일 애플리케이션을 재사용하는데 기본적인 어려움이 될 수밖에 없다. 이 때문에, 각 언어를 타언어로 변화하는 것에 대한 이식성을 확인하는 것이 필요하다.

② 플랫폼에서 지원하는 다양한 API의 차이

모바일 애플리케이션을 다른 플랫폼에 이식하기 할 때 가장 어려움을 겪는 부분은 API의 차이이다. A플랫폼의 애플리케이션 개발을 위해 사용되는 API는 B플랫폼에서 다른 API로 대체되어야만 하거나 대체할 수 있는 API가 없다면 다시 구현되어야 한다. 이 때문에 API의 차이를 확인하는 것은 이식을 얼마나 쉽게 할 수 있는가에 대해 영향을 미칠 수밖에 없다. 예를 들어, 애플 아이폰의 OS는 화상통화를 위한 전면 카메라를 지원하지 않아, 이에 관련된 API가 존재하지 않는다. 하지만 삼성 옴니아2에서 사용 중인 윈도우 모바일 6.5는 전면 화상통화용 API를 지원해야 한다. 이때 옴니아2의 화상통화용 API를 사용하는 애플리케이션을 아이폰에 이식하려 한다면, 대체할 수 없는 API가 없기 때문에, 이를 예외처리하거나 다른 방법으로 우회하여 사용할 수 있는 방법을 재개발해야 한다. 이런 문제들 때문에, API의 처리 방법에 대한 이식성 문제를 확인할 필요가 있다.

③ 플랫폼간의 GUI구성의 차이

모바일 디바이스의 디스플레이는 크기가 일반 컴퓨터에 비해 작기 때문에 가시성 및 가독성이 떨어지고 표현할 수 있는 정보의 양이 제한되는 문제를 가진다. 하지만 사용자는 GUI를 통해 애플리케이션과 커뮤니케이션하기 때문에 사용자에게 제일 빠르게 영향을 미치는 부분이라 할 수 있다. 이와 같은 GUI의 고려사항들은 사용자가 직접 경험하는 품질 척도이기 때문에 중요한 테스트 요소가 된다. 게다가, GUI는 각 플랫폼별로 구성하는 방법이 다르기 때문에 이식 시 반드시 확인해야 하는 고려사항이다. 또한 같은 플랫폼 안에서도 제조사별로 GUI를 다르게 구성할 수 있기 때문에, GUI구성은 재개발하는데 반드시 확인되어야만 한다. 이 때문에, GUI를 구성하는 것에 대한 이식성을 확인하는 것은 필요하다.

④ 사용자 인터렉션에 대한 처리

모바일 디바이스에 종속적인 하드웨어 버튼은 모바일 애플리케이션에도 영향을 미친다. 애플리케이션이 실행되다가 종료버튼을 눌러 애플리케이션을 종료시킬 수도 있고, 키패드를 가진 모바일기기에서는 키패드를 이용해 애플리케이션을 조작하지만, 터치폰에서는 이러한 키패드를 가지지 못했기 때문에 디스플레이에 가상의 키패드를 활성화하기도 한다. 또한 어떤 모바일 기기에서는 멀티터치를 지원하지만, 싱글터치만을 지원하거나 터치패드의 지원 없이 키패드만을 지원할 수도 있다. 이런 다양한 인터렉션은 모바일 디바이스에 종속적이지만, 모바일 애플리케이션에 큰 영향을 미친다. 이 때문에 애플리케이션을 이식하는데 있어서 모바일 애플리케이션이 각각의 인터렉션을 어떻게 공유하는지에 대한 지원방법도 고려할 필요가 있다.

⑤ 다른 애플리케이션과의 연계 처리

최근 애플리케이션들은 독립적으로 실행되기 보다는 다른 애플리케이션과의 연계를 통해 동작되기도 한다. 이때 연계된 다른 애플리케이션과의 상관관계를 파악하여 이에 해당하는 상관관계를 이식되는 애플리케이션에

서도 만들어 줄 필요가 있다. 예를 들어, A플랫폼에서 a1이라는 애플리케이션이 a2라는 애플리케이션을 호출하여 데이터를 수집한다면, B플랫폼에 a1을 이식하여 b1을 개발할 때 a2를 대체할 수 있는 b2 애플리케이션이 필요하다. 이 조건을 충족시키지 못한다면 b1 애플리케이션은 데이터를 수집하지 못하는 치명적인 문제를 가지게 된다. 이 때문에 애플리케이션을 이식하는데 있어서 다른 애플리케이션과의 연계성도 중요한 이식성의 이슈가 된다.

3.2 모바일 애플리케이션의 이식을 위한 고려사항과 ISO/IEC 9126 이식성의 부특성 매핑

이식성은 한 환경에서 다른 환경으로 전이될 수 있는 소프트웨어의 능력을 의미하며[4], 이식성은 소프트웨어가 특정 환경에서 다른 환경으로 적용할 수 있는 소프트웨어의 능력을 의미하는 **적용성**, 명세된 환경에서 설치될 수 있는 소프트웨어의 능력을 의미하는 **설치 가능성**, 동일한 환경에서 동일한 목적으로 다른 지정된 소프트웨어를 대신하여 사용될 수 있는 소프트웨어의 능력을 의미하는 **대체성**, 공통 자원을 공유하는 공동 환경에서 다른 독립적인 소프트웨어와 공존할 수 있는 소프트웨어의 능력을 의미하는 **공존성**, 마지막으로 이식성과 관련된 표준 및 관례를 준수하는 소프트웨어의 능력을 의미하는 **준수성**의 부특성으로 구성된다[12].

위 3.1절에서 고려한 모바일 애플리케이션 이식을 위한 고려사항을 위해 ISO/IEC 9126의 이식성 부특성에 매핑하기 위해 다기준(Multiple-criteria) 의사결정(Decision-making) 문제에 대안을 선택하는 AHP(Analytic Hierarchy Process)에서의 상대비중을 응용하여 매핑하려 한다. 이를 위해 먼저 다음 그림 4와 같이 이식성 특성과 고려사항을 매핑하기 위한 계층적 구조를 정의하였다.



그림 4 매핑을 위한 계층적 구조

이 계층적 구조를 통해 ISO/IEC 9126의 이식성 부특성들을 기준으로 모바일 애플리케이션을 이식하기 위한 고려사항들을 쌍대비교 함으로써, 상대적 비중이 높은 고려사항을 부특성과 매칭시켜 모바일 애플리케이션 이식을 위한 이식성 메트릭으로 구성하고자 한다. 이를 위해 모바일 애플리케이션 개발자를 대상으로 인터뷰를 통해 쌍대비교를 수행하였다. 표 4는 적용성을 기준으로 고려사항 ① ~ ⑤를 쌍대비교한 결과를 정리하여 쌍대비교표를 작성한 예를 보인다.

표 4 적응성을 기준으로 한 쌍대비교표

고려사항	①	②	③	④	⑤
①	1	1/5	1/5	1/3	1/2
②	5	1	1	2	3
③	5	1	1	2	3
④	3	1/2	1/2	1	2
⑤	2	1/3	1/3	1/2	1

이런 쌍대비교를 다섯 가지 부특성을 기준으로 각각 수행하고, 이를 통해 각 고려사항에 가중치를 산정하였다. 그리고 이를 표 5와 같이 정리하였다. 또한 AHP기법에서 인터뷰 대상자의 답변에 불일치성에 대한 검증을 위해 수행하는 일치율(Consistency Ratio) 평가를 식 1을 이용하여 확인하였다.

$$\text{Consistency Index} = \frac{\max - n}{-1}$$

$$\text{Random Consistency Index} = 1.12 (n=5)$$

$$\text{Consistency Ratio} = \frac{CI}{.CI} \quad (\text{식 } 1)$$

이때 일치율(C.R)이 0.1 이내일 경우는 일반적으로 받아들여지지만 0.2이상일 경우는 자료입력을 다시 할 것을 권하고 있다[13].

표 5 부특성별 고려사항의 가중치와 일치율

	적응성	설치가능성	대체성	공존성	준수성
①	0.061	0.434	0.196	0.090	0.200
②	0.326	0.193	0.087	0.074	0.200
③	0.326	0.191	0.127	0.127	0.200
④	0.180	0.096	0.118	0.477	0.200
⑤	0.107	0.086	0.472	0.231	0.200
일치율	0.0034	0.0100	0.0718	0.0469	0

산정된 가중치에 따라, 고려사항별로 가장 높은 가중치를 보이는 5가지 점수를 기준으로 ISO/IEC 9126 이식성 부특성과 모바일 애플리케이션 이식 시 고려할 사항을 매핑하도록 한다. 매핑된 결과는 그림 5와 같다.



그림 5 이식성 부특성과 고려사항 매핑

이 중 준수성에 대해 고려사항이 매핑되지 못한 것은 쌍대비교시 준수성을 기준으로 한 조사에서 모든 고려사항이 동일하게 준수성을 같은 중요도로 판단하여야 한다고 조사되었기 때문이다.

4. 모바일 애플리케이션의 이식성 테스트 매트릭

본 장에서는 모바일 애플리케이션의 이식성 테스트를 위해 3.2절에서 매핑된 모바일 애플리케이션 이식을 위한 매트릭을 기준으로, 모바일 애플리케이션의 이식성을 정량적으로 평가할 수 있는 모바일 애플리케이션 이식성 세부 테스트 모듈을 다음 표 6과 같이 설계하였다.

표 6 모바일 애플리케이션의 이식성 테스트 모듈

메트릭명	API 이식 편리성	
	애플리케이션 구현에 사용된 API와 같은 기능을 하는 API의 존재 유무 확인을 통해 API를 쉽게 구현 할 수 있는 능력	
측정 항목	A	대체 가능한 API 존재 여부 - 각 API별로 타겟 플랫폼에 이식가능한 API가 존재하는지 확인
	B	사용되는 API 총 갯수
계산식	- API 이식 편리성 = $\frac{A_i}{\sum_{i=1}^B B}$ A _i = Y(1) or N(0)	
결과영역	0 ≤ API 이식 편리성 ≤ 1	
메트릭명	GUI 구조 적응률	GUI가 이식되는 플랫폼에 적용할 수 있는 능력
측정 항목	A	레이아웃 표현 가능 정도 - 레이아웃 크기 비교를 통해 표현 가능 정도를 확인
	B	사용자 컨트롤 사용 가능 정도 - 사용자 컨트롤(입력박스, 버튼등)의 사용성 테스트를 통해 사용 가능 정도 확인
	C	사용된 콘텐츠 표현 정도 - 텍스트, 이미지, 소리등의 콘텐츠의 원본 애플리케이션과의 비교를 통해 표현 가능 정도를 확인
계산식	- GUI 구조 적응률 = (A + B + C) / 3 - 원본 레이아웃 크기 >= 대상 레이아웃 크기 A = 1-(표현 불가능 영역 넓이 비율) - 원본 레이아웃 크기 >= 대상 레이아웃 크기 A = 1-(낭비되는 영역 넓이 비율) - B = 사용가능한사용자컨트롤갯수 / 용자컨트롤갯수 - C = 표현가능한콘텐츠갯수 / 표현될콘텐츠총갯수	
결과영역	0 ≤ GUI 구조 적응률 ≤ 1	
메트릭명	언어별 설치가능성	개발언어의 차이에 따른 이식 후 동일하게 설치될 수 있는 능력
측정 항목	A	각 클래스의 블랙박스 테스트 성공 여부 - 원본 소스코드를 클래스 단위로 타겟 플랫폼의 개발언어로 재개발 후 블랙박스 테스트 수행
	B	전체 클래스의 갯수 - 개발언어에 따라 이식성 확인을 위해 클래스 단위로 분리하여 테스트
계산식	- 언어별 설치가능성 = $\sum_{i=1}^B \frac{A_i}{B}$ A _i = Y(1) or N(0)	
결과영역	0 ≤ 언어별 설치가능성 ≤ 1	

대 체 성	메트릭명	타 App과의 연계성	원본 애플리케이션과 타 애플리케이션과의 연계를 타겟 플랫폼에서 정확하게 대체하는 능력
	측정항목	A	각 연계점에 대한 대응 여부 - 다른 애플리케이션과 연결되는 부분 확인 후 이에 대응하는 애플리케이션 확인
		B	원본의 연계app과 타겟의 연계app의 기능 동일 여부 - 연계 app간의 블랙박스 테스트를 통하여 애플리케이션 기능 동일성 확인
		C	전체 타 애플리케이션과의 연계점
계산식	$- \text{타 App과의 연계성} = \frac{A_i + (\sum_{i=1}^C B_i)}{C} / 2$ $A_i = Y(1) \text{ or } N(0)$ $B_i = Y(1) \text{ or } N(0)$		
결과영역	0 ≤ 타 App과의 연계성 ≤ 1		
공 존 성	메트릭명	인터랙션 공존성	사용자 인터랙션과 애플리케이션이 충돌없이 동시에 공존할 수 있는 능력
	측정항목	A	인터랙션 공존여부 - 애플리케이션 실행 중 인터랙션 발생 후 애플리케이션에 이상 유무 확인 (항목 예) - 확인버튼, 1번키, A키, 네비게이션키등
		B	총 인터랙션 항목 수 - 모바일 애플리케이션에서 발생가능한 인터랙션의 총 개수를 확인
	계산식	- 인터랙션 공존성 = $\frac{Y \text{ 측정되는 } A \text{ 의 총 개수}}{B}$	
결과영역	0 ≤ 인터랙션 공존성 ≤ 1		

이 이식성 테스트 세부 모듈은 각 고려사항을 통해 이식성을 테스트할 수 있는 방법을 메트릭명, 측정항목, 계산식, 결과영역으로 나누어 정리하였다.

5. 결론 및 향후 연구

본 논문에서는 다양한 모바일 플랫폼에 모바일 애플리케이션을 이식하기 위해 고려되어야 하는 요인을 확인하고 이를 ISO/IEC 9126의 이식성 부특성에 기반하여 매핑함으로써 API 이식편리성, GUI 구조 적응률, 언어별 설치가능성, 타 App과의 연계성, 인터랙션 공존성이라는 이식성 메트릭을 제안하였다. 그리고 이에 따른 이식성 테스트 모듈을 제안하였다. 이 모듈을 통해 기존 모바일 애플리케이션의 이식성을 평가할 수 있으며, 모바일 애플리케이션의 다른 플랫폼으로의 이식이 좀 더 효율적으로 수행될 수 있을 것으로 기대된다. 또한 모바일 애플리케이션 설계 시부터 이 메트릭들을 고려한다면, 이식성이 뛰어난 범용적인 모바일 애플리케이션을 개발할 수 있을 것으로도 기대된다.

향후 제안된 이식성 테스트 세부 모듈을 기반으로 모바일 애플리케이션의 이식성 테스트를 수행함으로써 이식성 정도를 평가하고, 이 세부 모듈을 고려한 애플리케이션을 구현함으로써 테스트 모듈을 확장할 수 있는 방법과 적용하는 방법에 대한 연구를 진행할 필요가 있다. 또한 이식성을 자동으로 테스트하여 이식 시 수정되어야 하는 부분에 대한 확인을 도와주는 자동화 툴의 연구도 필요할 것으로 보인다.

6. 참고문헌

- [1] 박재석, “모바일 애플리케이션(앱스토어) 수용 방안에 대한 연구”, 한국경영정보학회 추계학술대회, 2009.
- [2] 노명기, “모바일 애플리케이션을 테스트하기 위한 아키텍처 모델”, 한국정보과학회 가을 학술발표논문집, Vol. 31, No.2, 2004.
- [3] 양해술, “디지털 저작권 관리 S/W의 이식성 시험 방법”, 한국콘텐츠학회논문지, Vol. 9, No. 4, 2009.
- [4] 한국정보통신기술협회, 소프트웨어 품질평가 용어, TTAS.KO-11.0063, 2006.
- [5] Sangwhan Cha, “Toward a unified framework for mobile applications”, Communication Networks and Services Research Conference, 2009.
- [6] Choi Yoonseok, “Application Framework for Multi Platform Mobile Application Software Development”, 11th International Conference for Advanced Communication Technology, 2009.
- [7] Ho-Won Jung, “Measuring software product quality: a survey of ISO/IEC 9126”, Software, IEEE, vol 21, Issue 5, 2004.
- [8] Sakura She, “Hermes: A Tool for Testing Mobile Device Application”, Australian Software Engineering Conference, 2009.
- [9] Roger S. Pressman, *Software engineering - A Practitioner's Approach*, Mcgraw Hill International Edition, p397-415, 2010.
- [10] Antti Jaaskelainen, “Automatic GUI Test Generation for Smartphone Application - an Evaluation”, International Conference on Software Engineering - Companion Volume, p112-122, 2009.
- [11] Valeria Lelli Leitao Dantas, “Testing Requirements for Mobile Applications”, 24th International Symposium on Computer and Information Sciences, p555-560, 2009.
- [12] ISO, ISO/IEC 9126-1 : Information Technology - Software Quality Characteristics and Metrics - Part 1: Quality characteristics and subcharacteristics, 1997.
- [13] Saaty. T. L., “The Analytic Hierarchy Process”, McGraw-Hill, New York, 1980.