
웹 앱
네이티브 앱
하이브리드 앱
+ 크로스 플랫폼

2019775068 하수빈

목 차

네이티브 앱

웹 앱

하이브리드 앱

+ 크로스플랫폼



Objective-C

Swift

01 네이티브 앱 (Native App)



모바일 기기에 최적화된 언어로 개발된 앱으로
안드로이드 SDK를 이용해 java, kotlin으로 만드는 안드로이드 앱과
iOS SDK를 이용해 Objective-C, Swift로 개발된 아이폰 앱 등

장점

성능이 웹앱, 하이브리드 앱에 비해 가장 높다

속도가 빠르고 안정적이며 운영체제 업데이트 시 최신 기능을 바로 반영할 수 있으며 보안성이 뛰어나다

카메라, 음성인식과 같은 스마트폰 기능과 2D, 3D 등의 기능도 활용할 수 있다

단점

안드로이드와 IOS용 앱을 각각 따로 개발을 해야 하기 때문에 시간과 비용이 많이 들고 개발 난이도가 높다



m.naver.com

02 웹 앱 (Web App)



모바일 웹과 네이티브 앱을 결합한 것으로
모바일 웹의 특징을 가지면서
네이티브 앱의 장점도 갖는다

모바일웹(Mobile Web)?

데스크탑 브라우저에 실행되는
웹 어플리케이션을
모바일 스크린 크기로 줄여 놓은 것



<기존 PC에 최적화된 웹사이트>



<모바일에 최적화된 웹사이트>

장점

웹사이트를 보는 것이기 때문에 따로 설치할 필요가 없다

모든 기기와 브라우저에서 접근할 수 있다

별도 설치 및 승인과정이 필요치 않아
유지보수가 용이하다

단점

플랫폼 API(카메라 등)를 사용할 수 없고
오로지, 브라우저 API만을 사용할 수 있다

앱을 다운로드해서 기능을 실행하는 것이
아니기 때문에 검색이나 주소 입력 등의 과
정을 거쳐야 한다



03 하이브리드 앱 (Hybrid App)



앱의 기반이 되는
콘텐츠 영역은 HTML 기반의 웹 앱으로 제작,
최종 앱 배포에 필요한 패키징 처리만
아이폰, 안드로이드 플랫폼 안에서 처리한 어플리케이션

장점

네이티브 API와 브라우저 API를 이용한
다양한 개발이 가능하다

웹개발 기술을 사용해 앱을 개발할 수 있다

한번의 개발로 다수의 플랫폼에 대응할 수 있다

단점

네이티브 기능에 접근하기 위해서는
네이티브 개발 지식이 필요하다

웹뷰에서 앱을 실행하는 경우이기 때문에
앱의 성능이 곧 브라우저의 성능이다

UI 프레임워크 도구를 사용하지 않는다면
개발자가 UI를 제작해야 한다

+ 크로스 플랫폼
(Cross Platform)



하나의 프로그래밍 언어로
두 운영체제에서 동작하는 앱을 만드는 것



React Native



Xamarin



Flutter

+ 크로스 플랫폼이 생긴 이유

네이티브 앱이 OS간 호환이 불가능하다는 한계점을 개선하고자 만들어졌다.

+ 네이티브 앱이 더 많은 이유

크로스플랫폼이 해결하지 못하는 단점들이 많이 있기 때문이다.
하이브리드와 크로스플랫폼만으로는 네이티브의 성능을 잘 살릴 수 없고,
사용할 수 없는 네이티브 API들이 많을 뿐더러 나중에 버그가 생길때 해결
할 수 없는 까다로운 문제가 나오는 경우가 많다.

따라서 빠르게 어플리케이션을 만드는 것이 중요하고 주된 사용자층이
고성능의 핸드폰을 사용하고 있고 다양한 네이티브 API를 사용할 필요가
없다면 하이브리드나 크로스플랫폼을 이용해보는 것도 좋을 것 같다.

출처

참고

<https://m.blog.naver.com/acornedu/221012420292>

<https://lillo.co.kr/blog/16480>

이미지

http://www.blueweb.co.kr/2010_mobile01

아이콘

<https://www.flaticon.com/kr>

<https://icon-icons.com/>

ppt 템플릿

<https://blog.naver.com/kingstargirl/223058462644>



감사합니다.