

PBL3- 크로스플랫폼 조사결과 보고서

목차

- 텀프로젝트의 특징
 - 다양한 모바일 크로스 플랫폼
 - 크로스 플랫폼 비교
 - 복잡하지 않은 비즈니스 로직
 - 한정적인 개발 코스트
 - 개발 언어
 - 유사한 UI와 기능 구성
 - 단점에 대한 보완/대응책
-

텀프로젝트의 특징

1. 비즈니스 로직이 복잡하지 않다
 2. 개발 인원과 자금이 한정적이다
 3. 팀원들이 JavaScript 를 다룰 줄 안다
 4. 안드로이드(java)를 기본으로 코드를 작성한다
 5. 페이스북과 비슷한 UI구성, 기능 및 비즈니스 로직을 구현한다
 6. 그래픽 등 난이도가 높은 애니메이션을 구동할 일이 없다.
-

다양한 모바일 크로스 플랫폼

- Xamarin
 - C# 언어 사용
 - 개발환경: Xamarin.ios, Xamarin.Android ...
 - 크로스플랫폼 개발툴 무료
 - Flutter
 - Dart 언어 사용
 - 개발환경: iOS, Android
 - 통합 개발 환경 지원(Android Studio, IntelliJ ...)
 - React Native
 - JavaScript 사용
 - 개발환경: iOS, Android
 - 개발 툴 필요 없음(CRNA, Expo)
 - etc.
-

크로스 플랫폼 비교

1) 복잡하지 않은 비즈니스 로직

- 비즈니스 로직이 커다란 규모의 프로젝트나 상용 앱은 native로 작성해야 크로스플랫폼을 사용한 프로젝트보다 성능면에서 월등
- 다만 텀프로젝트에서 개발중인 앱은 상대적으로 규모가 작다
- 크로스 플랫폼 사용도 무방

2) 한정적인 개발 코스트

- 개발 인원과 자금이 한정적: 오픈소스를 사용하는 것이 적절
 - Xamarin: 오픈 소스 라이브러리를 사용하는데 제한적. **적절하지 않다**
 - Flutter: 오픈 소스 사용 가능
 - React Native: 오픈 소스 사용 가능
- 크로스 플랫폼 사용시 Android, iOS, Window phone 등 플랫폼에 빠르게 배포 가능

3) 개발 언어

- 개발 시간이 한정적이므로 개발자에게 익숙한 언어가 좋다
 - Xamarin: C# 언어 사용. 본격적으로 사용해 본 팀원이 없어 불편하다
 - Flutter: Dart 언어 사용. 구글 언어인 Dart는 코드가 나열되어 가독성이 떨어지고, 제대로 접해본 팀원이 없어 익숙해지기까지 시간 소요.
 - React Native: 어느정도 익숙한 JavaScript를 사용하여 빠르게 React Native 로 변환 가능하다.

4) 유사한 UI와 기능 구성

- 텀프로젝트에서 개발하고자 하는 앱의 특성은 페이스북에서 사용중인 기능이 많다
 - 기본적 로그인 기능
 - 개인 프로필 정보
 - 다수를 초대 가능한 멀티 채팅방
 - 친구 수락
- 위 사항에 따라 이미 세계적으로 많은 사람들이 사용하고 어느정도 안정화 된 페이스북이 사용한 React Native를 사용하면 알맞다.

위 항목을 따져봤을 때 텀프로젝트에는 React Native가 가장 걸맞다.

이에 대한 단점을 아래에서 따져봤다.

5) 단점에 대한 보완/대응책(React Native에 대한 단점)

- 복잡한 UI 기능이 힘들다: 개발하고자 하는 앱의 UI는 심플한 UI를 가지고 있기 때문에 크게 문제되지 않는다.
- 그래픽 등 난이도가 높은 애니메이션 기능 구현이 안된다: 텀프로젝트 앱에서 애니메이션 기능을 사용할 일이 없으므로 단점에서 제외한다.

따라서 텀프로젝트에 사용할 크로스 플랫폼을 고른다면 **React Native** 를 사용할 것이다.