

마이 스토리(2) - 구글 “안드로이드 개발자 챌린지” 참가를 준비하며

2007년 11월 21일

1. 들어가며

구글은 2007년 11월 12일(현지시간) 안드로이드라는 모바일 플랫폼과 관련된 SDK와 샘플, 그리고 관련 소스들을 공개했다. 그리고, 전 세계 개발자들을 대상으로 총 상금 1000만 달러 규모의 안드로이드 개발자 챌린지(Android Developer Challenge)를 개최하겠다고 발표했다. 이번 개발자 챌린지는 여러가지 측면에서 많은 개발자들과 관련 회사들의 참여 욕구를 자극하는 듯 하다. 내 개인적으로도 지금까지 개발해 온 기술의 가능성을 검증할 수 있는 좋은 기회라는 생각으로 안드로이드 개발자 챌린지에 참여하기로 결정했다. 이 글은, 이번 개발자 챌린지를 준비하면서 정리해 본 글이다. (*이번 대회를 준비하는 다른 이들에게 작은 도움이 되었으면 한다.)

2. 구글 안드로이드 개발자 챌린지(Google Android Developer Challenge) 개요

안드로이드 개발자 챌린지는 크게 1차와 2차로 구성된다. 2차는 2008년 하반기에 예정되고 있으나, 아직 구체적인 일정이 발표되지 않은 이유로 굳이 언급할 필요가 없는 듯하다. 일단 1차 대회는 다음과 같은 일정으로 진행된다.

일 정	내 용	비 고
2008년 1월 2일	1차 예선 공모 접수 시작	
2008년 3월 2일	1차 예선 공모 접수 마감	
2008년 3월 말	본선 출품 당선작 발표 - (총 50개 당선작 발표)	상금 : 50개(각 \$25,000)
2008년 5월 1일	1차 본선 접수 마감	자격 : 본선 출품작 50개에 한함
2008년 5월 말	최우수 당선작 및 최종 당선작 발표 - (총 20개 당선작 발표)	상금 : 10개(각 \$275,000) 10개(각 \$100,000)

안드로이드 개발자 챌린지와 관련해서 고려해 볼 만한 사항은 다음과 같은 것들이 있는 것 같다. 일단 출품된 애플리케이션은, 안드로이드의 특성을 얼마나 잘 활용하여 사용자들에게 보다 나은 모바일 경험을 제공할 수 있는가에 의해 평가된다. 이와 관련해서 구글에서 제시한 출품주제 영역은 다음과 아래와 같은 것들이 있지만, 기본적으로 그 영역에 대한 제약은 존재하지 않는다. (아래의 주제영역은 참고 차원에서 한번쯤 살펴볼 만한 듯은 하다.)

주 제 영 역	비 고 (필자의견)
Social networking	사회적 네트워킹 영역은, 쉽게 생각하면 모바일 사이트와 같은 것이라고 생각해도 될 듯하다. 어떤 이들은 구글 Android의 목표가 모바일 폰이 아니라 모바일 인프라와 결합된 글로벌 facebook 사이트라는 의견을 제시하는 사람들도 있다.
Media consumption, management, editing, or sharing, e.g., photos	동영상이나 사진 등과 같은 매체에 대한 활용, 관리, 편집, 공유 등에 대한 것으로 이와 비슷한 모바일 애플리케이션은 많이 볼 수 있는 듯 하다.
Productivity and collaboration such as email, IM, calendar, etc.	전자메일, 일정관리, 메신저 등을 모바일과 연동함으로써, 생산성이나 협력을 증대시키는 애플리케이션에 대한 것으로, 이러한 영역들 역시 우리 주변에서 많이 볼 수 있는 듯 하다.
Gaming	이 영역은, 모바일 게임에 해당할 것이다.
News and information (weather, traffic,	이 영역은 최근 모바일 대기화면이나 모바일 위젯에서 많은 사례들이

sports, stocks, etc.)	나오고 있는 듯하다.
Rethinking of traditional user interfaces	모바일 인터페이스에 대해서는, 이미 애플이 iPhone 핑커터치 인터페이스를 통한 확대/축소, 스크롤 및 block 기반의 확대/축소 최적화 등을 선보인 것 같다. 이 부분에 대해서는 구글 Android 도 많은 부분 모방을 하였으나, 보다 혁신적인 사용자 인터페이스는 매우 의미있는 영역이 될 듯 하다.
Use of mash-up functionality	아마도 이 부분의 가장 대표적인 사례는 구글 마이맵일 듯 하다. 구글 마이맵은 구글맵에 다양한 가젯을 추가한 서비스이다.
Humanitarian benefits (monitoring and response for diseases, climate change, natural disasters, etc)	이 영역은, 인도주의적 이익을 추구하는 것들로 질병, 기후변화, 자연재해 등에 대한 관측 및 대응에 대한 부분들이다.
Applications in service of global economic development for the 3 billion people living on less than \$2.00 per day	이 영역은, 돈이 없어서 컴퓨터의 혜택을 누리지 못한 저개발국가의 아이들을 위해 100 달러 미만의 컴퓨터를 만들고 있는 시도와 비슷한 듯하다. 하루 2 달러 미만으로 살고 있는 30억 인구를 위한 글로벌 경제 개발 서비스를 위한 애플리케이션 !.

더불어, 출품할 수 있는 자격은 개인 또는 단체, 그리고 비즈니스 주체 등이 가능하기 때문에 모든 이에게 부여되어 있다고 할 수 있다. (단, 미 연방법의 제약이나, 해당국의 제약이 없어야 하며, 구글에서 일하는 사람과 Open Handset Alliance 에 참여하는 회사와 관련 협력회사는 참여할 수 없으며, 해당 국가에서 인정하는 법적 성년에 한하여 출품자격이 있다.)

출품작의 수에 대해서는 제약이 없으나, 구글이 추천하는 바는 양보는 짝이며, 출품작에서 사용하는 언어에 대한 제약도 없으나, 언어에 대한 기본적인 지원을 포함하고 있어야 한다.

마지막으로 출품된 작품에 대한 모든 지적재산권과 제반 권한들은 개발자가 보유하게 되며, 이번 챌린지를 위해 구글은 해당 출품작에 대한 평가 및 테스트 라이선스를 부여 받게 되기 때문에 저작권과 관련해서도 특별한 제약이 존재하지 않는다.

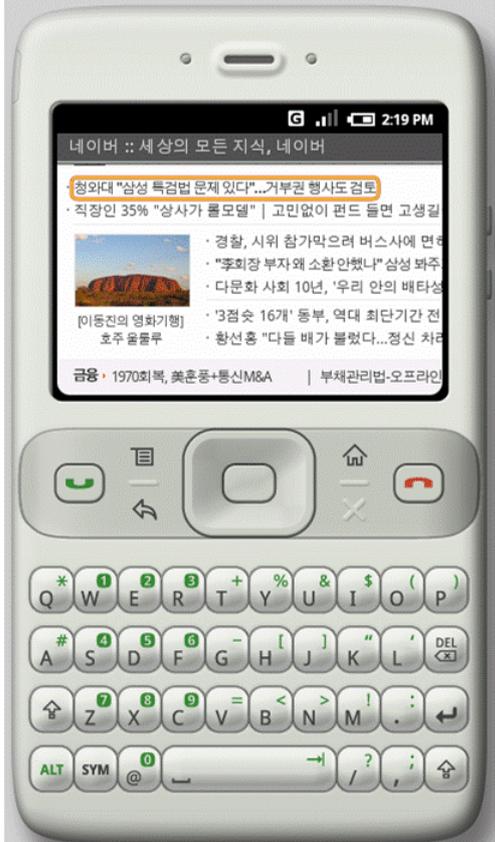
3. 구글 안드로이드 기반 애플리케이션 개발 시 고려사항.

3-1. 개발 환경 설정 및 예제 애플리케이션 제작

개발환경 설정은 간단하다. 기본적으로 구글에서 제공하는 Android SDK 를 설치하고, Eclipse Plug-in 을 설치/설정 후, 애플리케이션을 제작하여 Android SDK 에 있는 에뮬레이터 상에서 실행하면 된다. 이와 관련해서는 Google 의 Android 사이트를 참조하면 된다.

(구글 안드로이드 SDK 설치 가이드 : <http://code.google.com/android/intro/installing.html>)

아래의 그림은 필자가 위의 절차를 통해 간단히 만들어 본 애플리케이션으로, 구글 안드로이드 에뮬레이터상에서 실행해 본 화면 스크랩이다.

	
<p>위의 그림은 안드로이드 상에서 한글입력이 가능하도록 하기 위해, 한글 입력 키패드를 가상으로 구현한 것임. 더불어 간단한 텍스트 에디터를 구현하였음.</p>	<p>위의 그림은 안드로이드 상에서 작동하는 웹 브라우저상에서 한글 인코딩 문제를 지원하도록 하여 한글 웹 사이트가 보일 수 있도록 처리한 예제 애플리케이션임.</p>

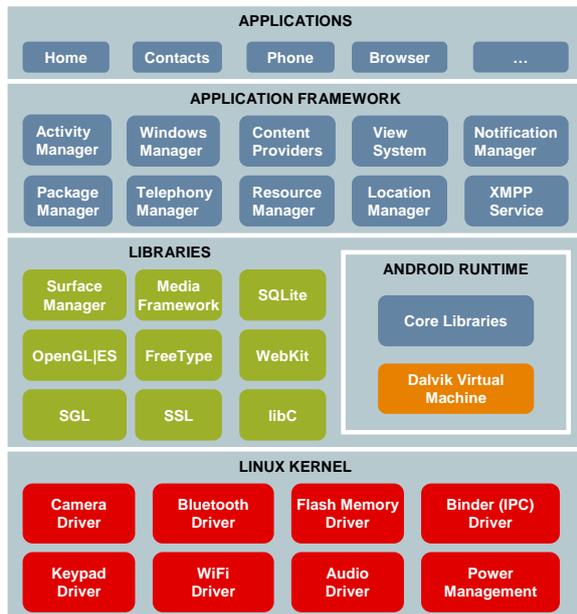
3-2. 구글 안드로이드에 플랫폼에 대한 이해

오른쪽의 그림은 안드로이드 아키텍처에 대한 것이다. 안드로이드는 Linux Kernel 버전 2.6을 기반으로 하고 있다. 이 기반 위에 다양한 C/C++ Library 들과 Android Run-time 이 존재한다. Android Run-time 은 Java 프로그래밍 언어의 Core Library 들에서 유용한 대부분의 functionality 들을 제공하는 Core Library 들과 Dalvik 이라는 Java 가상 머신을 포함하고 있다. Dalvik 는 .dex 라는 포맷의 Dalvik Executable Format 을 실행한다. .dex 파일은 'dx' 도구에 의해 생성되며, Dalvik 은 .dex 파일내에 탑재된 코드를 Java 컴파일러로 컴파일한 결과물인 Class 를 실행하게 된다.

이러한 Library 들과 Android Run-time 위에 애플리케이션 프레임워크가 존재한다. 이는 이미 탑재된 Core 애플리케이션들에 의해 사용되고 있는 것과 동일한 프레임워크 API 를 개발자들에게 제공하며, 더불어 해당 컴포넌트를 개발자들이 대체할 수 있는 구조를 지원한다.

또한, 안드로이드는 email Client, SMS 프로그램, 캘린더, Map, Contacts, 브라우저, 기타 등등의 Core 애플리케이션들을 탑재하고 있다.

여기에서 한가지 주의할 점은, 구글 안드로이드가 오픈 플랫폼이지 완전한 오픈 소스는 아니라는 것이다. 현재 상태에서 안드로이드 플랫폼을 오픈 소스차원에서 정리해 보면 다음과 같다



주요 Platform Stack	소스 공개유무	비 고
Application	비공개	이 부분은 비공개 영역이지만, Application Framework API 를 통해 개발될 것이기 때문에, 동일한 수준이 애플리케이션 제작은 가능함.
Application Framework	비공개	이 부분은 안드로이드 SDK 의 주요 API 영역이나 모두 비공개되어 있음. 단 overriding 을 통해 대체는 가능함.
Android Run-time	비공개	이 부분은 안드로이드 SDK 에서도 동일하게 사용되고 있으나 모두 비공개 영역으로 존재함.
Libraries	공개 및 비공개	http://android.googlecode.com/files/webkit-522-android-m3-rc20.tar.gz 이 부분에는 다양한 Free s/w 들이 활용되고 있으나, 구글에서 수정한 영역중에는 Webkit 소스만 공개되어 있음.
Linux Kernel	공개	http://android.googlecode.com/files/linux-2.6.23-android-m3-rc20.tar.gz
Android SDK	공개 및 비공개	http://android.googlecode.com/files/android-emulator-m3-rc20.tar.bz2 http://android.googlecode.com/files/adt.m3-rc20.tar.gz 현재, 에뮬레이터와 Eclipse Plug-in 소스만 공개되어 있음.

3-3. 구글 안드로이드 API 이해

다음은 안드로이드 API 관련 정보임. (단, java.*, javax.*, org.*, junit.* 패키지는 제외했음)

android	Contains the resource classes used by standard Android applications.
android.app	High-level classes encapsulating the overall Android application model.
android.content	Contains classes for accessing and publishing data on the device.
android.database	Contains classes to explore data returned through a content provider.
android.database.sqlite	Contains the SQLite database management classes that an application would use to manage its own private database.
android.graphics	Provides low level graphics tools such as canvases, color filters, points, and rectangles that let you handle drawing to the screen directly.
android.graphics.drawable	Provides classes to manage a variety of visual elements that are intended for display only, such as bitmaps and gradients.
android.graphics.glutils	Provides a variety of classes to enable using OpenGL for embedded systems (OpenGL ES) to draw graphics on an Android device.
android.hardware	Provides support for hardware devices that may not be present on every Android device.
android.location	Classes defining Android location-based and related services.
android.media	
android.net	Classes that help with network access, beyond the normal java.net.* APIs.
android.opengl	Provides OpenGL utilities.
android.os	Provides basic operating system services, message passing, and inter-process communication on the device.
android.provider	Provides convenience classes to access the content providers supplied by Android.
android.sax	A framework that makes it easy to write efficient and robust SAX handlers.
android.speech.recognition	Provides classes for speech recognition.
android.telephony	Provides tools to make, receive, and monitor phone calls and phone status.
android.telephony.gsm	Provides classes to control or read data from GSM phones.
android.text	Provides classes used to render or track text and text spans on the screen.
android.text.method	Provides classes that monitor or modify keypad input.
android.text.style	Provides classes used to view or change the style of a span of text in a View object.
android.text.util	
android.util	Provides common utility methods such as date/time manipulation, base64 encoders and decoders, string and number conversion methods, and XML utilities.
android.view	Provides classes that expose basic user interface classes that handle screen layout and interaction with the user.
android.view.animation	Provides classes that handle tweened animations.
android.webkit	Provides tools for browsing the web.
android.widget	The widget package contains (mostly visual) UI elements to use on your Application screen.
com.google.android.maps	
com.google.android.xmmpService	

4. 구글 안드로이드 개발자 챌린지에 대한 응모 계획

4-1. 나의 고민 - 현재 보유한 기술을 통한 출품작 경쟁력 확보 검토.

아래는 지금까지 모바일 웹 브라우저를 개발해 오면서 축적한 기술 항목에 대한 내용이다. 하지만 아래의 모든 영역은 기본적으로 ANSI-C 기반으로 만들어진 반면, 이번 안드로이드 개발자 챌린지는 기본적으로 Java 애플리케이션만을 허용하고 있다. 더불어 기본적으로 웹브라우저를 가능하게 하는 Webkit 기반의 브라우저가 탑재되어 있고, 기본적인 2D/3D 그래픽 엔진과 안드로이드 레이아웃팅을 위한 XML 리소스 처리 기능을 제공하고 있기 때문에 아래의 기술들을 활용하여 출품작의 경쟁우위를 만드는 것은 그렇게 쉬운 문제는 아닌 듯하다.

주요영역	세부항목	설명
모바일웹 G/W Server	Web Robot	Async/Sync Non-blocking Network IO
	분산 Web Robot	위의 Web Robot 기반의 분산 아키텍처 지원 (Support Virtual Pipelining for High Performance Net. IO)
	SGML/XML/HTML Parser	통합된 SGML/XML/HTML Parser
	CSS Engine	CSS 2.1 기반의 Parser 및 Apply Processor
	JavaScript Engine	SpiderMonkey 에 대한 Embedding Wrapper
	HTML DOM Engine	HTML DOM 구현 및 Simple Garbage Collector
	HTML Layout Engine	HTML 4.01 기반의 레이아웃 엔진
	Font Engine	Font 메타 정보 및 Glyph Rendering Engine
	Charset/Encoding Engine	Iconv 기반 Charset 변환 및 Encoding Auto Detection
	XSL Engine	XSLT, XSL-FO, Xpath 등에 대한 프로세서
	HTML Conversion Engine	XHTML,PDF,Image, Meta 등으로의 HTML 변환
	Other tools.	Cookie, Preferences, Bookmark, SSL 등 지원.
모바일웹 Thin Client	Meta Format Parser	Vector Rendering 을 지원하는 메타 포맷 Parser
	SGML/XML/HTML Parser	통합된 SGML/XML/HTML Parser
	Font Engine	Font 메타 정보 및 Glyph Rendering Engine
	Rendering Engine	2D-Graphic Rendering 엔진
	Simple UI Widget	Window, Menu, SelectBar, Radio, Checkbox, TextView, TextEdit 에 대한 간단한 Widget Tool.
	Mobile Interface	Zoom In/Out, SubArea Layouting, 자동 스크롤, Text 보기 등.
모바일 위젯 Library	XUL Parser	Self Defined XUL 구조에 대한 Parser
	XUL Layout Engine	Self Defined XUL 구조에 대한 Layout 엔진
	XUL Rendering Engine	Self Defined XML 객체에 대한 2D Graphic Rendering 엔진
	Event Handler	이벤트 Dispatch & Handler
기타	한글 조합형 벡터 Font Engine	한글 조합형 벡터 폰트의 생성 및 실시간 Glyph Rendering
	PDF Viewer	PDF 파서 및 PDF 객체 Rendering, 그리고 UI
	Dynamic Input Method Engine	다양한 Automata 기반이 Input Method 제공
	SVG Viewer	SVG 파서 및 SVG 객체 Rendering, 그리고 UI

4-2. 출품작 경쟁우위를 확보하기 위한 전략

안드로이드 챌린지에 “어떤 주제의 작품을 제출할 것인가?”에 앞서서, “어떤 종류의 작품을 출품하는 것이 우리에게 경쟁력이 있는가?” 라는 질문을 해 볼 필요가 있다. 이런 필요성에 대해 좀 더 구체적으로 접근하기 위해, 구글이 제시한 출품작 주제영역 예를 간단히 살펴보자. 앞서서 언급했던 구글의 출품작 주제영역 예를 기술적인 차원에서 구분해 보면 다음과 같이 구분될 수 있다.

- 유형 1 : 안드로이드 플랫폼상에서 구동하는 단일 애플리케이션
- 유형 2 : 안드로이드 플랫폼상에서 구동되는 특정 애플리케이션과 연동되어야 하는 애플리케이션
- 유형 3 : 안드로이드 플랫폼상에서 구동되는 모든 애플리케이션에 적용될 공통 영역 애플리케이션
- 유형 4 : 안드로이드 플랫폼상에서 구동되지만, 구글의 정신(?)을 잘 반영한 아이디어 기반 애플리케이션

위와 같은 방식으로 유형을 분류한 것은, 우리가 이번 대회에 참가하면서 갖게 되는 핸디캡을 분명히 하기 위함이다. 먼저 가장 큰 핸디캡은, 한국어에 대한 지원의 부족과 한국인임에도 불구하고 영어와 영어권 문화기반의 콘텐츠를 기반으로 애플리케이션을 제작해야 하는 조건이다. 이것은 구글 안드로이드의 i18n 관련 영역이 여전히 미완성 상태에서 기인하며, 더불어 전세계를 대상으로 한 대회이기 때문에 영어를 표준으로 채택할 수 밖에 없는 상황 때문일 것이다.

(*internationalization 의 약어로 처음 ‘i’ 자와 끝 ‘n’ 자 사이에 18 개의 글자가 있다는 의미의 약어임)

(참고: i18n 에 대한 문제는 기본적으로 Input Method, Charset/Encoding, 폰트에 대한 문제이지만, PC 에서와 같이 단순한 문제는 아닌 듯하다. 왜냐하면, 휴대폰에 이를 적용할 경우에는 Key Pad 나 Windows Mobile 에서와 같은 SIP(Soft Input Panel) 과 같은 가상의 Input 도구가 제공되어야 하는데, 이와 관련된 자판배열이 특허권과 관련되어 있기 때문에 단순한 문제는 아닐 수 있다.)

그리고 위의 유형 2,3 영역은, 안드로이드에서 제공하는 SDK 범위 및 안드로이드의 구조에 의해 제약을 받기 때문에 구현 자체가 어떤 부분에선 불가능할 수 도 있다. 이런 차원에서 볼 때, 이번 개발자 챌린지가 주제영역 면에서 제한이 없다고 하지만, 실제로는 대부분 유형 1, 4로 국한될 수 밖에 없는 한계가 있는 듯하다.

하지만, Stand-alone 애플리케이션을 출품할 경우엔 언급한 핸디캡들이 존재하기 때문에, 핸디캡을 없애기 위해서 가급적 유형 2, 3 영역 관련 작품을 출시해야 한다고 본다. (단, 영어와 영어권 문화 및 비즈니스 환경에 자신이 있다면, 이런 고민은 필요 없을 수 있다.)

이슈 1) 유형 2, 3 영역 애플리케이션 개발 시, 안드로이드의 SDK 범위와 구조 제약을 어떻게 극복할 것인가?

이런 이슈와 관련해서 두 가지 이야기들이 나오고 있는 것 같다. 하나는 안드로이드가 C/C++ 애플리케이션을 지원하지 않는다는 지적이고, 다른 하나는 안드로이드는 개방형 플랫폼이긴 하지만, 진정한 오픈 소스가 아니라는 지적들이다. 이 두 가지 모두에 대해서 구글의 입장은 분명한 듯하다. 이번 개발자 챌린지는 Java 애플리케이션을 대상으로 한다는 것이고, 현재 시점에서 안드로이드 플랫폼 전 영역을 오픈하지는 않겠다는 것이다. 그렇다면 이에 대한 해법은 무엇일까? 내 생각엔, C/C++ 로만 구현할 수 밖에 없는 영역은 서버에서 구현하여 안드로이드 플랫폼상의 애플리케이션과 연동시키는 방법 일 듯 하고, 안드로이드 애플리케이션들과 연동함에 있어서 SDK 제약을 극복하기 위해서는 기존의 모든 안드로이드 애플리케이션을 안드로이드 Framework API 를 사용하여 다시 구현함으로써 해결해야 할 듯하다. 개인적으로는 이번 출품작을 이런 방식으로 만들 생각이다.

4-3. 안드로이드 개발자 챌린지에 출품할 작품 주제 정의

일단 지금까지의 안드로이드와 개발자 챌린지에 대한 생각에 기반해서, 이번에 출품할 작품은 아래의 주제 중 하나를 선택할 생각이다.

〈출품작 주제 1〉 안드로이드 웹 브라우저의 개선 또는 대체

AS-IS :

현재 안드로이드는, KDE의 KHTML/KJS에서 시작되어 애플 iPhone의 WebCore/JavaScriptCore로 변해 온 WebKit 기반의 웹 브라우저를 탑재하고 있다. 하지만, 애플 Safari의 많은 장점들은 Max OS X에 의해, 그리고 애플에서 특히 등록한 사용자 인터페이스와 결합되어 있다. 그리고 이 영역은 WebKit 내에 포함되어 있지는 않다. 이런 이유로 안드로이드에 탑재된 WebKit 기반의 웹 브라우저는 애플 Safari 수준에 미치지 못하고 있다. 더불어 애플 Safari가 지니고 있는 단점, 즉 Cellular Network 상에서의 성능 제약은 안드로이드상에서도 여전히 존재할 수 밖에 없다.

To-BE :

- 안드로이드 웹 브라우저에 애플 Safari 사용자 인터페이스 추가
- 안드로이드 웹 브라우저의 Network 성능 개선을 통해 Cellular Network 상에서의 성능 제약 극복
- 새로운 개념의 모바일 웹 사용자 인터페이스의 추가

〈출품작 주제 2〉 웹 기반 안드로이드 애플리케이션 개발 도구

AS-IS :

현재 안드로이드의 애플리케이션을 개발하기 위해서는 Eclipse를 사용하여 개발해야 한다. 하지만, 기본적인 Layout을 위한 XML 기반의 Resource Layout을 제공하고 있으며 이에 Java Functionality를 결합하는 형태로 안드로이드 애플리케이션을 제작할 수 있게 하고 있다. 하지만, 누구나 보다 쉽게 안드로이드 애플리케이션을 웹 상에서 HTML 제작하듯이 만들 수 있다면 큰 의미가 있을 듯하다.

To-BE :

- XML 기반의 안드로이드 Resource Layout Editor 제공
- Dynamic Interface 및 안드로이드 API 연동구조 마련
- Web 상에서 안드로이드 애플리케이션 에뮬레이터 제공
- 안드로이드 플랫폼상으로 애플리케이션 Download 구조 제공 및 설치 방법 마련

〈출품작 주제 3〉 안드로이드 글로벌 Input Method

AS-IS :

현재 안드로이드는 QWERTY 자판 기반의 Input Method와 Unicode 지원 및 Unicode 기반의 글로벌 폰트 인프라를 탑재하고 있다. 하지만, 각 국가별 Input Method와 Charset Conversion 구조가 지원되지 않고 있다. 또한 Handset 벤더가 이러한 안드로이드 플랫폼을 탑재하기 위해서는 자사가 보유하고 있는 자판 배열 관련 특허와 연동해야만 한다.

To-BE :

- Dynamic Input Automata 관리 구조 추가
- 벤더 및 국가별 Input Method와 안드로이드 Input Method 간의 연동
- 사용자 중심의 Input Method Configuration 기능 제공

4-4. 안드로이드 개발자 챌린지 출품작 개발 일정

출품작을 개발하기 위해선, 어쨌든 험난한 시간을 좀 많이 보내야 할 듯하다. 다른 말이 필요 없을 듯. 단지, 최선을 다 해 보는 수 밖에..~!

(주의 : 항상 미국 현지시각으로 고민 요함)

구 분	일 정	설 명
출품작 주제 확정	~ 2008년 11월 30일	주제확정과 더불어 Functionality 까지 확정 1차 Prototyping 관련 상세 명세서 및 개발일정확정
1차 Prototyping 완료	~ 2008년 12월 30일	1차 Prototyping 완료 및 핵심 이슈 도출/대안 마련
개발 명세서 완료	~ 2008년 1월 15일	개발 명세서 확정
개발 완료	~ 2008년 2월 15일	총 개발 기간 1개월
출품작 테스트	~ 2008년 2월 22일	테스트 1주일
출품작 최종 정리	~ 2008년 3월 1일	최종 정리 1주일
예선 접수 완료	~ 2008년 3월 2일	
2차 Prototyping 완료	~ 2008년 3월 31일	2차 Prototyping 관련 상세 명세서 및 개발일정확정
개발 완료	~ 2008년 4월 25일	총 개발 기간 15일
출품작 테스트 및 최종정리	~ 2008년 4월 30일	테스트 및 최종 정리 약 1주일
본선 접수 완료	~ 2008년 5월 1일	

5. 글을 마치며

“The frequency spectrum is technology, industry, money, culture, and power”

(J.D Bedin in Struzak 2000)

“주파수는 기술, 산업, 돈, 문화, 그리고 힘이다”란 말이 있다. 나는 개인적으로 지금까지의 모바일 환경에 대해서 많은 의문을 갖고 있다. 그것은 공공자원인 주파수를 국가가 특정 기업에게 사용권을 빌려주었을 뿐인데, 그 기업들이 이를 기반으로 다양한 영역에 걸쳐 독점적 지위를 강화하고 있는 현실은 극복되어야 한다고 본다. 분명한 것은, 망개방, 통신사 중심의 요금제, USIM Lock 해제, 플랫폼 개방 등과 관련된 이슈는 공정경쟁 차원에서 분명하게 바로 잡혀야 한다.

더불어, 개인적으로 구글을 찬양하는 사람은 아니지만, 구글의 안드로이드는 긍정적인 측면이 있다고 본다. 그것은 구글 안드로이드가 완벽한 오픈 소스는 아니더라도 오픈 플랫폼을 지향하고 있기 때문이다.

마지막으로, 보다 많은 사람들이 이번 구글 안드로이드 챌린지에 도전할 수 있기를 희망하며, 더불어 모두에게 건투를 빕니다. 이 글과 관련해서 문의하고 싶거나 의견을 교환하고 싶은 분들은 언제든지 <http://cafe.naver.com/mobitoday> 에 글을 남겨 주시기 바랍니다. 화이팅~.

P.S. 출품작 아이디어 관련해서.

- 예전에 우연히 <http://www.popome.com/> 란 사이트를 운영하는 분을 만난 적이 있다. Mini MSN 을 만들어 릴리즈하고 있었던 것으로 기억난다. 이런 Mini MSN 을 구글 안드로이드에 탑재하면 어떨까?