

디지털 앱북 상호작용 라이브러리 설계 및 구현

이강운 백아람 최해철

국립한밭대학교 정보통신전문대학원 멀티미디어공학과

kkawoons@naver.com, aram98123@naver.com, choihc@hanbat.ac.kr

Design and Implementation of the Digital App-Book Interaction Library

Kang-Woon Lee A-Ram Beak Haechul Choi

Multimedia Engineering, Graduate School of Information and Communication,

Hanbat National University

요약

최근 스마트 폰 가입자 수 증가와 함께 모바일 기기의 보급이 대중화 되었다. 모바일 기기 대중화로 다양한 형태의 멀티미디어 콘텐츠 소비가 많아지면서 새로운 콘텐츠를 요구하는 사용자들 늘어나고 있다. 다양한 소비 계층에서 콘텐츠의 수요가 늘어나고 있지만, 현재 시장에서는 공급량이 수요에 미치지 못하고 있다. 모바일 기기중 하나인 디지털 앱북 콘텐츠 또한 시장에서 더 많은 콘텐츠가 필요하다. 본 논문에서는 디지털 앱북의 콘텐츠 증가를 위해 디지털 앱북 제작에 있어 코드 재사용성 증가로 인한 체계적인 개발과 유지보수의 용이함을 높이기 위해 사용자와의 다양한 상호작용 기능들을 라이브러리로 구현하고 검증하였다. 라이브러리는 사용자 동작부와 기기 동작부로 구성하고 각 동작부의 명령들을 조합하여 상호작용 메소드 형태로 존재하도록 설계하였다. 구현된 라이브러리는 코드 사용량을 크게 줄여 개발 시간 단축에 영향을 주어 디지털 앱북 콘텐츠 제작에 있어 효율을 높여준다.

1. 서론

국내 스마트 폰 가입자 수 증가에 따른 스마트 폰을 포함한 모바일 기기의 확산으로 대중화가 되었다. 모바일 기기의 대중화로 인해 소비자들은 다양한 형태의 멀티미디어 콘텐츠를 쉽게 접할 수 있게 되었고 자연스럽게 콘텐츠 소비가 증가하고 있다. 이로 인해 멀티미디어 콘텐츠는 특정한 소비 계층만 이용하는 콘텐츠가 아닌 누구나 이용할 수 있고, 장소와 시간에 상관없이 이용할 수 있는 문화콘텐츠산업 성장에 있어서 중요한 역할을 맡고 있으며 관련 산업들도 성장세를 보이고 있다[1].

전자책은 멀티미디어 콘텐츠 기기의 대표적인 단말기이다. 아마존은 풍부한 콘텐츠를 확보하면서 전용단말기인 킨들을 출시하였고 전자책 시장의 대표적인 기업이 되었다. 아마존은 킨들을 앞세워 전자책 시장을 주도하며 빠르게 성장하고 있다[2]. 전자책 시장의 성장세와 더불어 스마트폰이나 스마트 패드와 같은 스마트 단말기에서 제공하는 콘텐츠인 디지털 앱북의 등장으로 다양한 단말기를 통한 콘텐츠의 수요가 급속도로 증가하고 있지만, 콘텐츠 시장의 콘텐츠 공급량이 수요를 따라가지 못하고 있어 더 많은 콘텐츠를 필요로 하고 있다. 따라서 본 논문에서는 디지털 앱북에서 다양한 상호작용 기능들을 라이브러리 설계 및 구현을 하고 디지털 앱북 개발에 적용하여 코드 라인 수 비교를 통해 검증하고자 한다.

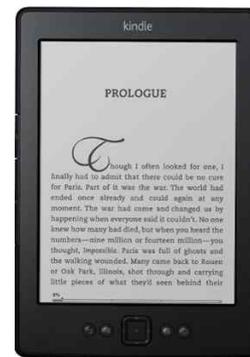


그림 1. 전자책 단말기 아마존의 킨들

2. 디지털 앱북 상호작용 라이브러리 설계

디지털 앱북 상호작용 라이브러리는 크게 사용자와 단말기의 동작부로 나누었다. 기기 동작부는 스마트 단말기를 이용하여 디지털 앱북을 조작할 때 기기가 사용자 입력에 대한 반응하는 기능들로 이루어져 있다. 사용자 동작부는 사용자가 동작을 입력하여 스마트 단말기에서 동작을 인식하여 판별하는 기능들로 구성하였다. 사용자 동작부와 기기 동작부 각각의 기능을 조합하여 상호작용 기능을 만들어 사용할 수 있도록 라이브러리화 하였다.

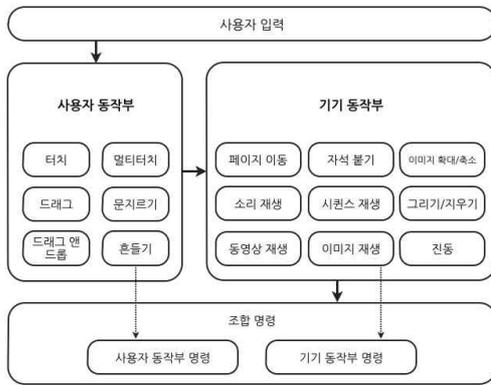


그림 1. 디지털 앱북 상호작용 라이브러리의 수행 구조

사용자 동작부는 6가지 명령, 기기 동작부는 9가지 명령으로 이루어져 있다. 이를 조합하게 되면 54가지의 상호작용 명령어를 만들어 내고 디지털 앱북 개발자가 손쉽게 이용할 수 있다. 또한 현재 각 동작부에 포함되어 있는 명령어들뿐만 아니라 다른 명령들이 추가되면 더 많은 상호작용 조합이 가능해진다. 추후에 각각의 동작부에 새로운 명령은 개수에 상관없이 자유롭게 추가할 수 있도록 하였다. 이처럼 다양한 사용자 동작과 기기 동작을 결합한 명령어들을 라이브러리화 하여 디지털 앱북 제작에 있어 다양한 상호작용 적용이 쉽도록 설계하였다.

3. 구현

라이브러리 구현 환경은 다음과 같다. 사용 언어는 자바를 사용하였고, 개발툴은 Eclipse 4.2.1과 ADT v22.3.0 플러그인을 사용하였으며, 안드로이드 SDK 4.1.2 버전을 사용하였다.

상호작용 라이브러리 구성은 기기 동작 메소드들과 이를 포함하고 있는 사용자 동작 메소드들로 구성하였다. 각각 동작부의 명령들은 하나의 메소드로 이루어져 있다. 기기 동작 메소드는 그 자체만으로도 작동이 가능하지만 사용자 동작 메소드는 사용자의 동작을 판별만 할 뿐 별도의 동작을 하지 않는다. 그렇기 때문에 사용자 동작 메소드에서 사용자의 동작을 판별 후 기기 동작 메소드를 호출하도록 하였다. 이렇게 결합된 메소드는 하나의 상호작용 기능을 하는 메소드가 된다.

```

a 상호작용 라이브러리 적용
touch_link(iv, "Aclass");
a 상호작용 라이브러리 호출
iv.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
Intent intent = new Intent(this, Aclass.class);
startActivity(intent);
}
});
}
    
```

그림 2. 상호작용 라이브러리 적용과 미적용 인터랙션 구현 예

표 1. 상호작용 라이브러리 적용과 미적용 코드 라인 수 비교 예

| 인터랙션 | 상호작용 라이브러리 적용 | 상호작용 라이브러리 미적용 |
|-------------------|---------------|----------------|
| 터치 + 페이지 이동 | 1줄 | 8줄 |
| 드래그&드롭 + 자석 붙기 | 1줄 | 23줄 |
| 드래그 + 시퀀스 재생 | 1줄 | 35줄 |
| 멀티 터치 + 이미지 확대/축소 | 1줄 | 48줄 |

상호작용 라이브러리 메소드 구현 코드의 예로 그림 2에서 터치와 페이지 이동 인터랙션을 구현한 코드를 확인 할 수 있다. 상호작용 라이브러리를 적용한 코드는 한 줄로 표현하였다. 그러나 상호작용 라이브러리를 이용하지 않고 터치 + 페이지 이동 인터랙션을 구현 할 경우 대략 8줄이 필요하다. 이러한 인터랙션 코드 사용의 경우 7줄이 감소하고 반복적으로 사용되므로 그 효율성은 중복 구현에 많이 질수록 늘어나게 된다.

표 1은 일부 인터랙션의 상호작용 라이브러리 적용 유무에 따른 라인 수를 보여준다. 인터랙션 하나를 작성 할 때 드는 코드 라인 수는 적게는 8줄 많게는 48줄이 사용된다. 그러나 상호 작용 라이브러리를 적용을 하면 코드는 단 한 줄이다. 프로젝트의 제작 개수와 인터랙션이 사용된 횟수가 많을수록 전체 코드의 감소량은 증가한다.

4. 결론

디지털 앱북은 기존 전자책에 비해 많은 기능을 추가할 수 있다. 하지만 추가하는 기능이 많을 수록 개발시간은 늘어난다. 이는 디지털 앱북 콘텐츠의 생산성을 저해하여 시장 성장의 방해요소로 작용 할 수 있다. 이러한 문제점들을 해결하기 위해 본 논문에서는 디지털 앱북의 콘텐츠 증가를 위해 라이브러리는 사용자 동작부와 기기 동작부로 구성하고 각 동작부의 명령들을 조합하여 상호작용 메소드 형태로 존재하도록 설계하였다. 또한 디지털 앱북 제작에 있어 코드 재사용성 증가로 인한 체계적인 개발과 유지보수의 용이함을 늘려 개발 효율을 높였다. 이러한 상호작용 기능들을 라이브러리로 구현하고 검증하였다.

향후 iOS나 타이젠과 같은 다양한 플랫폼에 대한 대응과 새로운 상호작용들 연구하여 다양한 상호작용을 지원 할 수 있도록 할 예정이다.

참고 문헌

[1] 유진룡, 콘텐츠산업 통계조사, 문화체육관광부, 2013.
 [2] 송양희, 강영식, "전자출판물 산업경쟁력 강화를 위한 표준화 추진 방향", TTA 저널, 제 130권 pp.14-19, 2010.