

총괄시험



모바일 환경에서의 콘텐츠 공유 흐름을 자연스럽게 연결하는 전략

준비 주제

딥링크가 필요한 이유

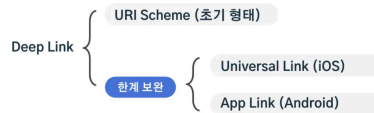
만약, 상품 광고를 보고 눌렀는데

메인화면부터 시작해서 사용자가 직접 그 상품을 찾아가야 한다면

사용자는 이탈

해버리고 말것입니다.

개념



<https://www.airbridge.io/blog-ko/deeplink-101-for-marketers-and-developers>

방식	한계
URL Scheme (Custom Scheme)	1. 앱이 설치되어 있을때만 실행할 수 있음 2. 원하는 스킴을 아무나 만들 수 있어서 같은 스킴을 사용하는 경우가 생김(Scheme의 주인이 없음) 3. 같은 스킴일 경우, Android는 어떤앱으로 열지 선택하는 팝업이 뜨고, iOS는 마지막에 설치한 앱이 열림 → 그래서 악의적으로 다른 앱에서 사용하는 스킴과 똑같은 스킴을 정의해서 내 앱이 열리도록 하게 할 수도 있습니다.
App Link & Universal Link	- Universal Links 는 주소 입력창에 링크를 직접 복사 붙여 넣기 할 경우 작동하지 않습니다. - Universal Links 는 JavaScript로 트리거 될 경우 리다이렉트가 되지 않습니다. - Universal Links 는 openUri과 같이 앱 내에서 프로그래밍 방식으로 링크를 열 경우, 작동하지 않습니다. - 각각의 플랫폼에서 자신들이 만든 앱(크롬, 사파리, 메일 등)으로 실행하면 의도대로 잘 동작하지만 그외에서는 동작하지 않는 경우도 있음 ios 9이상 사용 가능 IOS 10 부터는 Open with 다이얼로그까지 뜸 안드로이드는 6부터 사용가능 (API version)
Deferred Deeplink	TODO iOS도 있나?

주의사항

- 안드로이드의 경우, 열고자 하는 앱들의 패키지를 정보 manifest 파일에 등록해두어야함
 - 패키지를 등록하지 않으면 앱이 설치되어 있어도 스토어로 이동하고, Google 정책을 위반해요.

```
<queries>
  <package android:name="com.kakao.talk" /> <!-- 카카오톡 -->
  <package android:name="com.shcard.smartpay" /> <!-- 신한페이핀 -->
  <package android:name="com.shinhacard.smartshinhan" /> <!-- 신한페이핀-공동인증서 -->
  <package android:name="com.hyundaicard.appcard" /> <!-- 현대카드 -->
  <package android:name="com.lumensoft.touchenappfree" /> <!-- 현대카드-공동인증서 -->
  <package android:name="kr.co.samsungcard.mpocket" /> <!-- 삼성카드 -->
  <package android:name="nh.smart.nhallonepay" /> <!-- 올원페이 -->
  <package android:name="com.kbcard.cxb.appcard" /> <!-- KB Pay -->
  <package android:name="com.kbstar.liivbank" /> <!-- Liiv(KB국민은행) -->
  <package android:name="com.kbstar.reboot" /> <!-- Liiv Reboot(KB국민은행) -->
  <package android:name="kvp.jjy.MispAndroid320" /> <!-- ISP/페이북 -->
```

- 웹뷰에서 열고자 하는 url이 일반적인 웹 Uri인지, app link url인지, intent인지 판단함
 - 웹 url → 웹뷰에서 그냥 이 url로 이동시킴
 - app link url → startActivity
 - intent → startActivity(Intent()) 시도
 - 에러나면 앱스토어로!
- iOS의 경우
 - 먼저 Info.plist 에 **LSApplicationQueriesSchemes** 를 추가하고 카드사, 은행의 앱 스킴을 배열에 넣어 주세요. 설정하지 않으면, 앱이 열리지 않고 콘솔 쪽에 **canOpenURL : failed for URL** 에러가 발생해요.

```
Info.plist

<key>LSApplicationQueriesSchemes</key>
<array>
  <string>supertoss</string>
  <string>kb-acp</string>
  <string>liivbank</string>
  <string>newliiv</string>
  <string>kbbank</string>
  <string>nhappcardansimclick</string>
```

- <https://docs.tosspayments.com/blog/android-ios-webview-deeplink>
- URL 스킴이 http 혹은 https 이 아니라 커스텀 스킴이면, 링크를 열어봐요. 정상적으로 링크가 열린다면 성공이고 앱이 이동해요. **안 열린다면 사용자 폰에 앱이 설치 안 되어 있어요. 사용자를 수동으로 앱 스토어로 이동시킬 수는 있지만, 이 방법은 유지 보수가 어려워요.** 앱 스토어 아이디가 바뀔 수 있기 때문이죠. 2023년 3월 기준으로 국내 카드·은행 앱의 앱 스토어 아이디는 아래 코드에서 확인할 수 있어요.
- **URL 스킴이 http 혹은 https 이라면 Universal Link 이겠쎄. 웹뷰에서 바로 열면 됩니다. 사용자 폰에 앱이 설치되어 있지 않으면 바로 이동하고, 설치되어 있지 않으면 보통 각 카드·은행 앱(웹사이트)에서 사용자를 스토어로 이동시켜요.**

URI Scheme - 앱이 설치안되어있을때 - 안드로이드

Intent 스킴은 위와 같이 상당히 복잡한 형태를 가지고 있어요. 앱의 스킴, 패키지 등 많은 정보를 담고 있죠. 복잡하지만 개발자 입장에서는 Intent 스킴에서 제공하는 정보를 가공해서 다양한 방식으로 앱을 열 수 있죠. **예를 들어 앱이 설치되지 않았거나 invalid 한 상태인 것을 확인하고 앱 패키지 정보로 플레이 스토어를 열 수 있어요.**

- 안드로이드의 경우 인텐트 스킴으로 스토어 설치페이지로 이동 - 그러나, 만약 앱 id가 바뀐다면 대응해야함 → 유지보수에 좋지 않음
- iOS에서는 제공하지 않는 기능!

URI Scheme - 앱이 설치안되어있을때 - iOS

iOS에서는 안드로이드의 intent처럼 다른 앱을 실행시키기가 힘들다.

그래서 Best는 Universal link를 이용하는 것이지만, universal link가 별도로 앱에 세팅되어있지 않은 경우 사용하지 못함

따라서 아래와 같은 코드를 이용하여 예외에 대응한다.

```
useEffect(() => {
  function handler() {
    clearTimeout(timerRef.current);
```

```

}

// (3) window의 visibilitychange 이벤트가 발생하면 앱이 실행됐다고 가정한다. timeout을 제거한다.
window.addEventListener('visibilitychange', handler);
}, []);

const launchApp = (url: string) => {
  // (1) 주어진 딥링크로 앱을 실행한다
  location.href = url;

  // (2) 일정한 시간동안 앱이 실행되지 않으면 특정 기능을 실행하는 timeout을 생성한다
  timerRef.current = setTimeout(() => {
    onLaunchFail();
  }, 2000);
}

```

- 위 로직은 ios 10 버전 이상에서 “Open app with ... ” 다이얼로그도 당연히 핸들링가능
- 토스에서는 딥링크를 통해 이동하고자 하는 앱이 설치되어있지 않다는 에러메세지를 띄운다.
- iOS에서는 아마 이런 링크를 통해 웹에서도 앱스토어로 이동가능하다:

- `window.location = "https://apps.apple.com/app/myapp";`

위의 내용은 딥링크 클릭을 우리 앱 안에서 할때 얘기이고

만약 친구한테 공유받은 url scheme에 해당하는 앱이 존재하지 않는 경우

- ios
 - 아무일도 일어나지 않는다 → 즉, 앱 설치가 안되어있을때 대응이 불가능하므로 좋지 못한 방법
- 안드로이드도 동일

Universal Link, App Link 소유자 인증 받기

URL 인증 받는것은 어떻게해?- 앱 빌드 시 앱을 열 도메인을 등록하고, 보안을 위해 등록된 도메인의 특정 경로에 인증 텍스트를 심어 놓습니다. 그러면 앱이 설치된 후 OS 레벨에서 해당 도메인 경로의 인증 텍스트와 앱 빌드 시의 값이 동일한지 검증하여 소유권을 인지하게 됩니다. 일단 검증이 완료된 후에는 앱을 삭제하기 전까지 해당 검증은 유효합니다.

- Android: <https://deeplink.heydealer.com/.well-known/assetlinks.json>
- iOS: <https://deeplink.heydealer.com/.well-known/apple-app-site-association>

Universal Link & App Link - 앱이 설치되지 않았을때 - android & iOS

앱이 설치되어 있지 않았다고 자동으로 스토어로 이동하는 것은 아님!

1. 앱이 설치되어 있다면 **해당 딥링크에 맞는 화면이 실행됨**
2. 앱이 설치되어 있지 않다면, **웹 URL로 인식되어 실제 해당 웹 링크를 열려고 시도함**
3. 웹서비스도 함께 제공하고 있다면 해당 웹서비스에서 맞는 콘텐츠를 보여줄것임
4. 하지만 그렇지 않다면, 해당 URL로 웹으로 열었을때앱 마켓(구글 플레이, 앱스토어)으로 이동시키도록 처리해두면 됨
 - a. 이 경우 어차피 우리앱을 여는거라서 app id나 마켓스토어 내의 우리 앱 주소 관리 하기 편함

흐름

1. 공유기능 클릭
2. 유니버설 링크 같이 복사해주기
 - a. 현재 앱이 설치된 기기의 OS 버전에 따라 다른 처리
 - i. app link, universal link를 못쓰는 경우에는

1. url schme을 써야하고
2. 그 경우 앱이 안깔려있는 경우 OS에서 아무런 반응이 없으니
3. OS 버전이 낮아서 공유기능이 지원이 안될 수 있어요~ 이런 문구 써놓기
3. 다운로드 안되어있을때 처리
 - a. iOS
 - i. 둘다 위에 나옴
 - b. 안드로이드
4. 다운로드되어있을때 구현
 - a. 앱으로 진입해서 알맞은 화면을 보여주면됨
 - b. 어렵지 않음

기타

- 앱 배너는 공유를 위한 전략은 아니지 않을까? 광고를 위한 전략이지 않을까?

설계

- ☐ 무엇을 공유할 것인가
 - ☐ (단순 텍스트, 특정 데이터)
- ☐ 어떻게 공유할 것인가
 - ☐ 딥링크
 - ☐ 앱링크
 - ☐ TODO 공유기능에서 선택하는 앱에는 뭘 전달하는 거지? 그냥 텍스트??
 - ☐ TODO 아 공유할때 https/http url을 사용하면 유니버설링크가 되는구나! 마치 네이버 지도앱처럼!
- ☐ OS 간 차이는 어떻게 되는가?
- ☐ 받는 앱에서는 어떻게 처리할 것인가?
- ☐ 받고나서는 원래앱에서는 무엇을 할 것인가?

고려사항

- ☐ 인스타그램, 카카오톡, 네이버 블로그 등 외부 앱으로 콘텐츠를 공유할 때 발생할 수 있는 제약사항 2가지를 기술하시오.
- 예시 답: Custom Scheme 인증 필요 / 외부 SDK 버전 호환성 문제 등
- ☐ 동일 앱그룹인가?
 - ☐ Fallback 설치 방법

안드로이드 딥링크 구현

```

<activity ...>
  ...
  <intent-filter android:autoVerify="true">
    <action android:name="android.intent.action.VIEW" />

    <category android:name="android.intent.category.DEFAULT" />
    <category android:name="android.intent.category.BROWSABLE" />

    <data android:scheme="http" android:host="www.airbridge.io/" />
    <data android:scheme="https" android:host="www.airbridge.io/" />
  </intent-filter>
  <intent-filter>
    <action android:name="android.intent.action.VIEW" />

    <category android:name="android.intent.category.DEFAULT" />
    <category android:name="android.intent.category.BROWSABLE" />

    <data android:scheme="airbridge" />
  </intent-filter>
  ...
</activity>

```

☐ data android:scheme="airbridge"는 뭐지?

구현 순서

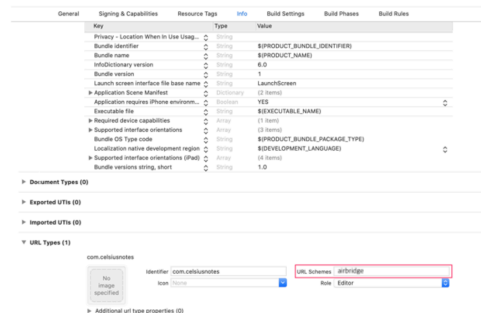
1. 안드로이드 스튜디오 활용해서 activity 별로 universal link 설정
2. 안드로이드 스튜디오 활용해서 assetlinks.json에 들어갈 내용 생성 & [도메인/.well-known/assetlinks.json](#) 에 업로드
- 3.

iOS

URI 스킴

iOS에서는 하나의 화면으로 딥링크가 열리도록 설정하고, 이후 해당 화면에서 각각의 상세 페이지로 이동하도록 구현하는 방법만 지원합니다.

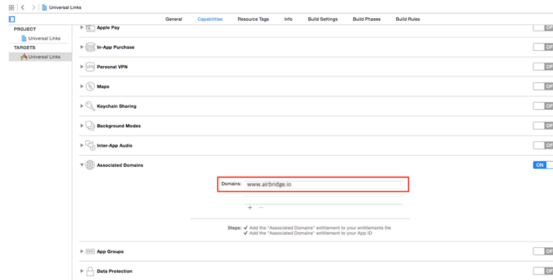
URI Scheme을 설정할 때는, 아래 이미지와 같이 Xcode > Project 파일 > Info > URL Types에서 iOS URI Scheme을 입력합니다.



☐ TODO 그러면 이 화면에서 다른화면으로 가는건 어떻게 처리해?

유니버설 링크

Universal Links의 경우, 아래의 이미지와 같이, Xcode > Project 파일 > Signing & Capabilities 로 이동 후 + Capability 를 눌러 Associated Domains 에 Universal Links로 설정하고 싶은 도메인을 추가하면 됩니다.



기술 구현

☐ 딥링크, 앱링크

☐ 웹에서 앱을 이동시킬 땐 어떤 설정이 필요한지, 사용자 폰에 앱이 설치되어 있는지, 안 되어 있으면 사용자를 어디로 이동시킬지 모두 구현해야 돼요.

☐ 커스텀 스킴의 문제점

☐ 그러나 커스텀 스킴에는 서로 다른 앱에서 같은 스킴을 사용할 수 있다는 문제가 있어요. 앱이 스킴을 등록할 때 이미 있는 스킴인지 아닌지를 확인할 수가 없거든요.

☐ Android에서는 같은 스킴을 가진 앱이 두 개가 있다면 “어떤 앱을 열래?”라고 묻는 선택 UI가 뜨지만, **iOS에서는 이런 문제를 해결할 방법이 없어요.**

☐ 앱링크 & 유니버설 링크

☐ 안드로이드 6.0이상에서만 사용가능한데 그전 버전은 어떻게 처리?

☐ **그러나 App Link와 Universal Link는 사용자의 액션으로만 작동해요. 스크립트로 클릭을 유발하면 앱으로 이동하지 않고, 그대로 웹 브라우저에서 링크가 열려요. 그래서 여전히 커스텀 스킴 URL과 함께 사용되고 있어요.**

☐ 유니버설 링크

참고자료

- <https://medium.com/prnd/딥링크의-모든것-feat-app-link-universal-link-deferred-deeplink-61d6cf63a0a5>
- https://nbcamp.spartaclub.kr/blog/개념-꼭-안드로이드-deep-links와-app-links의-차이점-21552?utm_source=google&utm_medium=PMAX2&utm_campaign=nbc&utm_content=trackall&utm_term=&gad_source=1&gad_
- <https://docs.tosspayments.com/blog/how-to-use-deep-links>
- <https://docs.tosspayments.com/blog/android-ios-webview-deeplink>