

Mobile Web

Mobile Developer Summary

Web App

Hybrid App
Native App
Mobile Web
Mobile Web App

Native App

아이폰과 같은 모바일 기기에 직접 설치되고
운용되는 전용 애플리케이션이다.
각 프로그래밍 언어와 SDK로 개발해야 하며
기기의 고유정보에 직접 접근이 가능하다.
기기의 하드웨어, 메모리 등과 밀접하게 상호작용
할 수 있어 모바일에 고성능 애플리케이션을
개발할 수 있으며 앱스토어에 등록하고 판매할 수
있다.

Mobile Web

최신 모바일 기기는 풀 브라우징을 지원한다.
모바일 웹은 일반적인 웹 기술로 개발되고 모바일
브라우저에 의해 실행되는 웹 애플리케이션을 통
칭. 하지만 모바일 웹앱과 조금은 구분되는 개념
으로 일반적인 웹 사이트의 느낌을 모바일로 그대
로 옮긴 형태라 할 수 있다.

Mobile Web App

일반적인 웹사이트보다 모바일에 더 최적화 되고
네이티브 애플리케이션화된 형태를 한정하는 표현.
모바일 웹앱은 웹 기술만 사용해서 풀 스크린모드,
애니메이션 효과, 터치 상호작용, 비동기통신, 로
컬 저장소, 오프라인 지원, 향상된 스타일 등을 구
현하여 모바일 환경에서 네이티브 애플리케이션과
비슷한 실행환경, 사용자 경험을 제공하는 한 형태.

Hybrid App

이름에서도 유추할 수 있듯이 웹과 네이티브앱의
특징을 조합한 형태를 말한다. 일반적인 웹 기술로
도 모바일 기기의 고유 정보와 상호작용할 수 있
게 하고 앱 스토어에 올리고 판매도 할 수 있는 형
태의 애플리케이션이다. 모바일 기기의 고유 기능
을 웹 기술로 호출할 수 있는 추상화된 API가 필요
하다. PhoneGap, Titanium을 이용하여 내부는 웹
으로 외부는 앱으로 보이게끔 할 수 있다.

모바일 애플리케이션 종류에 따른 특징

하드웨어 접근성	높음	낮음(웹에서는 장치의 고유정보에 접근불가)
구현 자유도	높음(거의 모든형태의 앱구현가능)	보통(웹 표현의 한계성)
실행속도	빠름	보통
앱스토어등록/판매	가능	불가능
사용자진입(접근성)	높음(앱스토어를 통한..)	낮음(브라우저를 이용한 URL접근)
개발 환경	Object-C, Java등 각 플랫폼에 맞는 언어와 SDK사용	HTML, Javascript, CSS등의 표준기술
개발 비용	높음(각 플랫폼마다 별도 개발환경)	낮음(기존 웹 기술과 도구, 경험을 그대로활용)
멀티 플랫폼 지원 (호환성, 이식성)	낮음(각 플랫폼마다 별도 구현)	높음(하나의 소스로 다양한 플랫폼구현)
업데이트 용이성	낮음	높음(즉각 업데이트 가능)

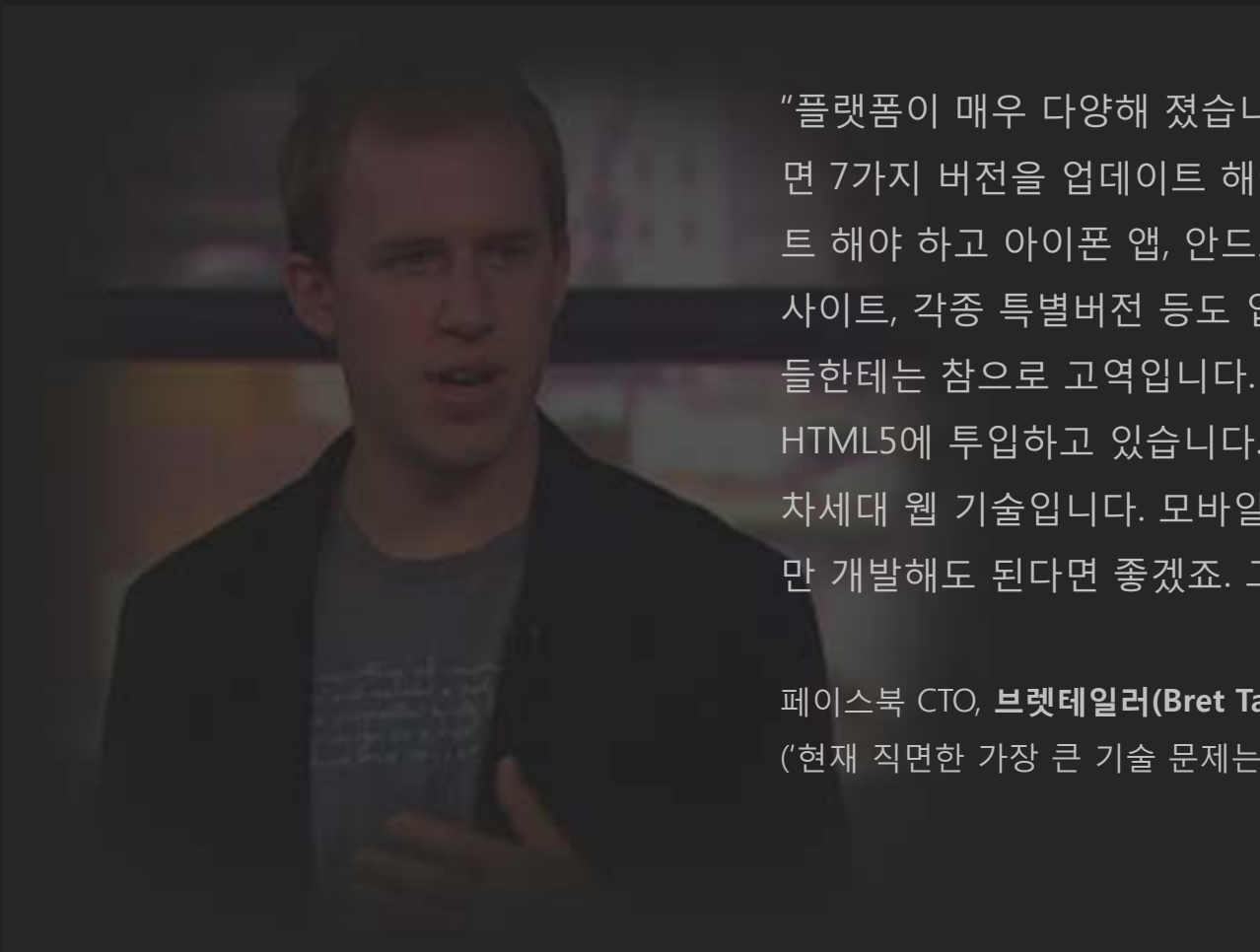
One Source Multi Platform Web App

웹으로 개발된 애플리케이션은 하나의 소스로 여러 플랫폼을 지원할 수 있다.

웹만의 장점인 즉각적인 업데이트 덕분에 개발 주기를 줄이고 사용자의 요구에 빠르게 피드백할 수 있다. 최신 웹 표준 기술인 HTML5를 접목하면 오프라인 지원, 로컬 저장소 활용, 2D그래픽 처리, 미디어 재생, 소켓 통신, 멀티 스레드 프로그래밍등의 구현이 가능해져 기존 웹의 한계를 훌쩍 뛰어넘을 수도 있다.

폰갭과 같은 API를 활용하면 앱 스토어에 올려서 사용자에게 쉽게 배포할 수 있고 유료화 모델을 적용할 수 있어 웹과 네이티브앱의 경계를 더욱 좁힐 수 있다.

결국 원소스 멀티플랫폼을 실현하면서도 네이티브의 장점을 누릴 수 있는 모바일 웹앱과 하이브리드앱은 기업의 모바일 대응에 주요한 전략이 될 것이다.



“플랫폼이 매우 다양해 졌습니다. 페이스북을 업데이트하려면 7가지 버전을 업데이트 해야 합니다. 웹사이트를 업데이트 해야 하고 아이폰 앱, 안드로이드 앱, 블랙베리 앱, 모바일 사이트, 각종 특별버전 등도 업데이트 해야 합니다. 엔지니어들한테는 참으로 고역입니다. 그래서 우리는 많은 시간을 HTML5에 투입하고 있습니다. HTML5는 모바일에 최적화된 차세대 웹 기술입니다. 모바일 기기용 페이스북 버전을 하나만 개발해도 된다면 좋겠죠. 그게 최종 목표입니다.”

페이스북 CTO, **브렛테일러(Bret Taylor)**의 말
(‘현재 직면한 가장 큰 기술 문제는 무엇인가’에 대한 답변중...)

폰갭, HTML5 그리고 일반 웹 기술의 조합은 웹을 더욱 가치있는 애플리케이션으로 만들어 줄 것이다. 결국 모바일 웹앱은 갖가지 상황에서 훌륭한 선택이 될 수 있다.



“Object-C를 배우는 데 어려움이 있었고 언어가 맥 프로그램 밖에서 약간만 사용된다는 사실에 환멸을 느꼈다.(중략) 앱스토어에 애플리케이션을 올리는 과정은 더욱 복잡했다. 한 두 주 이러한 변수와 사투를 벌인 후 내가 왜 이러한 문제들 속에서 허우적대고 있는지에 대한 의문이 생겼다.

애플리케이션이 승인을 받았다고 치자. 수백만 또는 수천만의 사람들이 애플리케이션을 내려받고 아직 요금을 지급받지 못한 상황에서 버그 보고서가 오기 시작한다. 몇 분 안에 버그를 찾아내어 수정한 후 아이튠즈에 수정한 애플리케이션을 올린다. 애플이 교정을 승인하는 동안 기다리고 또 기다린다. 그래서 며칠 또는 몇주 전에 버그가 이미 수정됐음에도 애플리케이션의 평점이 바닥으로 떨어지는 것을 앓아서 볼 수 밖에 없게 되는 것이다.

유명한 모바일 개발자이자 jQTouch관리자인 조나단 스타크의 말(‘HTML,CSS,자바스크립트를 이용한 아이폰 앱 개발’의 머리말 중)

Native(앱)와 Hybrid(웹앱)의 개발 방식의 차이.



This diagram depicts the major differences between multiple native and hybrid application development

Mobile Web Practice 요약본

W3C Mobile Web Best Practice

모바일 웹사이트 작성시 고려할 점

<http://www.w3.org/TR/mobile-bp>

하나의 웹을 위한 설계

다양한 장비를 고려하여 설계한 콘텐츠는 비용을 절감시키고, 유연성을 증대하며, 더 많은 사람들의 필요를 충족시킬 수 있다.

웹 표준의 준수

세계 곳곳의 제각기 다른 장비와 웹 브라우저의 호환을 위해서는 표준을 준수해야 한다.

유해 요소의 제거

모바일 장비의 제한된 화면크기, 키보드, 기타 기능등에 의해 발생하는 사용자의 문제를 고려하여 설계해야 한다.

장비 제한을 주의

특정 웹 기술을 사용하고자 할 때는 모바일 장비 선용이 제각기 다르다는 것을 염두에 두어야 한다.

웹 네비게이션 최적화

작은 화면과 키보드, 제한된 bandwidth에서는 단순한 네비게이션과 입력이 매우 중요하다.

Mobile Web Practice 요약본

W3C Mobile Web Best Practice

모바일 웹사이트 작성시 고려할 점

그래픽과 색상확인

이미지 색상, 스타일은 콘텐츠를 빛나게 하지만 지원되는 포맷과 저사양의 화면등에 유의해야 한다.

가볍고 간결한 웹 사이트

가벼운 웹 사이트는 시간과 비용을 절약하여 사용자를 즐겁게 한다.

네트워크 자원 절약

웹 프로토콜 기능을 활용하면 네트워크 부하와 대기 시간을 줄여 사용자 이용도를 향상 시킬수 있다.

사용자 입력 가이드

모바일 장비에서 키보드 및 각종 입력 방법은 비효율적일 수 있으며 효과적인 설계는 이러한 입력을 최소화 할 수 있다.

모바일 유저 배려

시간이 부족하고 번잡한 상황에 있는 모바일 사용자들은 간결한 정보를 얻고자 한다.

하나의 웹을 위한 설계

주제의 일관성: 다양한 장비에서 동일 URI에 접속했을 때 제공되는 정보와 기능은 일관성이 있어야 한다.

성능: 사용자에게 향상된 기능을 제공하기 위해 장비 성능을 상세히 확인해야 한다.

오작동: 다양한 장비가 각각 다르게 해석하여 발생할 수 있는 오작동을 염두에 두고 설계한다.

테스팅: 실제 장비와 에뮬레이터 상에서 테스트를 수행한다.

웹 표준의 준수

유효한 MARKUP: 표준 문법에 유효한 문서를 작성한다.

컨텐츠 포맷 지원: 장비가 지원하는 컨텐츠를 전송한다.

선호하는 컨텐츠 포맷 제공: 가능하면 장비가 선호하는 포맷으로 컨텐츠를 전송한다.

캐릭터 인코딩 지원: 장비가 지원하는 캐릭터 인코딩을 사용한 컨텐츠를 제공해야 한다.

캐릭터 인코딩 정보 활용: 장비에서 사용되는 캐릭터 인코딩을 확인한다.

스타일 시트 사용: 장비가 지원하지 않는 경우를 제외하고, 레이아웃과 형식의 설계는 스타일 시트를 사용한다.

구성: 논리적인 문서 구성을 위해 마크업 언어(markup language) 기능을 사용한다.

오류 메시지: 발생한 오류 정보를 알 수 있는 메시지를 제공하고, 그러한 오류를 피해 브라우징을 계속할 수 있도록 한다.

유해 요소의 제거

팝업창: 팝업창이나 새로운 창이 열리지 않도록 하고, 사용자에게 사전 정보없이 현재 창을 바꾸지 않는다.

테이블 속의 테이블: 테이블 내에 또 다른 테이블을 포함하지 않는다.

테이블 레이아웃: 테이블을 이용하여 레이아웃을 구성하지 않는다.

여백과 그래픽: 여백 공간에 그래픽을 사용하지 않는다.

프레임 비사용: 프레임을 사용하지 않는다.

이미지 맵(IMAGE MAPS): 이미지 맵은 장비에서 효과적으로 지원되지 않는 한 사용하지 않는다.

장비 제한을 주의

쿠키(COOKIES): 이용 가능한 쿠키에 의존해서는 안 된다.

객체나 스크립트: 임베드 되어 있는 객체나 스크립트에 의존해서는 안 된다.

테이블 지원: 장비에서 테이블을 지원하지 않는 경우 사용하지 않는다.

테이블 대체: 가능한 경우에는 테이블 형식을 대체할 수 있는 형식으로 작성한다.

스타일 시트 지원: 필요한 경우 스타일 시트 없이도 문서를 읽을 수 있도록 구성한다.

글꼴: 글꼴 관련 스타일링에는 의존하지 않는다.

색상 사용: 색상을 통해 얻을 수 있는 정보도 색상을 사용하지 않고 이용할 수 있게 한다.

웹 네비게이션(navigation)의 **최적화**

NAVBAR:페이지 상단에는 최소의 네비게이션 메뉴바를 제공한다.

네비게이션:일관된 네비게이션 메카니즘을 제공한다.

타겟 ID 링크:각 링크의 타겟을 명확히한다.

파일 포맷:이용되는 파일의 포맷이 장비에서 지원되는지 주의한다.

타겟의 접근:네비게이션 메뉴와 자주 사용하는 기능에는 키보드 숏컷(shortcut)을 지정한다.

URI:URI를 짧게하여 입력을 쉽게한다.

균형:사용자가 원하는 정보와 페이지에 제공되는 링크 수의 균형을 맞춰 원하는 정보를 쉽게 얻도록 한다.

그래픽과 색상을 확인

이미지 크기 조절:고유 크기가 있는 이미지는 서버에서 크기를 조절한다.

대용량 그래픽:장비에서 제공할 수 없는 이미지는 사용하지 않는다. 정보제공을 위해 꼭 필요한 것이 아니면 고사양의 고해상도 이미지의 사용은 피한다.

이미지 크기 설정:고유 크기가 있는 이미지는 마크업에 이미지 크기를 설정한다.

NON-TEXT 대체:텍스트가 아닌 요소(이미지, 애니메이션, 동영상 등)에 대해서는 그에 상응하는 텍스트를 제공한다.

색상 명암:전경과 배경의 색 조합은 적절한 명암 대비를 유지해야 한다.

배경 이미지 가독성:배경에 이미지를 사용할 경우 콘텐츠 내용을 볼 수 있어야 한다.

측정:마크업 언어 속성 값과 스타일 시트 속성 값에는 픽셀 측정과 절대 단위를 사용하지 않는다.

가볍고 간결한 사이트

최소화:간결하고 효율적인 마크업을 사용한다.

페이지 사이즈 제한:장비의 메모리 한계를 고려한 페이지를 설계한다.

스타일 시트 크기:스타일 시트의 크기를 최소화한다.

스크롤링:추가적인 스크롤링을 사용해야 하는 경우가 아니라면, 한 방향으로 스크롤을 하도록 제한한다.

네트워크 자원을 절약

오토 리프래쉬:사용자에게 사전에 알리거나 중지 방법을 제공하지 않는 한, 주기적인 페이지 오토 리프래시는 사용하지 않는다.

경로 변경:마크업으로 자동 경로 변경(redirection)을 사용하지 않고, 서버 설정을 통한 HTTP 3xx code 경로 변경을 사용한다.

외부 자원:외부 자원의 링크 활용을 최소화한다.

CACHING:HTTP 응답에는 캐시 정보를 제공한다.

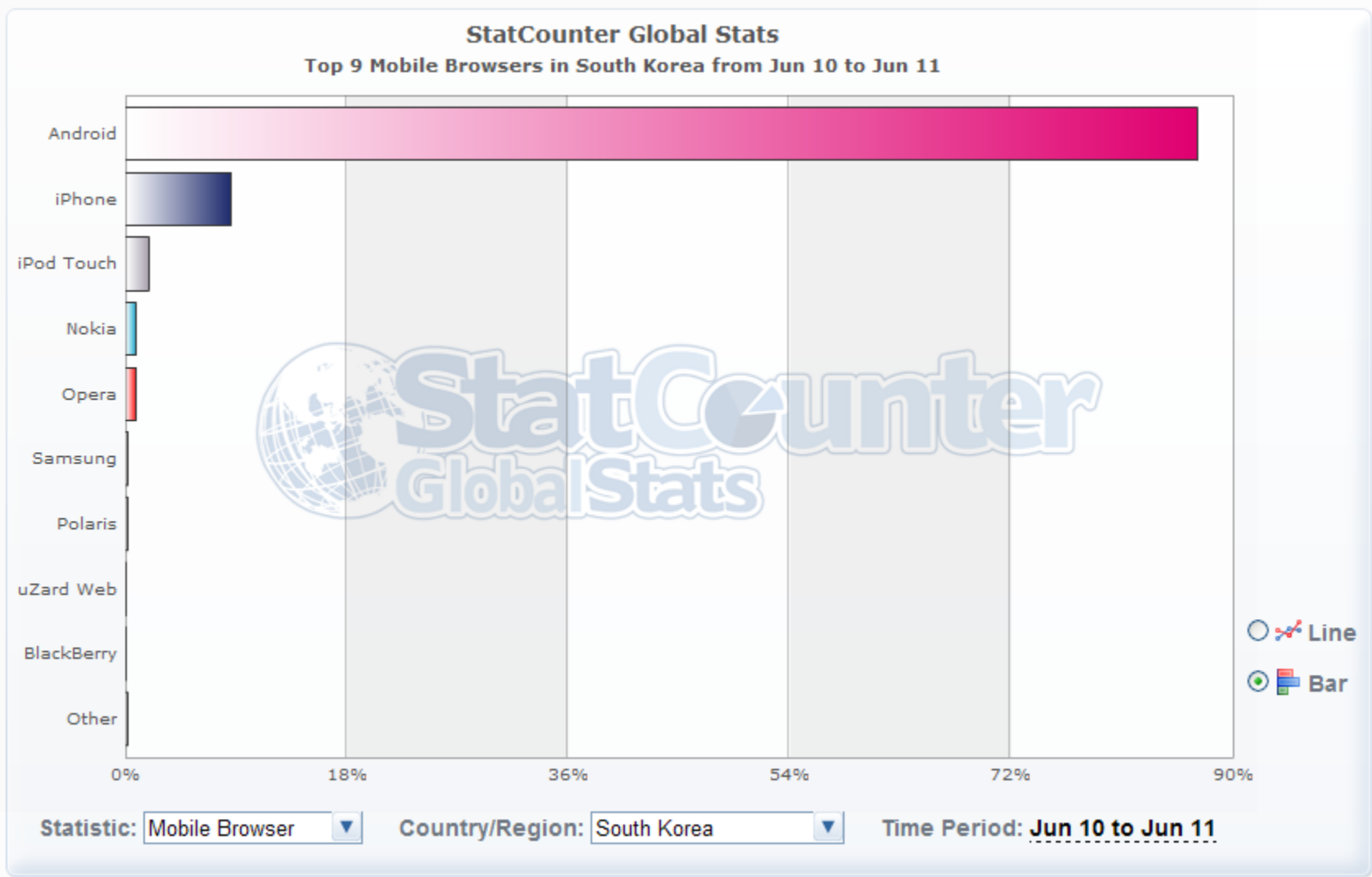
사용자 입력 가이드

- 키입력 최소화:키 입력 횟수를 최소화한다.
- 텍스트 입력 최소화:가능하면 텍스트 입력을 피한다.
- 기본값(default) 제공:가능하면 미리 선택된 기본값을 제공한다.
- 기본 입력 모드:장비가 지원하는 기본 텍스트 입력 모드, 언어, 포맷을 설정한다.
- 탭 순서:링크, 폼 컨트롤과 객체를 이용한 논리적 탭 순서를 생성한다.
- 라벨 컨트롤:모든 폼 컨트롤에 적절한 라벨을 붙이고, 명시적으로 라벨과 폼 컨트롤을 연계시킨다.
- 위치 제어:라벨과 연관된 폼 컨트롤이 서로 적절한 위치에 있도록 한다.

모바일 유저를 배려

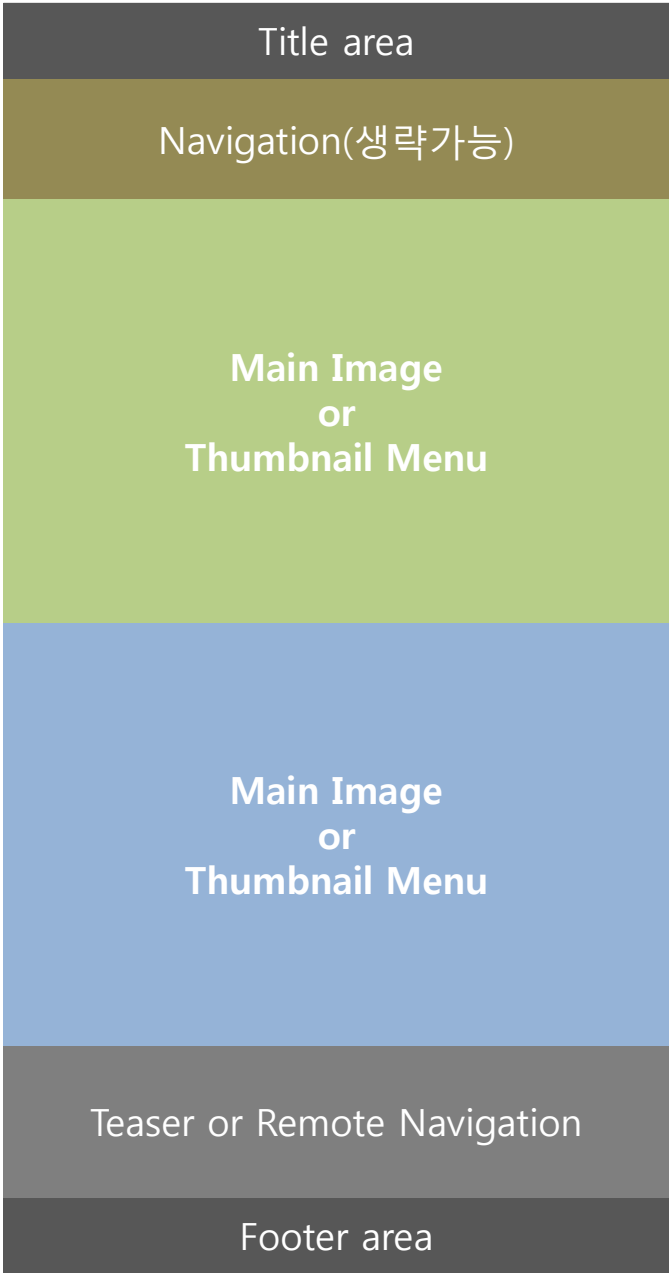
- 페이지 제목:간단 명료한 페이지 제목을 제공한다.
- 명확성:명확하고 간단한 언어를 사용한다.
- 한 눈에 의미 파악:페이지의 내용을 간단 명료하게 하여 다음 페이지의 내용까지 파악할 수 있도록 한다.
- 제한:사용자가 요청한 범위로 콘텐츠를 제한한다.
- 적절:모바일 환경 사용에 적절한 콘텐츠를 보장한다.
- 적절한 페이지 사이즈:장비의 성능과 네트워크 부하 등을 고려하여 사용하기 편리하고 유용한 크기로 페이지 사이즈를 분할한다.

한국의 모바일 브라우저 현황



for Demo

모바일 웹 레이아웃 잡기...



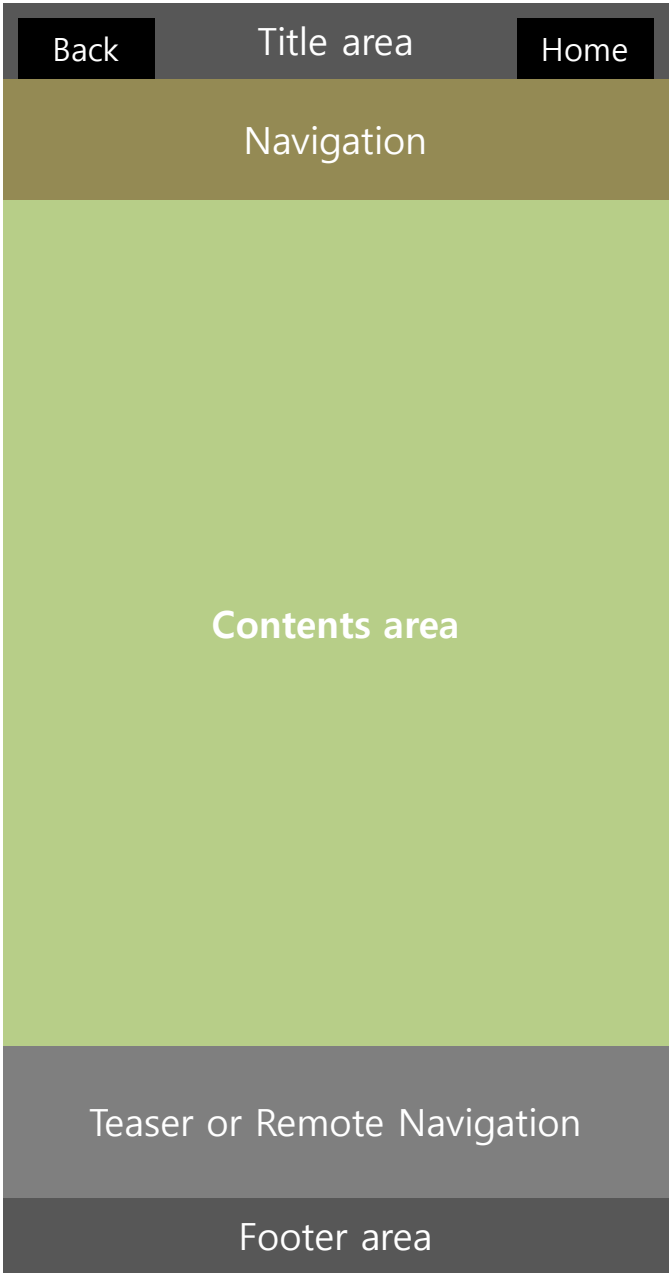
모바일의 화면은 좌우로 좁다. 따라서 길게 스크롤이 발생할 수 있다.

바로가기 메뉴처럼 썸네일로 만들어진 메뉴를 아래로 배치하면 사용자들이 보지 못하고 넘어갈 가능성이 있기때문에 적절한 메뉴갯수와 함께 이미지를 사용해야 한다.

각 디바이스에 맞는 고정형 사이트가 있고 어떤 디바이스에서 보던 잘 맞는 가변형 사이트가 있다.

서브페이지에서는 반드시 뒤로가기 버튼과 첫화면으로 돌아가는 버튼이 존재해야 한다.

가로보기와 세로보기 테스트를 해도 화면벗어남이나 깨짐이 없어야 한다.

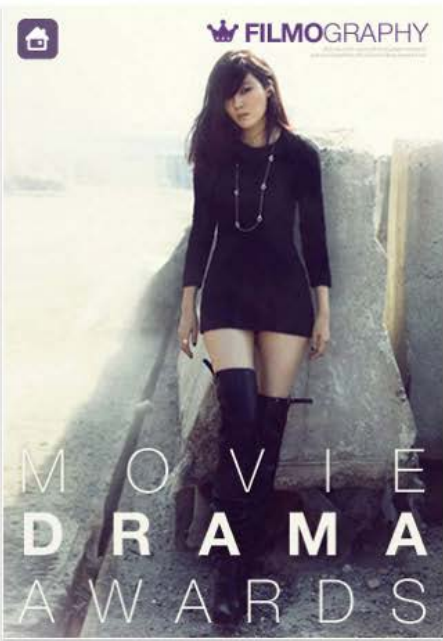


Mobile Site

다양한 느낌으로 만들어진 모바일 페이지를 벤치마킹.

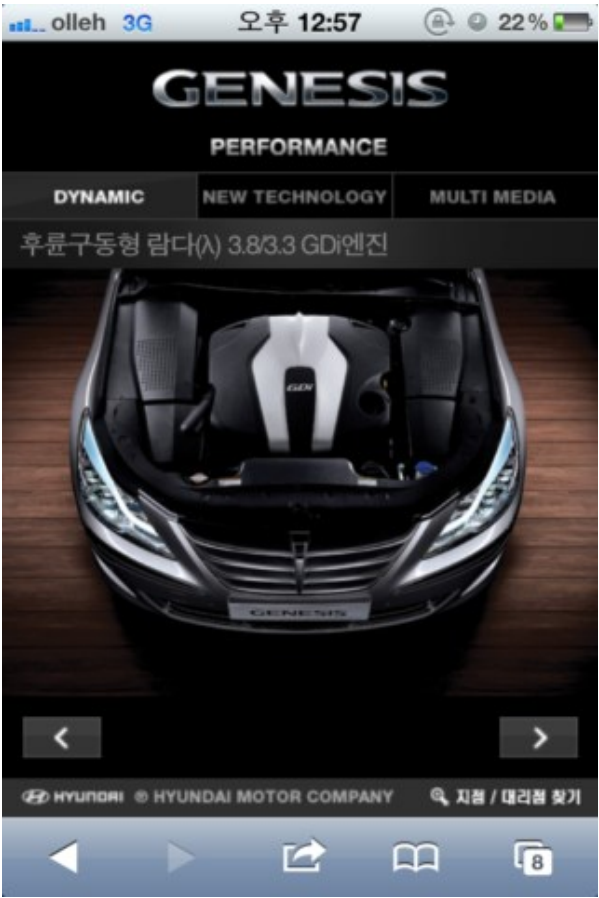
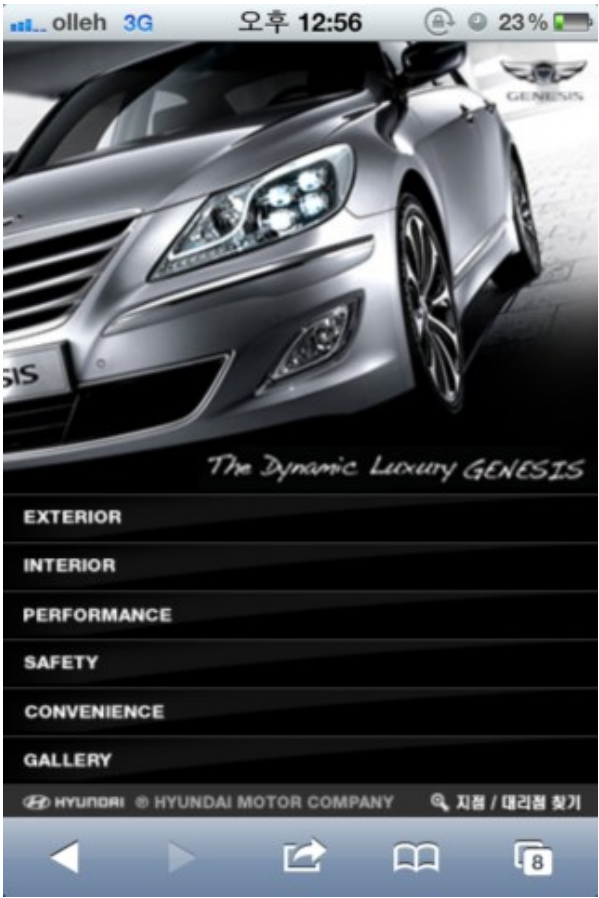
김하늘의 모바일 앱

비주얼이 강하게 표현된 모바일 페이지. 앱으로 만들어지긴 했지만 내부는 웹으로 만들어졌다. 모바일 웹앱류로 보인다. 사진을 활용하여 매우 화려한 느낌으로 제작되었다. 메인페이지가 독특하게 잡지의 표지처럼 만든게 눈길을 끈다.



제네시스 프로모션 모바일 웹

프로모션의 성격이 강한 모바일웹사이트.
따라서 강렬한 인상을위해 이미지가 많이 사용 되었다. 비주얼을 강조하는 프로모션의 사이트를 만들때
참고할 만 하다.



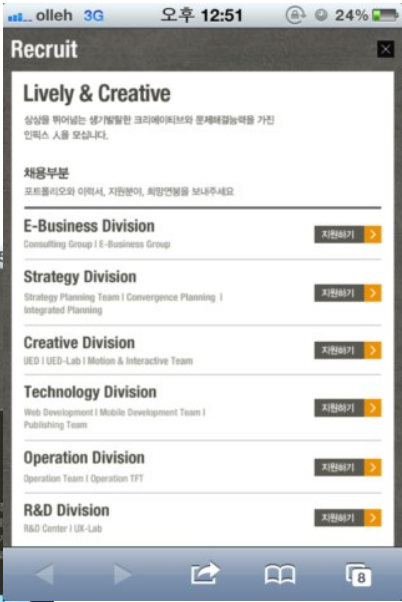
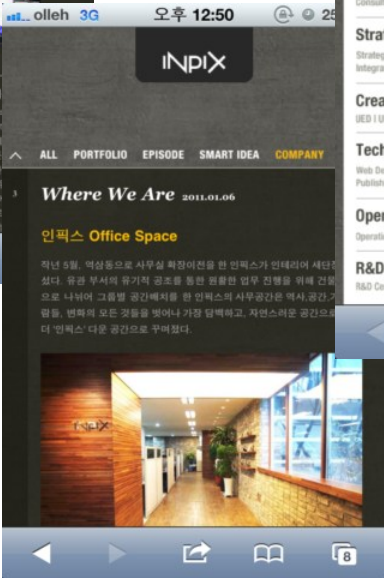
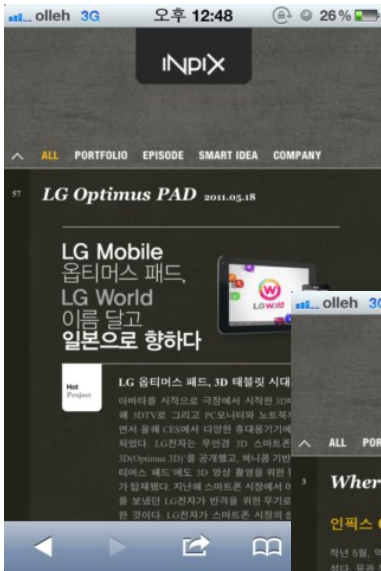
한스 델리의 모바일웹

레이아웃은 단순하게 구성하고 콘텐츠는 요리 사이트답게 사진들이 주로 많은 공간을 차지했다. 사진들속의 화려한 요리들을 부각시키기 위해 레이아웃은 좀더 단순한 구조로 만들었다. 콘텐츠를 강조하고 싶은 사이트를 제작시 참고할 만한 사이트.



인픽스의 모바일 웹

허브 느낌의 페이지. 콘텐츠로의 연결에 중점을 둔 메인에 눈에 띈다.
단순한 구조로 콘텐츠가 더 강조되는 비주얼로 제작되었다. 무겁고 중후한 느낌으로 고급스러운 느낌을 살렸다. 웹진형태의 모바일 웹이다.



호텔 엔조이

모바일 웹 초창기부터 많이 보이던 포맷으로 메인에 바로가기 버튼과 각각의 기능들을 하는 페이지들을 중심으로 구성. 기능에 충실한 디자인으로 업데이트가 용이하고 관리하기가 쉽다. 그러나 독창성이 없어보이는 단점이 있기때문에 금방질린다.



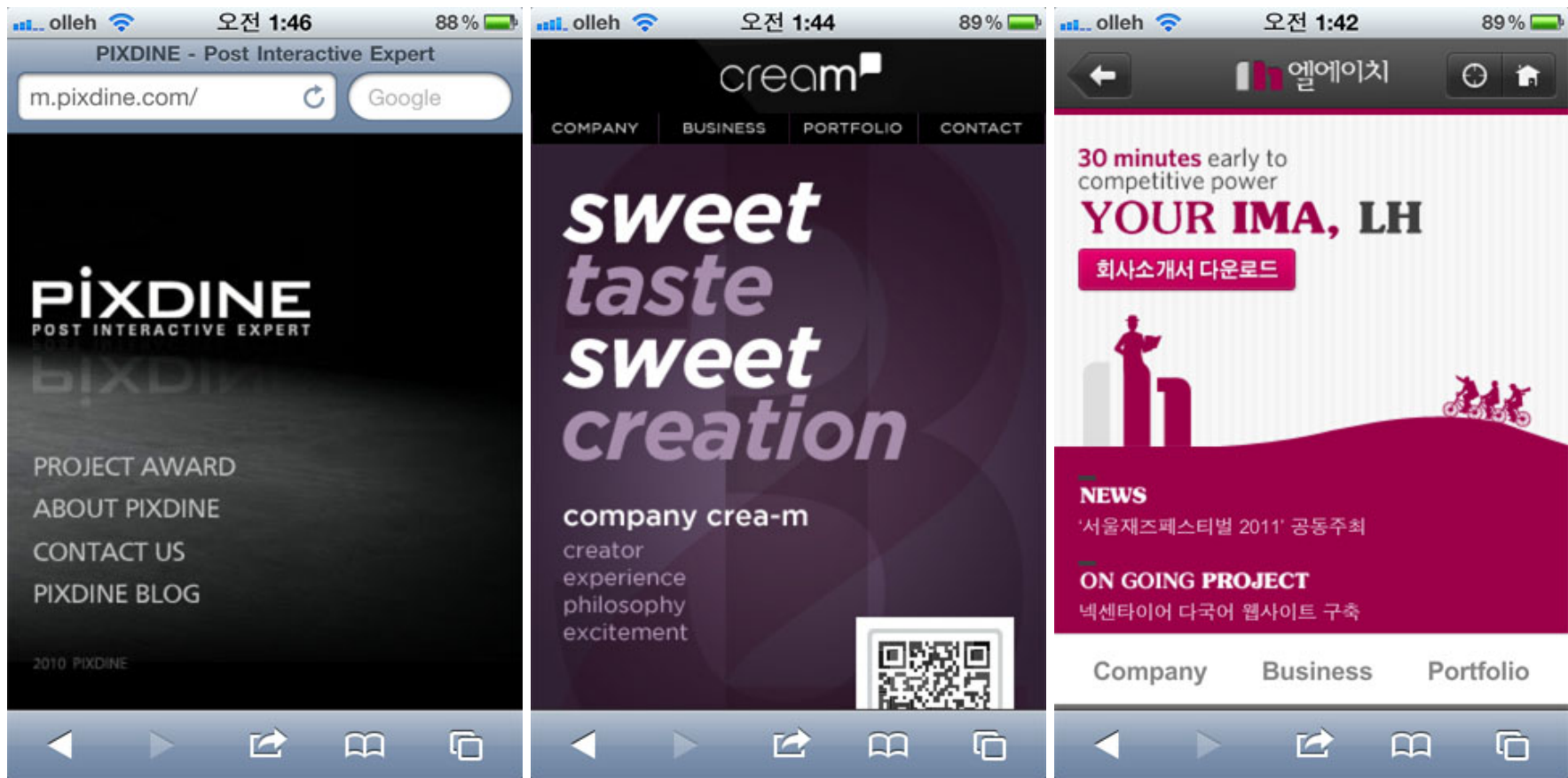
포털 페이지

대형 포털은 다루는 내용이 많은 편이라 기능성에 있어서 참고 할 만한 부분이 많다.
눈에 보이는 부분 뿐 아니라 내부 코딩과 같은 기능성에 있어서도 참고 할 만하다.
검색창을 눈에 띄는곳에 두고 뉴스를 메인으로 노출하고 있는점이 공통점이다.



회사 소개형

회사 소개형 홈페이지들은 요란한 이미지를 전면에 두기 보다는 회사의 슬로건이나 단순 메뉴의 나열등을 하는 경우가 많다. 요란하지 않지만 최대한 회사의 이미지를 전달하는데 노력한다.



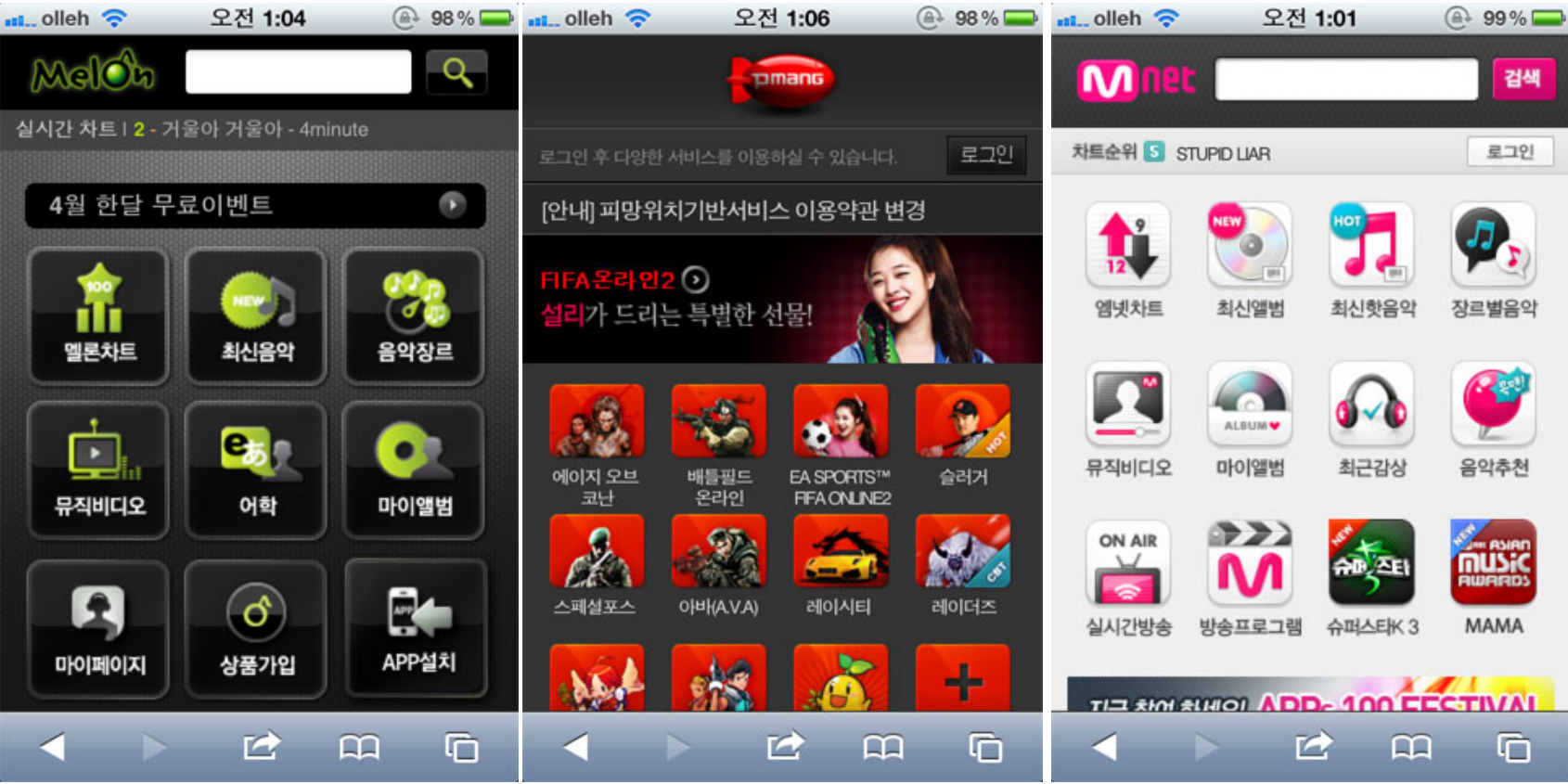
프로모션 형

무엇보다 화려하다. 짧은기간의 매출증진을 목적으로 하거나 많은 사람들의 이목을 끌어야 하는 페이지들이기 때문에 임팩트가 강하게 디자인 되었다는게 특징이다.



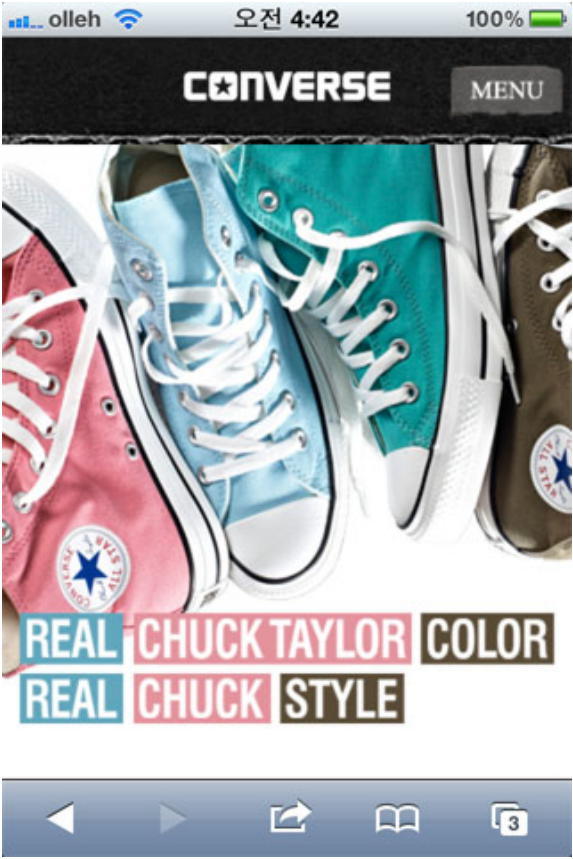
메뉴 나열 형

내부 콘텐츠가 많을 경우 탭형의 메뉴로는 콘텐츠로의 바로접근이 불가능하다.
따라서 많은 콘텐츠를 소화하기 위하여 버튼형태의 메뉴로 메인을 제작하는 경우가 많다.
가로로 넓게 봤을때도 깨지지 않고 나열시키는게 포인트. 버튼의 디자인을 잘못하면 촌스러워 질 수 있다.



이미지 강조형

사진 또는 사진처럼 구현된 3D이미지를 통해 비주얼을 강조한 사이트들.
개인 페이지도 본인의 프로필 사진등을 활용하여 충분히 화려하게 꾸민다면 쉽게 사이트를 구성할 수 있다.
제품을 강조하는 사이트로 프로모션의 성격도 강하다. 하지만 단기가 아닌 장기 프로모션의 성격이라 아주 화려하거나 들뜨는 느낌은 배제한다.



Link

- [맘스쿨 http://m.momschool.co.kr/](http://m.momschool.co.kr/)
- [플레이포럼 http://m.playforum.net/](http://m.playforum.net/)
- [다음아고라 http://m.agora.daum.net/](http://m.agora.daum.net/)
- [다음모바일 http://m.daum.net/](http://m.daum.net/)
- [네이버 모바일 http://m.naver.com/](http://m.naver.com/)
- [야후모바일 http://kr.m.yahoo.com/](http://kr.m.yahoo.com/)
- [파란 모바일 http://m.paran.com/mini/](http://m.paran.com/mini/)
- [네이트 모바일 http://m.nate.com/](http://m.nate.com/)
- [프랑스 기차여행 http://m.topfrenchcities.co.kr/](http://m.topfrenchcities.co.kr/)
- [국립박물관 문화재단 http://cfnmk.ipamad.com/](http://cfnmk.ipamad.com/)
- [거제 방문의 해 http://mvisit.geoje.go.kr/html/](http://mvisit.geoje.go.kr/html/)
- [국립민속박물관 http://m.nfm.go.kr/](http://m.nfm.go.kr/)
- [한화증권 http://m.koreastock.co.kr/mobile/](http://m.koreastock.co.kr/mobile/)
- [이글루스 http://www.egloos.com/m/](http://www.egloos.com/m/)
- [스윗트유 http://m.sweetu.co.kr/](http://m.sweetu.co.kr/)
- [토그미니 http://m.tog.kr/](http://m.tog.kr/)
- [파란블로그 http://m.blog.paran.com/](http://m.blog.paran.com/)
- [슈퍼스타K http://m.superstark.tving.com/ssk/common/mobile/korea/index.do](http://m.superstark.tving.com/ssk/common/mobile/korea/index.do)
- [현대백화점 http://m.ehyundai.com/mobile/index.jsp](http://m.ehyundai.com/mobile/index.jsp)
- [알라딘 http://m.aladin.co.kr/m/main.aspx?Partner=maladdin](http://m.aladin.co.kr/m/main.aspx?Partner=maladdin)
- [텐바이텐 http://m.10x10.co.kr/](http://m.10x10.co.kr/)

Mobile Device

다양한 모바일 기기의 환경에 맞춰 페이지를 개발하여야 한다.

iPad Smart Phone
iPhone4

기기별 해상도

Screen resolution

Mobile Device

320 x 480 : 아이폰3G(S), 안드로이드, 레전드

480 x 800 : 넥서스원, 넥서스S, 디자인어(HD), HD2, 갤럭시S/A,
시리우스, 베가, 옵티머스Q, 올레패드

480 x 854 : 모토로이, 모토쿼티, 모토슬램

640 x 960 : 아이폰4

600 x 1024 : 갤럭시탭

768 x 1024 : 아이패드



아이폰 3G의 경우 실제 가로 해상도는 **320px**이지만 아이폰의 사파리 웹 브라우저는 기본적으로 **980px**인 것처럼 작동합니다.

해상도의 진화

Screen resolution

Mobile Device



[보통의 모바일 웹]



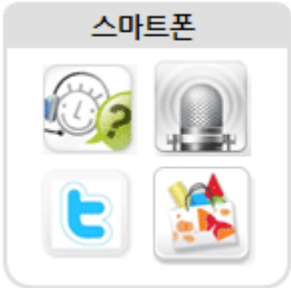
[해상도가 큰 스마트 기기에 맞춘 모바일 웹]

! 해상도가 큰 스마트 기기에 맞춰 디자인 하기 위해 기존 모바일 웹 제작 방식에서 약 4배정도 커진 해상도로 제작되고 있다.

해상도의 진화

Screen resolution

Mobile Device



스마트폰

무한돌봄사업은 실제 위기상황을 겪고 있지만 정부의 지원기준에 해당되지 않아 지원을 받지 못함으로써 어려움을 겪고 있는 '위기가정'을 경기도가 별도의 예산을 마련하여 생활안정을 돕는 사업입니다.

국세기본법 제5조(기한의통례) 규정에 따라 신고·납부 기한일이 공휴일·토요일 또는 근로자의 날에 해당하는 때에는 공휴일·토요일 또는 근로자의 날의 다음 날을 기한으로 합니다.

해상도 큰 스마트 기기 (아이패드, 갤럭시 탭 등)

+ 이용시간 : 평일 07:00 ~ 22:00
토.일.공휴일 및 이용시간 외에는 유실물센터가 있는 역의 역무실로 문의해 주세요

! 해상도가 큰 스마트 기기에 맞춰 디자인 하기 위해 기존 모바일 웹 제작 방식에서 약 4배정도 커진 해상도로 제작되고 있다.

해상도의 진화

Screen resolution

Mobile Device



! 해상도가 큰 스마트 기기에 맞춰 디자인 하기 위해 기존 모바일 웹 제작 방식에서 약 4배정도 커진 해상도로 제작되고 있다.

Viewport

meta tag

Mobile Device

모바일 기기에서 Media Query가 예상한대로 작동하기 위해서는 웹 브라우저의 가상 해상도를 물리적 해상도와 일치시킬 필요가 있습니다. 모바일 웹브라우저들은 웹브라우저의 가상 해상도를 조절할 수 있는 viewport메타 태그를 지원합니다.

```
<meta name="viewport"
```

```
content="  
width=[가상해상도],  
initial-scale=[초기 확대 축소 비율],  
maximum-scale=[최대 확대 비율],  
minimum-scale=[최소 축소 비율],  
user-scalable=[사용자에 의한 확대 축소 허용 여부]  
target-densitydpi=[기각의 실제 해상도]  
"
```

```
/>
```

```
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, minimum-scale=1, user-scalable=yes , target-densitydpi=medium-dpi">
```

Viewport

meta tag

Mobile Device

width : 넓이 – device-width | N px (200~10000 px, default 980 px)

height : 높이 – device-height | N px (223~10000 px)

initial-scale : 초기 확대/축소 배율

minimum-scale : 최소 축소 배율 – N (0~10, default 0.25)

maximum-scale : 최대 확대 배율 – N (0~10, default 1.6)

user-scalable : 확대/축소 가능 여부 – yes | no (default yes)

target-densitydpi : device-dpi, high-dpi(480x800), medium-dpi(기본값 320x480), low-dpi

※ HVGA(MDPI) 320x480 / WVGA(HDPI) 480x800

initial-scale은 중요하다.

480의 해상도를 갖고 있는 스마트폰을 위해선 디자인을 480에 맞춰서 해야 하는데.

320의 해상도를 갖는 스마트폰으로 보면 문제가 생길수도 있다(레이아웃이 깨지는등의)

따라서 정확하게 값을 지정해줘야 한다.

480디자인 → 320 뷰 : initial-scale=0.67, target-densitydpi=medium-dpi (두값 모두 320을 위한 값으로 지정한다.)

480디자인 → 480 뷰 : initial-scale=1, target-densitydpi=high-dpi (두값 모두 480을 위한 값으로 지정한다.)

다른 방법

480으로 1.5배 크게 디자인을 한뒤 코딩할때 1.5배를 줄여서 CSS를 정의하여 연결한다.

Making Mobile Web!

practice

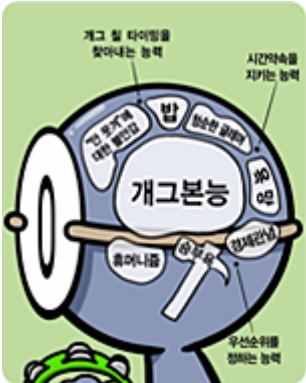
Mobile Device

혈액형에 관한 간단한 고찰

1. 세로보기를 중심으로 제작
2. psd 파일을 슬라이스 하기(320px)
3. 뷰포트를 삽입하여 모바일에서 문제없이 보도록 만들기
4. 이미지는 백그라운드가 아닌 이미지로 삽입
5. 표준으로 코딩하고 HTML5, CSS3 요소를 사용할 수 있는 부분은 사용해 만들기
6. 아이폰3, 아이폰4, 갤럭시S, 아이패드, 갤럭시탭등 다양한 기기로 보기.
7. 문제점 리포트 하기 - 조장발표



ABO
혈액형에 관한 고찰
A SIMPLE THINKING ABOUT BLOODTYPE



고정형 모바일 디자인

View choice

Mobile Device

뷰포트의 변화

```
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, minimum-scale=1, user-scalable=yes, target-densitydpi=medium-dpi">
```



```
<meta name="viewport" content="width=device-width, user-scalable=yes, target-densitydpi=device-dpi">
```

이미지가 중심이된 페이지인 경우 고정된 이미지 사용이 많을 수 밖에 없다.

문제는! 세로보기용으로 만들지만 **사용자는 가로보기 세로보기 구분없이 본다는 것이다.**

그럴 경우 scale값들을 고정해 놓으면 가로보기도 자동으로 폭이 정렬되지 않는다.

따라서 고정형일 경우는 scale값들을 삭제해 기본적인 사이즈 조정을 각 디바이스에 맡기는 방식으로 처리한다.

큰 사이즈의 이미지로 디자인 된 페이지를 하위 사이즈에 맞추는 때 scale값들을 0.5 및 0.67등으로 맞추는 방법을 사용하지만 이 역시도 **가변형에선 사용할 수 있지만 고정형에서 그런 값들을 입력할 경우 가로보기도 원하지 않던 하얀여백등이 나타나는걸 볼 수 있다.**

※ 문제는 강제 사이즈 조절시 이미지가 깨지는 경우가 발생하기 때문에 될 수 있으면 가장 큰 해상도를 기준으로 이미지 작업을 해야 한다. 줄었을때가 늘렸을때보다는 덜 깨지기 때문이다. (물론 작은 글씨들이 포함된다면 문제가 된다.)

Making Mobile Web!

practice

Mobile Device

혈액형에 관한 간단한 고찰2

1. 세로보기를 중심으로 제작
2. psd 파일을 슬라이스 하기(640px)
3. 뷰포트를 삽입하여 모바일에서 문제없이 보도록 만들기
4. 이미지는 백그라운드가 아닌 이미지로 삽입
5. 표준으로 코딩하고 HTML5, CSS3 요소를 사용할 수 있는 부분은 사용해 만들기
6. 아이폰3, 아이폰4, 갤럭시S, 아이패드, 갤럭시탭등 다양한 기기로 보기.
7. 320으로 만들었을때와의 차이점 찾기
8. 문제점 리포트 하기 - 조장발표



가변형 모바일 디자인

View choice

Mobile Device

뷰포트의 변화

```
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, minimum-scale=1, user-scalable=yes, target-densitydpi="medium-dpi">
```



```
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, minimum-scale=1, user-scalable=yes, target-densitydpi="medium-dpi">
```

가변형 디자인에서의 뷰포트는 크게 문제되지 않는다. 대신 달라지는 디바이스 사이즈에 맞춰서 기본 페이지의 정렬 값을 변경해줘야 한다. 그렇지 않을 경우 정렬값에 문제가 생길 수 있다.

뷰포트의 문제가 아니라 각각의 경우에 맞춰 미디어쿼리를 연결하고 다른 CSS값을 적용하면 된다.

하나의 모바일 페이지로 각각의 디바이스에서 깨지지 않는 정상적인 화면을 위해선 CSS를 각각의 디바이스에 맞게 정렬하고(마진값이나 폰트사이즈등..) 그 CSS를 미디어 쿼리로 연결하는 방법을 사용한다.

미디어 쿼리는 각각의 디바이스 사이즈에 맞춰 서로 다른 CSS값을 적용 할 수 있는 방법을 말한다.

각각의 사이즈값에 따라서 지정할 수 있고 또한 가로보기 세로보기시의 값도 따로 지정할 수 있다.

for iPhone

아이폰을 위한 각종 팁과 기술들...

iPhone Splash Screen

for iphone

Mobile Device

스플래시 스크린이란 웹 페이지를 홈 화면에 추가한 후 실행할때 잠깐동안 로고나 이미지를 보여주고 사라지는 화면입니다. 스프래시 스크린을 추가하는 방법은 간단합니다.

```
<link rel="apple-touch-startup-image" href="startup.png">
```

해당 이미지의 사이즈가 **320 x 460 px** 로 맞춰야 하며,

전체화면 모드에서 설명한 apple-mobile-web-app-capable Meta 태그를 적용해야 스프래시 스크린도 적용됩니다.

iPhone 폰트사이즈 고정하기

for iphone

Mobile Device

모바일 디바이스의 경우 화면을 가로/세로로 전환할 수 있습니다. 웹킷 기반의 웹 브라우저는 뷰포트의 크기에 따라 폰트 사이즈가 변경되는데 보통 그러한 기능이 화면을 깨트리는 경우가 발생하여 CSS를 이용해 폰트 사이즈를 고정시키기도 합니다.

폰트 사이즈를 고정시키려면 고정시킬 곳에 다음 CSS를 추가하면 됩니다.

`-webkit-text-size-adjust:none;`

`-webkit-text-size-adjust – auto | none | N% (default auto)`

iPhone 상단 바 컬러 설정

for iphone

Mobile Device

아이폰 상단의 상태 Bar의 경우 3가지 테마를 Meta 태그로 설정할 수 있습니다. 특별한 기능이 있는건 아니지만 디자인을 좀 더 통일성 있게 만들수 있다는 점에서 원하는 색상으로 적용하시면 됩니다.

// 기본 밝은 계열

```
<meta name="apple-mobile-web-app-status-bar-style" content="default">
```

// 검정색

```
<meta name="apple-mobile-web-app-status-bar-style" content="black">
```

// 반투명 검정색

```
<meta name="apple-mobile-web-app-status-bar-style" content="black-translucent">
```

iPhone4 레티나 디스플레이

for iphone

Mobile Device

```
.myImage {  
  height: 40px;  
  width: 100px;  
  -webkit-background-size: 100px 40px;  
  background: url("images/myImage.jpg");  
}  
  
@media screen and (-webkit-device-pixel-ratio: 2) {  
  .myImage {  
    background: url("images/myImage@2x.jpg");  
  }  
}
```

고해상도의 아이폰4가 출시하면서 우리는 고해상도의 이미지를 제공하여
iphone4 사용자의 만족도를 극대화할수 있습니다.
320x480의 2배인 640x960의 레티나 디스플레이

<http://aralbalkan.com/demos/high-dpi-images/> iphone4로 봤을때 이미지의 흐림 현상해결
결론은 이미지를 두배로 만들고 코딩시 320에 맞춰 코딩(CSS)했다. 1.5는 480기준으로 봤을때의 결과다.

iPhone 홈 화면 아이콘 등록

for iphone

Mobile Device

아이폰의 홈화면 등록하기에 사용될 아이콘을 저장한다.

```
// 반사광 효과를 주는 경우
<link rel="apple-touch-icon" href="home-icon-name.png">
// 반사광 효과가 없는 경우
<link rel="apple-touch-icon-precomposed" href="home-icon-name.png">
```

홈화면에 등록될 아이콘의 크기 **가로 57px X 세로 57px – PNG file**

아이폰3g, 아이폰3gs

```
<link rel="apple-touch-icon-precomposed" media="screen and (resolution: 163dpi)" href="/iOS-57.png" />
```

아이패드

```
<link rel="apple-touch-icon-precomposed" media="screen and (resolution: 132dpi)" href="/iOS-72.png" />
```

아이폰4

```
<link rel="apple-touch-icon-precomposed" media="screen and (resolution: 326dpi)" href="/iOS-114.png" />
```

아이폰을 위한 설정이지만 일반 안드로이드 폰에서도 홈화면 아이콘을 지원하고 있다.

iPhone 홈 화면 전체 화면 모드

for iphone

Mobile Device

웹 페이지를 홈 화면으로 추가하면 특정 Meta 태그를 통해 웹 브라우저의 상단의 주소 표시줄과 하단의 버튼 Bar를 없앨 수 있습니다. 단 홈 화면에 추가한 후 홈 화면의 아이콘을 통해 접속 했을 경우에만 적용이 됩니다.

```
<meta name="apple-mobile-web-app-capable" content="yes">
```

자바스크립트의 window.navigator.standalone의 결과값(true/false)으로 전체화면 모드 유무를 알 수 있습니다.

추가로 자바 스크립트로 웹 페이지에 접속시 자동으로 스크롤을 상단의 주소 표시줄만큼 내려 상단의 주소표시줄이 사라진 것 처럼 만들수 있습니다.

```
<script type="text/javascript">
  window.addEventListener("load",function(){
    setTimeout(scrollTo,0,0,1);
  },false);
</script>
```


회전에 따른 다른 CSS적용하기

for iphone

Mobile Device

```
<script type="text/javascript">
function orient()
{
  switch(window.orientation){
    case 0: document.getElementById("orient_css").href = "css/iphone_portrait.css";
      break;
    case -90: document.getElementById("orient_css").href = "css/iphone_landscape.css";
      break;
    case 90: document.getElementById("orient_css").href = "css/iphone_landscape.css";
      break;
  }
}
window.onload = orient();
</script>
<body onorientationchange="orient();" >
</body>
```

기본 어플 호출하기

for iphone

Mobile Device

- 전화걸기 : `블루웹 고객센터`
- 문자보내기 : `문자보내기`

아이폰 사파리에서 전화번호 링크를 마음대로 하는 경우가 있는데 이럴 경우
`<meta name = "format-detection" content = "telephone=no"/>`
하면 자동 링크가 되지 않는다.

for Android

안드로이드 폰을 위한 팁과 기술들...

안드로이드폰의 주소창 제거

for Android

Mobile Device

네이버나 다음의 모바일웹사이트에 가보시면 주소창이 사라지는걸 확인할수 있습니다. 아이폰과 안드로이드를 테스트한결과 둘다 사라집니다. 하지만 개인이 지식없이 만든 페이지의 대부분은 아이폰에서는 어떨지몰라도 안드로이드쪽에서는 100% 주소창이 남아있는 결과를 초래합니다. 결국 주소창만큼 페이지가 아래로 밀리게되고 보여지는 부분은 9/10이 되겠죠. 손으로 올려보면 주소창이 사라지지만 그건 근본적인 해결방안이 아닙니다. 위 사항은 자바스크립트문으로 간단하게 처리가 됩니다.

```
<script type="text/javascript" language = "javascript">
window.addEventListener('load', function() {
    setTimeout(scrollTo, 0, 0, 1);
}, false);
</script>
```

안드로이드폰의 폰트

for Android

Mobile Device

안드로이드의 모든 텍스트는 드로이드(Droid)폰트로 렌더링 됩니다. 드로이드 폰트 패밀리에는 특별히 모바일 디바이스를 위해 만들어졌습니다. 우수한 문자 셋과 드로이드 Sans, Sans Nono, Serif라는 세가지 특별한 폰트를 지원합니다.

Mobile. Device. Media Query viewport

다양한 환경에 맞게 제작된 CSS를 불러와 각 디바이스에 맞게 보여준다.

속성으로 구분

Media Query

Mobile Device

스마트폰 (가로/세로):

```
@media only screen and (min-device-width : 320px) and (max-device-width : 480px) {  
/* Styles */  
}
```

스마트폰 (가로):

```
@media only screen and (min-width : 321px) {  
/* Styles */  
}
```

스마트폰 (세로):

```
@media only screen and (max-width : 320px) {  
/* Styles */  
}
```

속성으로 구분

Media Query

Mobile Device

iPad (가로/세로):

```
@media only screen and  
(min-device-width : 768px) and (max-device-width : 1024px) {  
/* Styles */  
}
```

iPad (가로):

```
@media only screen and  
(min-device-width : 768px) and (max-device-width : 1024px) and (orientation : landscape) {  
/* Styles */  
}
```

가로보기인 경우의 값

iPad (세로):

```
@media only screen and  
(min-device-width : 768px) and (max-device-width : 1024px) and (orientation : portrait) {  
/* Styles */  
}
```

세로보기인 경우의 값

속성으로 구분

Media Query

Mobile Device

데스크탑 브라우저 (가로):

```
@media only screen and (min-width : 1224px) {  
/* Styles */  
}
```

큰 모니터:

```
@media only screen and (min-width : 1824px) {  
/* Styles */  
}
```

iPhone4와 같은 높은 해상도:

```
@media only screen and (-webkit-min-device-pixel-ratio : 1.5)  
{  
/* Styles */  
}
```

320으로 작업한 이미지의 경우 : 2.0
480으로 작업한 이미지의 경우 : 1.5

<http://aralbalkan.com/demos/high-dpi-images/> iphone4로 봤을때 이미지의 흐림 현상해결
결론은 이미지를 두배로 만들고 코딩시 320에 맞춰 코딩(CSS)했다.

쉽게 말해 가장 큰 해상도에 맞춰 이미지 작업을 한후 줄여서 보는게 최선이다. 그러나 배경이 아니라 일반 이미지 삽입시에는 CSS로 컨트롤 할 수 없다.

파일로 구분

Media Query

Mobile Device

스마트폰 (가로/세로):

```
<link rel="stylesheet" href="smartphone.css"
      media="only screen and (min-device-width : 320px) and (max-device-width : 480px)">
```

스마트폰 (가로):

```
<link rel="stylesheet" href="smartphone-landscape.css"
      media="only screen and (min-width : 321px)">
```

스마트폰 (세로):

```
<link rel="stylesheet" href="smartphone-portrait.css"
      media="only screen and (max-width : 320px)">
```

iPad (가로/세로):

```
<link rel="stylesheet" href="ipad.css"
      media="only screen and (min-device-width : 768px) and (max-device-width : 1024px)">
```

iPad (가로):

```
<link rel="stylesheet" href="ipad-landscape.css"
      media="only screen and (min-device-width : 768px) and (max-device-width : 1024px) and
      (orientation : landscape)">
```

파일로 구분

Media Query

Mobile Device

iPad (세로):

```
<link rel="stylesheet" href="ipad-portrait.css"
  media="only screen and (min-device-width : 768px) and (max-device-width : 1024px) and
(orientation : portrait)">
```

데스크탑 브라우저 (가로):

```
<link rel="stylesheet" href="widescreen.css"
  media="only screen and (min-width : 1224px)">
```

큰 모니터:

```
<link rel="stylesheet" href="widescreen.css"
  media="only screen and (min-width : 1824px)">
```

iPhone4와 같은 높은 해상도:

```
<link rel="stylesheet" href="iphone4.css"
  media="only screen and (-webkit-min-device-pixel-ratio : 1.5), only screen and (min-device-pixel-
ratio : 1.5)">
```

Making Mobile Web!

practice

Mobile Device

혈액형에 관한 간단한 고찰3

1. 세로보기를 중심으로 제작
2. psd 파일을 슬라이스 하기
3. 뷰포트를 삽입하여 모바일에서 문제없이 보도록 만들기
4. 이미지는 백그라운드가 아닌 이미지로 삽입
5. 표준으로 코딩하고 HTML5, CSS3 요소를 사용할 수 있는 부분은 사용해 만들기
6. 가로보기 시에도 깨지지 않고 한쪽으로 쏠리거나 하지 않게 코딩.
7. 미디어쿼리를 사용해서 가로보기시에 조금 덜 어색하게 만들기(아님 완벽하게 해도 됨)
8. 아이폰3, 아이폰4, 갤럭시S, 아이패드, 갤럭시탭등 다양한 기기로 보기.
9. 문제점 리포트 하기 - 조장발표



타이틀

썸네일
버튼

섬네일
버튼

썸네일
버튼

썸네일
버튼

썸네일
버튼

섬네일
버튼

썸네일
버튼

썸네일
버튼

카툰이미지들 삽입

Making Mobile Web!

practice

Mobile Device

일반 텍스트의 변화 보기

1. 세로보기를 중심으로 제작
2. 상단에 메뉴를 만들고 그 아래 일반 텍스트 넣기
3. 뷰포트를 삽입하여 모바일에서 문제없이 보도록 만들기
4. 표준으로 코딩하고 HTML5, CSS3 요소를 사용할 수 있는 부분은 사용해 만들기
5. 가로보기 시에도 깨지지 않고 한쪽으로 쏠리거나 하지 않게 코딩.
6. 텍스트 사이즈를 고정하지 않은 페이지와 고정한 페이지 비교하기
7. 아이폰3, 아이폰4, 갤럭시S, 아이패드, 갤럭시탭등 다양한 기기로 보기.
8. 문제점 리포트 하기

타이틀			
메뉴 1	메뉴 2	메뉴 3	메뉴 4
텍스트 내용 삽입			
푸터			

for Information Architecture

사용자들이 힘들지 않게 사용할 수 있도록 제작한다.

터치 오류를 줄여라

user interface

Mobile Device

사용자가 터치하는 영역은 **44px*44px**를 기준으로 하세요.

손가락으로 터치하는 면적이 최소 이 정도는 확보되어야 오터치를 줄일 수 있습니다.
너무 작은 경우 링크 영역 주변에 다른 링크요소가 없으면 크게 상관이 없을 수도 있겠지만
다른 링크요소들과 함께 44*44 이내에 다닥다닥 붙어 있다면
원하는 것을 터치하지 못할 확률이 높아집니다.

HTML5 미지원 웹 브라우저를 위한 설정 conditional comment

Mobile Device

HTML5 미지원 브라우저를 위해 conditional comment를 입력한다.

```
<!--[if lte IE 8]>  
<script src=http://html5shiv.googlecode.com/svn/trunk/HTML5.js> </script>  
<![endif]-->  
<link href="../css/html5reset.css" type="text/css" rel="stylesheet">
```

스마트폰 접속시 모바일페이지로 접속

Site Selected

Mobile Device

어떻게하면 기존에 웹페이지에서 pc로 접했을때와 스마트폰으로 접했을때 보여지는 페이지를 분류하냐는 겁니다.

이 문제또한 자바스크립트를 통해 스마트폰의 종류를 구별한다음 알맞게 페이지 브라우징 처리를 할수있습니다.

```
<script language="JavaScript">
```

```
var uAgent = navigator.userAgent.toLowerCase();
```

```
var mobilePhones = new Array('iphone','ipod','android','blackberry','windows ce',  
'nokia','webos','opera mini','sonyericsson','opera mobi','iemobile');
```

```
for(var i=0;i<mobilePhones.length;i++)
```

```
if(uAgent.indexOf(mobilePhones[i]) != -1)
```

```
    document.location="링크될 주소";
```

```
</script>
```

Making Your Mobile Web!

Personal site

Mobile Device

HTML5로 가변형 사이트 만들기

1. 세로보기를 중심으로 제작
2. 가변형으로 디자인한다
3. 뷰포트를 삽입하여 모바일에서 문제없이 보도록 만들기
4. 이미지 삽입은 자유롭게~
5. 표준으로 코딩하고 HTML5, CSS3 요소를 사용할 수 있는 부분은 사용해 만들기
6. 가로보기 시에도 깨지지 않고 한쪽으로 쏠리거나 하지 않게 코딩.
7. 미디어쿼리를 사용해서 가로보기시에 조금 덜 어색하게 만들기(아님 완벽하게 해도 됨)
8. 아이폰3, 아이폰4, 갤럭시S, 아이패드, 갤럭시탭등 다양한 기기로 보기.
9. 문제점 리포트 하기

제작 후 검증

W3C Mobile Web Initiative | W3C mobileOK Checker



W3C mobileOK Checker

Is your Web site mobile-friendly?






[Expand all](#) | [Expand one level](#) | [Collapse one level](#) | [Collapse all](#)

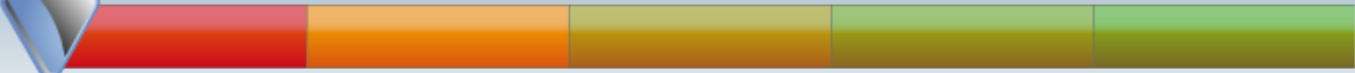
► Address

http://m.naver.com/

▼ Result





0%










0%

FAILURES PER SEVERITY

	CRITICAL	1
	SEVERE	3
	MEDIUM	2
	LOW	4

FAILURES PER CATEGORY

	Rely on Web standards	3
	Be cautious of device limitations	1
	Check graphics and colors	3
	Keep it small	2
	Use the network sparingly	1

↑ TOP

관련 사이트

한국 모바일 웹 2.0 포럼 <http://www.mobileok.kr/web/index.php>

W3C mobileOK Checker <http://validator.w3.org/mobile/>

모바일웹 프랙티스 한국 http://www.w3c.or.kr/Translation/mwbp_flip_cards/

모바일웹 프랙티스 원본 <http://www.w3.org/TR/mobile-bp/>