

HTML5 기반 스마트 TV 플랫폼 표준 앱 개발도구 확장¹

김정현, 홍성욱, 김호년, 이동훈, 박동영
한국정보통신기술협회

{jllh, swhong, hykimfnd, dhlee, dypark}@tta.or.kr

Expansion of Standard Conformance Software Development Kit
for Smart TV Platform based on HTML5

Jung-Hyun Kim, Ho-Youn Kim, Dong-Hoon Lee, Dong-Young Park
Telecommunication Technology Association

요 약

한국정보통신기술협회(이하,TTA)에서는 “HTML5 기반 스마트 TV 플랫폼(TTAK.KO-07.0111/R2)” 표준(이하, 스마트 TV 2.0 표준)을 2016 년 6 월에 개정하였으며, 본 논문에서는 스마트 TV 2.0 표준을 준수하여 개발되는 스마트 TV 용 앱 개발도구(SDK, Software Development Kit)의 두 가지 확장 내용을 다룬다. 첫 번째 확장 내용은 스마트 TV 2.0 표준이 개정됨에 따라 추가되는 기능인 고급입력 API, 녹화 및 다운로드 API 인터페이스를 확인하고 검증할 수 있는 검증기능 구현에 대한 내용이며, 두 번째 확장 내용은 표준기반 TV 애플레이터를 스마트 TV 앱 개발도구에서 사용 가능하게 하는 인터페이스 구현에 대한 내용이다. 본 논문에서는 표준기반 서비스를 만드는 개발자에 대한 지원과 표준기반 TV 앱 생태계의 활성화를 위한, 스마트 TV 플랫폼 표준 앱 개발도구 확장기능의 설계 및 구현에 대하여 소개한다.

1. 서론

“HTML5 기반 스마트 TV 플랫폼” [1][2][3] 표준은 제조사와 플랫폼 사업자 중심으로 파편화된 스마트 TV 앱 생태계의 한계를 극복하고 표준 기반의 공통 플랫폼 기술을 통해 앱 생태계를 활성화 및 촉진시키기 위해 제정된 개방형 스마트 TV 플랫폼 기술 규격이다. 이 표준은 2014 년 스마트 TV 플랫폼에 대한 기본 기능과 멀티스크린, 적응형 스트리밍 등의 부가 기능을 지원하는 스마트 TV 1.0 표준(TTAK.KO-07.0111/R1)으로 정의되었으며, 2016 년에는 원격 앱 제어, 모션·음성 등 고급 입력, 콘텐츠 동기화, 결제 등의 고급 기능 지원하는 스마트 TV 2.0 표준(TTAK.KO-07.0111/R2)으로 개정되었다.

표준 기반 앱 생태계 활성화를 위한 활동의 일환으로, TTA 에서는 앱 개발자들이 표준기반 앱을 보다 편리하게 개발할 수 있도록 표준기반 스마트 TV 앱 개발도구 [4]를 개발하였다. 개발도구는 표준기반 앱 개발의 전체 라이프사이클에 필요한 프로젝트 관리, 앱 오류검사, 애플리케이션, 패키징 및 앱 서버 전송 기능을 자동화하였다. 표준에 대한 이해도가 낮은 개발자들이 표준기반 앱을 개발할 수 있도록 코드 자동 완성 기능과 표준 적합성 검사 기능 및 샘플 앱 등을 제공함으로써, 개발자들이 앱 자체의 기능에 대한 설계와 분석에만 집중할 수 있도록 하여 보다 완성도 높은 앱을 개발할 수 있도록 한다. 본 논문에서는 위 스마트 TV 앱

개발도구의 두 가지 확장기능에 대하여 소개한다.

첫 번째 확장기능은 스마트 TV 앱 개발도구에 포함된 앱 검증기능 [5]에 관한 내용이다. 앱 검증기능은 개발자들의 개발된 코드가 표준에서 정의한 속성 및 기능을 제대로 사용하고 있는지 검증해 주는 역할을 한다. 스마트 TV 2.0 표준이 개정되면서 고급입력 API, 녹화 및 다운로드 API 등 스마트 TV 플랫폼에서 사용할 수 있는 고급 기능들이 추가되었다. 스마트 TV 앱 검증기능 역시 추가된 고급 기능들에 따라 확장될 필요성이 있기 때문에 TTA 에서는 스마트 TV 2.0 표준의 기능을 추가하여 앱 검증기능을 확장 구현 하였다.

두 번째 확장기능은 스마트 TV 애플레이터와 스마트 TV 앱 개발도구와의 인터페이스 구현에 대한 내용이다. 스마트 TV 애플레이터는 TV 장치 없이 PC 개발환경에서 표준기반으로 개발된 앱을 실행할 수 있는 TV 애플레이터이다. 기존에 스마트 TV 애플레이터와 스마트 TV 앱 개발도구는 독립적으로 구현이 되어있었으나, 본 논문에서 스마트 TV 앱 개발도구에 애플레이터와의 인터페이스를 구현하여 개발도구 안에서 앱을 실행시킬 수 있는 환경을 구현하였다.

본 논문은 다음과 같이 구성된다. 2 장에서는 스마트 TV 앱 검증기능의 확장에 대한 내용을, 3 장에서는 스마트 TV 애플레이터와 개발도구의 인터페이스 구현 내용을 다룬다. 5 장에서는 확장된 개발도구의 기능 테스트 및 결과 내용을 다루고, 마지막으로 6 장에서는 논문의 결론을 맺는다.

¹ 본 연구는 미래창조과학부 및 정보통신기술진흥센터의 정보통신·방송 연구개발사업의 일환으로 수행하였음. [B136-16-1002, 방송융합 기반기술 테스트 환경구축]

2. 스마트 TV 앱 검증기능 확장 구현

2016년 6월 스마트 TV 2.0 표준은 고급입력 API, 녹화 및 다운로드 API 등 사용자들의 편의성과 표준의 확장성을 고려한 고급 기능들이 추가되었다. 기존 스마트 TV 앱 검증기능은 스마트 TV 2.0 표준의 추가 기능에 대한 검증은 지원을 하지 않는다. 따라서 다음과 같은 구현을 통하여 추가 표준기술에 대한 내용도 표준기반 서비스를 개발하는 개발자들이 검증기능에서 사용할 수 있도록 한다.

검증기능은 추가된 인터페이스에 대한 정의를 규칙으로 만들어 검증기능의 실행 시 정의된 규칙을 체크함으로써 개발자들이 개발한 코드가 표준에서 정의한 속성 및 기능을 올바르게 사용하고 있는지 검증할 수 있다.

가. 고급입력 API

고급입력 API는 스마트 TV 앱에서 리모콘이 아닌 사용자의 동작과 음성 같은 스마트 TV의 고급 입력을 활용할 수 있도록 지원하는 인터페이스를 정의한다. 스마트 TV 수신기는 카메라 혹은 마이크와 같은 입력 장치를 통해서 사용자로부터 동작과 음성에 대한 정보를 입력 받을 수 있으며, 이러한 정보를 스마트 TV 앱에 전달하기 위한 인터페이스를 표준으로 정의하고 이 표준에 따라서 입력 정보를 앱에 전달할 수 있다. 스마트 TV 앱은 기존의 리모콘이나 포인팅 디바이스의 제한된 입력 방식을 넘어 다양한 사용자 동작과 음성 정보를 활용한 앱 시나리오를 구현함으로써 사용자 상호작용을 더욱 강화할 수 있다.

다음과 같은 기능들을 제공하기 위해 스마트 TV 2.0 표준에는 사용자 동작 입력정보를 전달하기 위한 InputGesture 인터페이스와 사용자 음성 입력정보를 전달하기 위한 VoiceGesture 인터페이스를 정의하고 있다. 검증기능에서 고급입력 API 인터페이스를 체크할 규칙을 정의하기 위해 검증기능 프로젝트에 inputgesture.js 파일과 voicegesture.js 파일을 추가하였다. inputgesture.js 파일은 InputGesture 인터페이스의 속성 및 이벤트 핸들러에 대한 규칙을 정의해둔 파일이다. voicegesture.js 파일은 VoiceGesture 인터페이스에 정의된 속성 및 이벤트 핸들러에 대한 규칙을 정의해둔 파일이다. 다음 그림 1은 inputgesture.js 파일에 정의된 InputGesture 인터페이스 규칙 코드이다.

```

6  /**
7   * InputGesture API
8   */
9  function InputGesture (){};
10 InputGesture.EVENT_GESTURE_BEGIN      = 0x1;
11 InputGesture.EVENT_GESTURE_END        = 0x2;
12 InputGesture.EVENT_GESTURE_HAND_UP    = 0x10;
13 InputGesture.EVENT_GESTURE_HAND_REFT = 0x20;
14 InputGesture.EVENT_GESTURE_HAND_RIGHT = 0x40;
15 InputGesture.EVENT_GESTURE_HAND_OK    = 0x100;
16 InputGesture.EVENT_GESTURE_HAND_SHAKE = 0x200;
17 InputGesture.EVENT_GESTURE_2HAND_ZOOM = 0x400;
18 InputGesture.EVENT_GESTURE_2HAND_ROTATE = 0x800;
19 InputGesture.EVENT_GESTURE_OTHERS     = 0x1000;
20
21
22
23 /**
24 * received gesture event
25 * @param {Channel} channel
26 */
27 InputGesture.prototype.onReceivedGestureEvent = function(event){};
28
29

```

그림 1 inputgesture.js 파일

나. 녹화 및 다운로드 API

녹화 및 다운로드 API는 스마트 TV 플랫폼에서 실시간 방송 콘텐츠에 대한 녹화(Record) 및 인터넷 망을 통한 콘텐츠 다운로드(Download) 기능을 지원하기 위한 인터페이스를 정의한다. 스마트 TV 앱에 의해서 녹화 및 다운로드된 콘텐츠는 수신기의 저장 장치에 저장되며, 저장된 콘텐츠는 HTML5 <video> 요소를 이용하여 VoD 스트리밍 콘텐츠와 유사하게 재생 및 제어를 수행할 수 있다. 또한, 인터페이스를 통해서 저장된 콘텐츠를 조회할 수 있으며, 저장 장치에서 콘텐츠를 삭제하는 기능도 함께 정의한다.

녹화 및 다운로드 API 기능을 제공하기 위해 표준에서는 콘텐츠 녹화에 대한 정보를 전달하기 위한 ContentsRecord, Record, RecordList 인터페이스와, 콘텐츠 다운로드에 대한 내용을 전달하기 위한 ContentsDownload, Download, DownloadList 인터페이스가 정의되어 있다. 검증기능에서 녹화 및 다운로드 API에 대한 규칙을 정의하기 위해 검증기능 프로젝트에 contentsrecord.js 와 contentsdownload.js 두 가지 파일을 만들었다. contentsrecord.js 파일은 콘텐츠 녹화에 대한 정보를 전달하기 위한 ContentsRecord, Record, RecordList 인터페이스에 대한 규칙을 가지고 있으며, contentsdownload.js 파일은 콘텐츠의 다운로드에 대한 내용을 전달하기 위한 ContentsDownload, Download, DownloadList 인터페이스에 대한 규칙을 가지고 있어 관련 API에 대한 검증을 가능하게 하였다. 그림 2은 ContentsRecord 규칙의 예를 통해 구현 내용을 보여주고 있다.

```

4  /**
5   * ContentsRecord API
6   */
7  function VoiceGesture (){};
8  VoiceGesture.RECORD_REC_SCHEDULED      = 0x1;
9  VoiceGesture.RECORD_REC_STARTED        = 0x2;
10 VoiceGesture.RECORD_REC_COMPLETED      = 0x4;
11 VoiceGesture.RECORD_REC_PARTIAL_COMPLETED = 0x8;
12 VoiceGesture.RECORD_REC_DELETED        = 0x10;
13 VoiceGesture.RECORD_REC_FAILED          = 0x20;
14 VoiceGesture.RECORD_ERR_NONE            = 0x100;
15 VoiceGesture.RECORD_ERR_RESOURCE        = 0x200;
16 VoiceGesture.RECORD_ERR_STORAGE         = 0x400;
17 VoiceGesture.RECORD_ERR_UNKNOWN         = 0x800;
18
19
20 /**
21 * record program
22 * @param {Program} program
23 */
24 VoiceGesture.prototype.recordProgram = function(program){};
25
26 /**
27 * record in channel
28 * @param {unsigned long} starttime
29 * @param {unsigned long} duration
30 * @param {unsigned short} repeatDays
31 * @param {DOMString} sourceURI
32 */
33 VoiceGesture.prototype.recordInChannel = function(starttime, duration, repeatDays, sourceURI){};
34

```

그림 2 contentsrecord.js 규칙의 예

3. 스마트 TV 개발도구와 애플레이터의 연동

스마트 TV 앱 개발도구는 스마트 TV 앱 개발자가 표준에서 정의한 인터페이스를 활용하여 스마트 TV 앱을 쉽고 빠르게 개발 검증할 수 있는 개발 환경을 제공하고 있다. 개발도구를 통해 개발된 앱은 인터페이스가 코드 안에서 정확히 구현되고 있는지에 대한 소프트웨어 적인 측면만 고려가 되어, 실제로 표준의 기능을 적용시킨 스마트 TV 혹은 수신기를 통한 동작의 유효성 검증이 된 것은 아니다. 따라서 표준이 적용된 디바이스가 없는 개발자들은 실제 앱 동작의 유효성 검증이 어렵기 때문에 TTA에서는 스마트 TV 앱 실행 환경을 제공하기 위한 표준기반 스마트 TV 애플레이터를 구현하였다.

스마트 TV 앱 개발도구와 스마트 TV 에뮬레이터는 서로 독립적으로 구현이 되어있다. 따라서 스마트 TV 앱 개발도구를 통해 구현된 앱을 다시 에뮬레이터를 통해 실행시키는 번거로움이 존재한다. 이를 개선하기 위해 스마트 TV 앱 개발도구 환경과 에뮬레이터를 연동시키는 방법을 통해 그 개선 방법을 제공할 수 있다.

스마트 TV 앱 개발도구는 Eclipse [6]에서 플러그인 설치를 통해 개발 환경을 구축한다. 따라서 외부 프로그램과 연동되는 기능을 구현하기 위해 Eclipse 에서 제공하는 External Tools [7]을 이용하여 구현하였다.

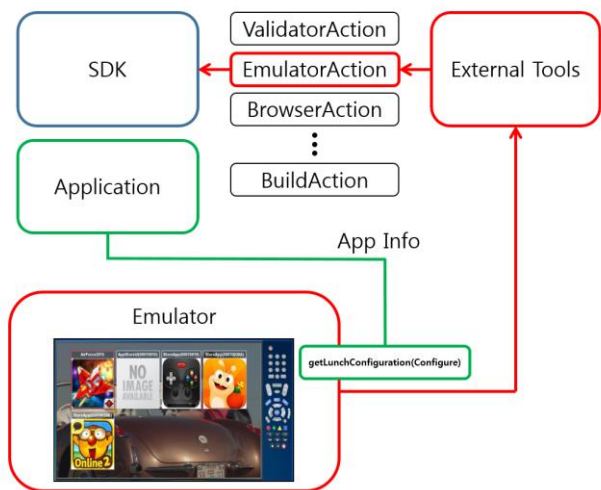


그림 3 스마트 TV 에뮬레이터 연동 구조도

그림 3은 스마트 TV 에뮬레이터와 개발도구의 연동 구조도를 나타낸다. 개발도구는 앱 검증기능, 브라우저 실행도구, 앱 패키지도구 등 다양한 도구들을 가지고 있다. 여기에 에뮬레이터 실행 도구 모듈을 만들어 에뮬레이터를 실행할 수 있는 기능을 추가시켰다. 그림의 EmulatorAction 에 해당하는 에뮬레이터 모듈은 External Tools 를 통해 에뮬레이터를 실행 시킨다. External Tools 은 실행시키기 전에 표준기반 스마트 TV 앱의 프로젝트 정보를 함께 가지고 에뮬레이터를 실행 시키게 된다.

다음과 같은 방법을 통하여 스마트 TV 개발도구는 개발도구 환경에서 사용자가 개발한 서비스를 에뮬레이터를 통해 실행 시킬 수 있다. 또한 스마트 TV 앱 개발도구에 포함된 에뮬레이터의 기능을 개발자들이 보다 쉽게 사용할 수 있도록 개발도구의 툴바를 이용하여 실행 시킬 수 있는 방법과 스마트 TV 앱 메뉴를 통해서 실행시키는 방법 두 가지를 제공한다.

4. 개발도구 기능 테스트 및 결과

가. 스마트 TV 앱 검증기능

스마트 TV 앱 개발도구는 다음과 같이 검증기능을 Eclipse 에서 실행 할 수 있는 방법을 제공한다. 그림 4은 검증기능 실행방법을 보여주고 있다.

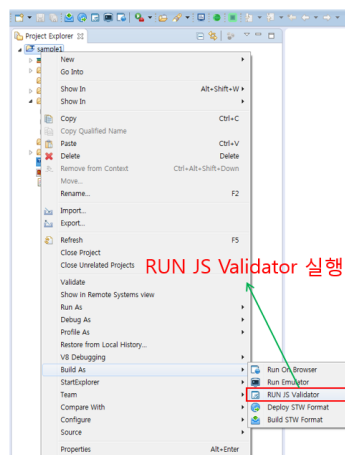


그림 4 검증기능 실행방법

검증기능 테스트를 위해, 스마트 TV 2.0 표준의 확장 기능을 사용하는 샘플코드를 통해 구현된 검증기능이 정상적으로 작동하는지 테스트 하였다.

검증을 위해 만든 샘플 코드는 신규 API 를 활용하여 코드를 작성하였으며, 앱 검증기능이 정확히 작동하는지 테스트를 하기 위하여 잘못된 속성을 가지는 코드들을 함께 사용하였다. 다음 샘플 코드를 검증기능으로 실행시킨 결과는 다음과 같다.

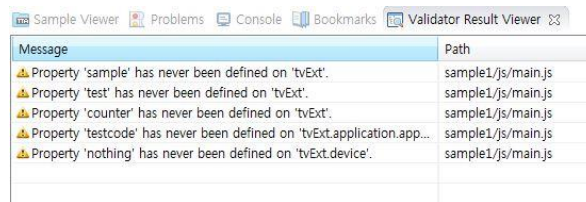


그림 5 검증기능 실행결과 창

그림 5는 검증기능 실행결과를 보여준다. tvExt 에 정의되어있지 않은 속성인 sample, test, counter 가 잘못 된 것을 확인하였으며, tvExt.application.appmgr 에 있지 않은 속성인 testcode 역시 선언 되지 않은 것을 확인할 수 있었다. 그러나, 추가가 된 표준 기능들에 대한 내용은 문제가 없어 결과 창에 표시가 되지 않음을 그림 5를 통해 확인을 할 수 있다.

나. 스마트 TV 개발도구와 에뮬레이터의 연동

스마트 TV 에뮬레이터와 스마트 TV 개발도구의 연동을 구현함에 따라 사용자는 개발도구 환경에서 실행 할 수 있게 된다.

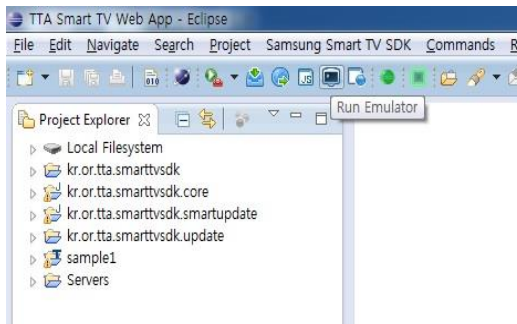


그림 6 툴바를 이용한 에뮬레이터 실행

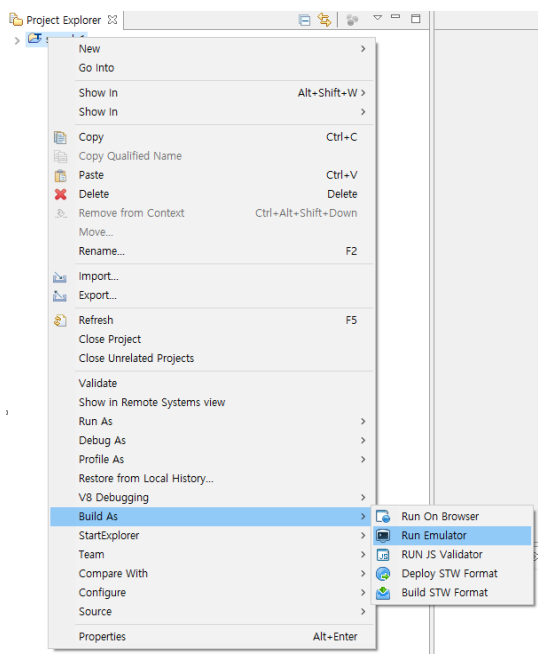


그림 7 프로젝트 메뉴를 통한 애플레이터 실행

그림 6과 그림 7은 툴바를 이용한 애플레이터 실행과 프로젝트 메뉴를 통한 애플레이터 실행 방법을 보여준다. 위 방법을 통해서 애플레이터를 실행 한 결과 실행 결과는 다음과 같다.

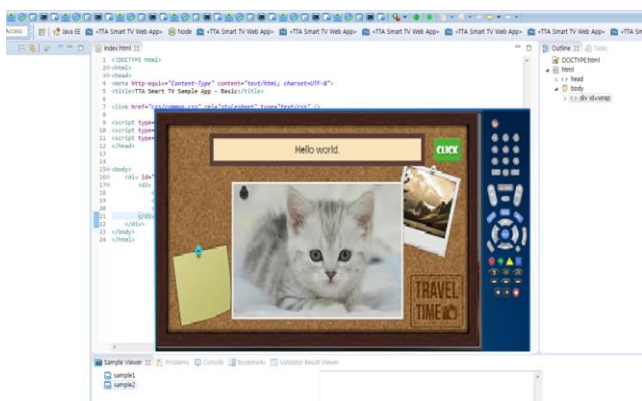


그림 8 스마트 TV 애플레이터 실행 화면

그림 8은 스마트 TV 개발환경에서 애플레이터를 실행시킨 실행 화면이다. TTA 개발도구에서 제공하는 샘플 앱이 애플레이터 안에서 정상적으로 실행되는 것을 확인 할 수 있다.

5. 결론

본 논문에서는 표준기반 서비스를 만드는 개발자에 대한 지원과 표준기반 TV 앱 생태계의 활성화를 위한, 스마트 TV 플랫폼 표준 앱 개발도구 확장기능의 설계 및 구현에 대하여 소개하였다.

개발자는 확장된 검증기능을 통해 스마트 TV 2.0 표준의 추가기능 검증이 가능하며, 개발도구에서 애플레이터의 실행이 가능해짐에 따라 표준이 적용된 디바이스 없이 앱 동작의 유효성에 대한 검증 또한 가능하게 되었다. 이에 스마트 TV 앱

개발환경은 개발에 드는 정합시간 및 비용은 줄었으며 개발의 편의성은 더욱 높아졌다. 본 논문에서 소개한 개발도구의 확장을 통해 표준기반 스마트미디어 서비스가 널리 개발되고 상용화 되기를 기대한다.

앞으로 TTA 는 확장 개발된 스마트 TV 앱 개발도구를 더욱 많은 앱 개발사에게 배포하고 개발사의 요구사항을 적극 반영하여 파편화된 스마트 TV 앱 생태계를 더욱 활성화 시킬 수 있도록 지원할 예정이다.

참고 문헌

- [1] TTA.KO.07-0111/R2, “HTML5 기반 스마트 TV 플랫폼”, 2016.06
- [2] 이동훈, “HTML5 기반 스마트 TV 플랫폼 표준”, TTA Journal Vol.147, 2013.05
- [3] 박동영, 이동훈, “스마트 TV, 이제 표준과 손 잡아야 할 때 (개방형 생태계로의 진화를 위한 스마트 TV 표준화)”, SW 지식채널 칼럼 25 호
- [4] 황희선, 김호년, 이동훈, 박동영, “HTML5 기반 스마트 TV 플랫폼 표준 앱 개발도구 개발”, 2015 년 방송공학회 하계학술대회, 2015. 06
- [5] 황희선, 김호년, 이동훈, 박동영, 이은향, “HTML5 기반 스마트 TV 플랫폼 표준 앱 검증 도구 개발”, 2014 년 방송공학회 하계학술대회, 2014. 06
- [6] Eclipse Homepage “<https://eclipse.org/>”
- [7] Eclipse External Tools
“ http://help.eclipse.org/mars/topic/org.eclipse.platform.doc.user/concepts/concepts-exttools.htm?cp=0_2_10_1”