



# CodePush를 통한 React Native 앱 자동 업데이트

|       |                        |
|-------|------------------------|
| 🕒 작성일 | @2023년 4월 3일 오전 11:12  |
| 🕒 수정일 | @2023년 5월 31일 오전 10:28 |
| 👤 작성자 | 🐼 행크                   |
| 👤 참조  | 📱 진저 🐼 원 🔥 우기          |

## 목적

상시 실행중인 react native 앱이 업데이트 될 시 자동으로 앱을 업데이트

## 시나리오

1. React Native 코드 혹은 Asset변경 → 코드 푸시 → 자동으로 업데이트 후 앱 자동 재실행
2. Native 코드 변경 → 코드 배포 → 네이티브 코드 변경 감지 → 사용자에게 앱을 다시 시작하라는 메시지 출력 (코드푸시 사용 불가)

- 시나리오 1

코드푸시를 통해서 소스코드가 변경되어도 앱이 재실행되지 않는 한 변경된 코드가 적용되지 않습니다.

React Native에서 사용하는 JavaScript 런타임이 앱이 실행되는 동안 코드를 자동으로 다시 로드하지 않고 컴파일된 코드를 캐시하고 메모리에서 실행하기 때문입니다.

즉, 업데이트된 코드는 JavaScript 런타임이 다시 시작될 때까지 사용되지 않습니다. (종료 후 재실행 전)

그래서 수동으로 앱을 꺾다 키거나 혹은 CodePush에서 제공하는 restartApp 기능을 이용해서 앱이 재실행되도록 해야 합니다.

- 시나리오 2

다만 앱의 Native Code (Objective-C 또는 Java로 작성된 코드)가 변경된 경우에는 사용자에게 앱을 수동으로 다시 시작하라는 메시지를 출력하여 앱을 재실행하도록 해야합니다.

Native코드가 변경되면 Javascript runtime을 재실행하는것이 아닌 앱 전체를 recompile해야하기 때문입니다. restartApp 을 통해서도 앱이 다시 컴파일되지 않습니다.

## CodePush를 통한 앱 업데이트

CodePush는 MS에서 만든 오픈소스로 React Native나 Apache Cordova로 개발한 앱을 심사과정 없이 바로 업데이트 할 수 있도록 해주는 서비스입니다.

특정 업데이트를 중앙 저장소 역할을 하는 App Center에 JS Bundle과 assets을 업로드하면 모바일 클라이언트는 앱 시작 시 마다 업데이트 버전이 있는지 확인하여 동기화 합니다.

\*App Center또한 MS에서 만든 플랫폼으로 React Native에서만 제공되는 CodePush와 달리 다른 플랫폼에서도 비슷한 기능을 제공하고 모니터링, 리포팅 기능 등 또한 제공합니다.

코드 푸시는 **js 단에서의 변경 사항만** 관리할 수 있으며, 네이티브 코드와 관련된 모든 변경 사항은 스토어 플랫폼을 통해 배포되어야 합니다.

### 프로세스

1. npm을 사용하여 AppCenter CLI를 설치합니다. `$ npm install -g appcenter-cli`
2. AppCenter계정을 생성하고 앱을 등록합니다.
3. 앱의 루트 디렉터리에서 다음 명령을 실행하여 CodePush SDK를 React Native 앱에 추가합니다.

```
npm install --save react-native-code-push
```

4. CodePush HOC로 루트 구성 요소를 래핑하여 CodePush SDK를 앱에 통합해서 앱이 CodePush 서버에서 업데이트를 확인하고 다운로드할 수 있도록 합니다.

# codePush.restartApp

JavaScript

복사

```
codePush.restartApp(onlyIfUpdateIsPending: Boolean = false): void;
```

앱을 즉시 다시 시작합니다. true 값이 매개 변수에 `onlyIfUpdateIsPending` 제공된 경우 실제로 적용 대기 중인 업데이트가 있는 경우에만 앱이 다시 시작됩니다.

이 메서드는 고급 시나리오를 위한 것이며 다음 조건이 충족될 때 주로 유용합니다.

1. 앱이 또는 `LocalPackage.install` 메서드를 호출 `sync` 할 때 또는 `ON_NEXT_RESUME` 의 `ON_NEXT_RESTART` 설치 모드 값을 지정합니다. 이렇게 하면 앱이 다시 시작되거나(최종 사용자 또는 OS에서) 다시 시작될 때까지 업데이트가 적용되지 않으므로 업데이트가 최종 사용자에게 즉시 표시되지 않습니다.
2. 앱별 사용자 이벤트(예: 최종 사용자가 앱의 홈 경로로 다시 이동)를 통해 눈에 거슬리지 않는 방식으로 업데이트를 적용할 수 있으며, 다음 다시 시작 또는 다시 시작할 때까지 기다리는 것보다 더 빨리 최종 사용자에게 업데이트를 받을 수 있습니다.

```
1  import React from "react";
2  import CodePush from "react-native-code-push";
3  import { Provider } from 'react-redux';
4  import { store } from "./src/store";
5  import RootNavigator from "./src/navigator/RootNavigator";
6
7  const App = () => (
8    <Provider store={store}>
9      <RootNavigator />
10    </Provider>
11  );
12
13  export default CodePush(App);
```

5. AppCenter CLI를 사용하여 CodePush 서버에 새 업데이트를 업로드합니다.

```
appcenter codepush release-react -a {user name}/{앱 이름} -d {type}
```

**-m 옵션**을 붙여서 실행하면 해당 업데이트는 사용자의 의사와 관계없이 **강제로 업데이트를 실행**

하게 할 수 있습니다.

CodePush 서버에 새 업데이트를 업로드하면 앱이 자동으로 백그라운드에서 업데이트를 확인하고 다운로드합니다. 업데이트된 코드는 다음에 앱을 시작하거나 백그라운드에서 다시 시작할 때 로드됩니다.

## 참고

<https://velog.io/@minwoo129/React-Native에서-CodePush-사용하기>

<https://medium.com/humanscape-tech/react-native-code-push로-배포하기-26b320d87f8>

<https://learn.microsoft.com/ko-kr/appcenter/distribution/codepush/rn-api-ref#codepushrestartapp>

<https://medium.com/mj-studio/react-native-codepush-as-over-the-air-ota-a-stroke-of-genius-7fbd0cf59ba0>