

## INIpayMobile WebView 구현 시 참고문서

---

### SETUP GUIDE

# 1. 앱 환경의 설치방법(안드로이드)

## 1-1. 기본적인 설치방법

안드로이드 어플리케이션 내 WebView (이하 WebView) 에서 이니페이모바일™ 을 구현하는 경우에 해당됩니다.

WebView 에서 INIpayMobile 을 띄우는 방식은 기본 웹 연동방법과 동일합니다.

이에 해당 문서에서는 mobileISP 앱 호출시 주의사항, 카드사 백신 앱 스키마 호출 및 “미설치 시, 앱스토어 이동 이슈” 등의 내용을 주로 다룹니다.

### A. mobileISP 연동방법 - 앱 미설치 체크로직 직접구현 or 자동체크

mobileISP 앱의 기본정보는 하기와 같습니다.

Application Scheme	ispmobile://
Install Url	http://mobile.vpay.co.kr/jsp/MISP/andown.jsp

상기 Scheme 과 Install Url 정보로 구현가능한 안드로이드 코드는 하기와 같습니다.

- ① WebViewClient 를 상속받은 클래스를 구현하시고, shouldOverrideUrlLoading() 을 호출 하십시오.

```
private class INIP2PWebView extends WebViewClient {

    @Override
    public boolean shouldOverrideUrlLoading(WebView view, String url) {

        ...

    }
}
```

- ② 상기 shouldOverrideUrlLoading() 함수 내에, try{} catch{e} 를 통해, try 내에서는 startActivity(intent) 를 구현하시고, catch Event 발생 시, 앱 스토어로 이동할 수 있도록 조치하시면 됩니다.

하기에 안내되는 소스는 상기 방식에 대한 Full-Source 입니다. (다음페이지)

[shouldOverrideUrlLoading 부]

```
private class INIP2PWebView extends WebViewClient {
    @Override
    public boolean shouldOverrideUrlLoading(WebView view, String url) {
        ...
        Uri uri = Uri.parse(url);
        Intent intent = new Intent(Intent.ACTION_VIEW, uri);
        try{
            startActivity(intent);
            //삼성카드 안심클릭을 위해 추가
            if( url.startsWith("ispmobile://")) finish();
        }
        catch(ActivityNotFoundException e)
        {
            //url prefix 가 ispmobile 일거우만 alert 를 띄움
            if( url.startsWith("ispmobile://"))
            {
                view.loadData("<html><body></body></html>", "text/html", "euc-kr");
                alertIsp.show();
                return true;
            }
        }
        ...
        return true;
    }
}
```

[ISP 앱스토어 이동처리 부] - alertIsp

```
protected void onCreate(Bundle savedInstanceState) {
    ...
    alertIsp = new AlertDialog.Builder(PaymentView.this)
        .setIcon(android.R.drawable.ic_dialog_alert)
        .setTitle("알림")
        .setMessage("모바일 ISP 어플리케이션이 설치되어 있지 않습니다. \n\n 설치를  
눌러 진행 해 주십시오.\n\n 취소를 누르면 결제가 취소 됩니다.")
}
```

```

.setPositiveButton("설치", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialog, int which) {

        //ISP 설치 페이지 URL
        paymentView.loadUrl("http://mobile.vpay.co.kr/jsp/MISP/andown.jsp");

        finish();
    }

})

.setNegativeButton("취소", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialog, int which) {
        Toast.makeText(PaymentView.this, "(-1)결제를 취소 하였습니다."
            Toast.LENGTH_SHORT).show();
        finish();
    }
}).create();
...
}

```

- ③ mobileISP 가 단말기에 기 설치되어 있는 경우, mobileISP 가 정상구동 될 것이며,
- ④ mobileISP 가 단말기에 미 설치되어 있는 경우, 설치 후, 이니페이모바일™ 을 다시 띄워 주시면 됩니다. 23 페이지의 예시 [shouldOverrideUrlLoading 부] 의 ① 에 대하여 true 혹은 false 를 설정하는 것은 하기의 표를 참고바랍니다.

apprun_check	작동방식	앱 미설치 시, 앱스토어 이동 후, 결제페이지 잔존여부	①true / false 설정
Y	ISP, 계좌이체앱 - intent 작동	상태 유지	true
N or 미설정	ISP, 계좌이체앱 - appScheme 작동	하기 그림 1 과 같이 Display 됨	false

[shouldOverrideUrlLoading 부] 의 ①을 true 로 할 경우, 이니페이모바일 TM 을 띄운 WebView 는 사라집니다.

따라서, app Scheme 형태로 결제 앱을 호출 할 경우에는 “그림 1” 과 같이 오류 페이지가 Display 되기 때문에, WebView 를 remove 하는 것이 좋습니다.



[그림 1]

[shouldOverrideUrlLoading 부] 의 ①을 false 로 할 경우, 이니페이모바일™ 을 띄운 WebView 는 사라지지 않기 때문에, apprun\_check=Y 를 통해 현 결제 페이지가 유지되는 방식을 사용하는 것이 좋습니다. 이 방법을 자동체크방식이라 합니다.

단, apprun\_check 옵션을 통해 설치체크로직이 작동되므로, alertIsP 함수는 구현될 필요가 없습니다.

또한, Intent 호출에 대하여 예외처리를 반드시 체크하셔야 합니다.

## B. mobileISP 연동방법 – 인증결과 전송

ISP 앱에서 인증과정이 완료되면, 다시 당사 모바일 결제창으로 돌아와서 하기의 이미지와 같이 ‘확인’ 버튼을 클릭해야, 승인과정을 시작하게 됩니다.



안드로이드는 운영체제 특성 상, 현재 앱이 종료될 경우, 이 앱을 실행시킨 이전 앱이 다시 자동으로 수행됩니다. (LIFO 방식) 따라서, ISP 앱이 종료되면, 가맹점의 앱은 자동으로 다시 활성화 될 것입니다.

### C. 안심클릭 결제 시, 카드사 백신 앱 연동

이니페이모바일™ 은 BC 계열을 제외한, 나머지 카드사의 결제창은 카드사 페이지에서 결제가 이루어지고 있습니다. 이에, 카드사에서 개별적으로 사용하는 백신 앱의 경우, 가맹점 앱에서도 하기의 유의사항을 반드시 체크하셔야 합니다.

- ① WebView 내에서 http 와 https URL, 그리고 App Url 을 분기하여 처리해야 함.
- ② shouldOverrideUrlLoading() 처리로직을 하기와 같이 구현함.

App Url 일 경우	activity 호출
Web Url 일 경우	WebView 에서 Loading

상기의 유의사항을 고려한 샘플 코드는 하기와 같습니다. (Kitkat 이하 정상구동여부 확인됨)

```
private class SampleWebViewClient extends WebViewClient {
    @Override
    public boolean shouldOverrideUrlLoading(WebView view, String url) {
        Log.d("<INICIS_TEST>", "URL : "+url);
        /*
         * URL 별로 분기가 필요합니다. 어플리케이션을 로딩하는것과
         * WEB PAGE 를 로딩하는것을 분리 하여 처리해야 합니다.
         * 만일 가맹점 특정 어플 URL 이 들어온다면
         * 조건을 더 추가하여 처리해 주십시오.
         */
        if( !url.startsWith("http://") && !url.startsWith("https://")
            && !url.startsWith("javascript:") )
        {
            Intent intent;
            try{
                intent = Intent.parseUri(url, Intent.URI_INTENT_SCHEME);
                Log.d("<INICIS_TEST>", "intent getDataString : " + intent.getDataString());
            } catch (URISyntaxException ex) {
                Log.e("<INICIS_TEST>", "URI syntax error : " + url + ":" + ex.getMessage());
                return false;
            }
        }
    }
}
```

```

try{
    startActivity(intent);
}catch(ActivityNotFoundException e){
    /* ISP 어플이 현재 폰에 없다면 아래 처리에서
    * 알림을 통해 처리하도록 하였습니다.
    * 삼성카드 및 기타 안심클릭에서는
    * 카드사 웹페이지에서 알아서 처리하기때문에
    * WEBVIEW 에서는 별다른 처리를 하지 않아도 처리됩니다.
    */
    if( url.startsWith("ispmobile://"))
    {
        //onCreateDialog 에서 정의한 ISP 어플리케이션 알럿을 띄워줍니다.
        //(ISP 어플리케이션이 없을경우)
        showDialog(DIALOG_ISP);
        return false;

    }

}else if (url.startsWith("intent")) {
    //일부카드사의 경우 ,intent:// 형식의 intent 스키마를 내려주지 않음
    //ex) 현대카드 intent:hdcardappcardansimclick://

try {
    Intent tempIntent = Intent.parseUri(url, Intent.URI_INTENT_SCHEME);
    String strParams = tempIntent.getDataString();

    Intent intent = new Intent(Intent.ACTION_VIEW);
    intent.setData(Uri.parse(strParams));

    startActivity(intent);

    return true;
} catch (Exception e) {
    e.printStackTrace();

    Intent intent = null;

    try {
        intent = Intent.parseUri(url, Intent.URI_INTENT_SCHEME);
        Intent marketIntent = new Intent(Intent.ACTION_VIEW);
        marketIntent.setData(Uri.parse("market://details?id=" + intent.getPackage()));
        startActivity(marketIntent);
    }
}
}

```

```

        } catch (Exception e1) {
            e1.printStackTrace();
        }
        return true;
    }
}

    }
}
else
{
    view.loadUrl(url);
    return false;
}
return true;
}
}

```

#### D. 결제 금액이 30 만원 이상일 때의 공인인증 앱 연동 방법

만약 상점에서의 판매가격이 30 만원 이상일 수 있고 카드로 결제할 경우, 사용자는 공인인증서 서명과정을 거쳐야 합니다. 안드로이드의 경우, 개별 카드사 앱에서 공인인증서 서명을 할 수 있습니다. 이에, 카드사 창 내에서 호출하는 intent 혹은 app Scheme 를 허용할 수 있도록 가맹점 앱에서 처리해줘야 합니다. 이는 **(1.C. 안심클릭 결제 시, 카드사 백신 앱 연동)** 의 코드를 반영하면 해결됩니다.

#### E. Android API Level 21 이상 일 때, 체크사항

Android API Level 21 (Lollipop 출시 때 배포) 부터는 webview 에서 Insecurity Page 에 대한 Access 및 Mixed contents, Third party cookies 사용을 차단할 수 있게 업데이트 되었습니다.

먼저, Insecurity Page 에 대한 Access 차단으로 P\_NEXT\_URL 의 Scheme 을 Http 로 하는 경우, 페이지가 호출되지 않아 인증결과가 전달되지 않을 수 있습니다.

하기의 설정을 확인하십시오.



상 태	코 드
<b>Insecurity 페이지 차단</b>	<pre>WebSettings web = paymentView.getSettings(); web.setMixedContentMode(web.MIXED_CONTENT_NEVER_ALLOW);</pre>
<b>Insecurity 페이지 허용</b>	<pre>WebSettings web = paymentView.getSettings(); web.setMixedContentMode(web.MIXED_CONTENT_ALWAYS_ALLOW);</pre>

P\_NEXT\_URL 의 Scheme 이 Http 일 경우, 반드시 "Insecurity 페이지 허용" 으로 설정되어야 합니다.

또한, Third party cookies 사용의 차단으로 안심클릭 카드 결제 시, 보안 키보드를 불러오지 못 하는 이슈 등이 발생할 수 있으니 하기 설정을 확인하십시오.

상 태	코 드
<b>Third party cookies 허용</b>	<pre>CookieManager cookieManager = CookieManager.getInstance(); cookieManager.setAcceptCookie(true); cookieManager.setAcceptThirdPartyCookies(sampleWebView, true); // false 설정 시 오류 발생</pre>

## 2. 앱 환경의 설치방법(iOS)

### 2-1. 기본적인 설치방법

IOS 어플리케이션 내 WebView (이하 WebView) 에서 이니페이모바일™ 을 구현하는 경우에 해당됩니다.

WebView 에서 INIpayMobile 을 띄우는 방식은 기본 웹 연동방법과 동일합니다.

이에 해당 문서에서는 mobileISP 앱 호출시 주의사항, 카드사 백신 앱 스키마 호출 및 “미설치 시, 앱스토어 이동 이슈” 등의 내용을 주로 다룹니다

#### A. 신용카드 ISP, 계좌이체 연동방법

신용카드 ISP 및 계좌이체 앱이 종료 된 뒤, 가맹점 앱을 다시 띄우기 위한 조치사항을 안내합니다.

IOS 는 Android 계열과 다르게도 ISP 및 계좌이체 앱이 종료된 뒤, 가맹점 앱은 Background 에 머문채, 바탕화면이 개제됩니다. (IOS 의 운영체제 특성에 기반) 이 때문에, 결제 앱이 종료되면서, 가맹점 appScheme 을 호출하도록 구성해야 합니다. 하기와 같이 셋팅 시, 요구사항과 같이 가맹점 앱이 다시 기동됩니다. (신용카드,계좌이체 동기옵션 사용시에만 설정 가능)

신용카드	P_RESERVED=app_scheme=가맹점스키마명://
계좌이체	P_RESERVED=iosapp=Y&app_scheme=가맹점스키마명:// P_RETURN_URL=가맹점스키마명://

상기 옵션 셋팅 시, 가맹점스키마명 뒤 **://** 은 필수로 입력해주셔야 mobileISP 앱 종료 후 가맹점 앱이 호출 됩니다. (Ex. 가맹점 스키마명이 INIpayMobile 일 경우 app\_scheme=INIpayMobile:// 로 셋팅해 주시면 됩니다.)

#### B. 안심클릭 결제 시, 카드사 백신 앱 연동

IOS 환경에서는 카드사에서 별도로 백신을 구동하지 않습니다.  
따라서, 해당 부분은 체크하실 부분이 없습니다.

### C. 카드사 앱 연동 방법

하기의 조건을 충족하는 경우에 결제가 가능하오니, 이점 유의 바랍니다.

- ① 고객 단말기의 OS 버전이 4.x 이상인 경우
- ② 가맹점 Application 이 Multi switching 이 지원되는 경우
- ③ OS 버전이 9.x 이상일 경우, 하기 "IOS9 Application 구현 시, 확인사항" 내용을 참고바랍니다.

### D. 쿠키 설정

이니페이모바일™ 을 IOS WebView 에서 호출하고, 안심클릭 계열 서비스를 사용하는 경우, 세션만료 오류경고가 발생할 수 있습니다. 이에, 하기의 샘플과 같이 쿠키를 허용해야 합니다.

```
(BOOL)application:(UIApplication *)application
didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    [[NSHTTPCookieStorage sharedHTTPCookieStorage]
    setCookieAcceptPolicy:NSHTTPCookieAcceptPolicyAlways];

    ...

    return YES;
}
```

### E. IOS9 버전 Application 구현 시, 주의 사항

IOS9 업데이트 이후, APP 내 보안정책 강화로 canOpenUrl 또는 openUrl 함수 사용 시, info.plist 파일에 LSAApplicationQueriesSchemes 배열을 정의하여 호출할 App scheme list 를 등록 해주셔야 합니다. (White List 등록)

아래는 LSAApplicationQueriesSchemes 등록 예시이며, 기존 앱 스키마에서 "://" 부분을 제거 후, 등록하시면 됩니다.

Information Property List		Dictionary	(23 items)
LSApplicationQueriesSchemes	Array		(0 items)
item0	String		AppScheme0
item1	String		AppScheme1
item2	String		AppScheme2

예시 1. XCODE info.plist

```

<key> LSAApplicationQueriesSchemes </key>
  <array>
    <string> fbapi </string>
    <string> fbauth2 </string>
    <string> fbshareextension </string>
    <string> fb-messenger-api </string>
    <string> twitter </string>
    <string> whatsapp </string>
    <string> wechat </string>
    <string> line </string>
    <string> instagram </string>
    <string> kakaotalk </string>
    <string> qq </string>
    <string> vk </string>
    <string> qq </string>
  </array>

```

예시 2. XML info.plist

아래 MOBILETX 에서 사용 중인 Custom Scheme List 를 참고하셔서 지불수단 별, 필요한 부분 사용 바랍니다

<2015.10.05 Custom Scheme List>

지불수단	App	custom scheme
신용카드	신한 앱카드	shinhan-sr-ansimclick://
	신한 공인인증 앱 (일반결제)	smshinhanansimclick://
	현대 앱카드	hdcardappcardansimclick://
	현대 공인인증 앱 (일반결제)	smhyundaiansimclick://
	삼성 앱카드	mpocket.online.ansimclick://
	삼성 공인인증 앱 (일반결제)	scardcertiapp://
	하나 앱카드	cloudpay://
	하나 공인인증 앱 (일반결제)	hanaskcardmobileportal://

	농협 공인인증 앱 (일반결제)	nonghyupcardansimclick://
	국민 앱카드	kb-acp://
	롯데 앱카드	lotteappcard://
	롯데 스마트페이	lottesmartpay://
	KPAY	kpay://
	ISP	ispmobile://
	PAYCO	payco://
	Syrup 앱카드	tswansimclick://
	농협 앱카드 (올원페이)	nhallonepayansimclick://
	씨티 앱카드 (2017.01.05 신규)	citimobileapp://
	LPAY	lpayapp://
	카카오페이	kakaotalk://
	SSGPAY	shinsegaeasypayment://
	우리 앱카드 (우리페이)	wooripay://
	토스페이	supertoss://
	하나멤버스월렛	hanawalletmembers://
	차이페이	chai://
해외 지불수단	은련카드	uppay://

### 3. 계좌이체 연동 유의사항 안내

가맹점 안드로이드/iOS 앱에서 계좌이체를 연동하시는 경우 아래의 앱스키마 정보에 대해  
가맹점앱에서 처리 가능하도록 구현되어 있어야 하며, OS 별 상세 처리는 아래를 참고바랍니다.

App	custom scheme
뱅크페이 (기본)	kftc-bankpay://
뱅크월렛	bankwallet://
케이뱅크 페이	ukbanksmartbanknonloginpay://
산업	kdb-bankpay://
기업	ibk-bankpay://
국민	kb-bankpay://
KEB 하나	keb-bankpay://
수협	sh-bankpay://
NH 농협	nhb-bankpay://
농축협	nh-bankpay://
우리	wr-bankpay://
SC	sc-bankpay://
신한	s-bankpay://
씨티	ct-bankpay://
대구	dg-bankpay://
부산	bnk-bankpay://
광주	kj-bankpay://
제주	jj-bankpay://
전북	jb-bankpay://
경남	kn-bankpay://
우정사업본부	kp-bankpay://
신협	cu-bankpay://
새마을	mg-bankpay://
K BANK	kbn-bankpay://
카카오뱅크	kkb-bankpay://

### 3-1. IOS 연동 시 유의사항

IOS 가맹점 앱에서 계좌이체 앱 실행을 위해서는 위의 스키마정보가 info.plist 파일에 White List 로 등록되어 있어야합니다. 해당 스키마 설정 추가와 관련하여 상세 내용은 이전 장에서 안내된 가이드를 참고바랍니다.

“2.앱 환경의 설치방법(IOS)”->“ E. IOS9 버전 Application 구현 시, 주의 사항”

### 3-2. 계좌이체 연동 시 권장 옵션

환 경	필드 및 옵션
안드로이드 웹/앱, 아이폰 웹	별도 설정 필요치 않음
아이폰 앱	P_RESERVED=iosapp=Y&app_scheme={가맹점앱스키마};// P_RETURN_URL={가맹점앱스키마};//

## 4. 케이페이 연동방법

당사 간편결제 서비스인 케이페이는 영업담당자를 통해 지불수단에 추가할 수 있습니다.

#### A. Android WebView 에서의 처리

케이페이는 호출 시, Intent 방식으로 호출됩니다.

따라서, 상기에서 안내된 shouldOverrideUrlLoading() 을 처리했다면, 더 이상 처리할 것은 없습니다. 또한, Intent 방식으로 호출되기 때문에, 케이페이가 설치되지 않은 단말기일 경우, 자동으로 앱스토어로 이동됩니다. 다만, 앱스토어 이동 및 설치 후에는 결제단계를 처음부터 수행해야 합니다.

#### B. IOS WebView 에서의 처리

IOS에서는 AppScheme 형태로 호출되며, 기본적인 동작 방식은 Android 와 동일합니다.

또한, 앱스토어 이동 후에 결제단계를 처음부터 시도해야 하는 로직 또한 동일합니다.

### C. 케이페이 바로가기 옵션

P\_RESERVED(복합필드) 에 하기의 옵션을 추가할 경우, 다른 카드리스트를 Display 하지 않고, 바로 케이페이를 노출시킬 수 있습니다. 하기의 스크린샷을 참고하여, 사용하시기 바랍니다.

스크린샷	Key&Value	설명
	d_kpay=Y	케이페이 설치버튼이 있는 결제페이지를 노출합니다.
	d_kpay=Y& d_kpay_app=Y	별도의 결제페이지를 노출하지 않고, 즉시 케이페이를 실행합니다.



## 5. 앱 연동 시 주의사항

### 주의사항 1) 브라우저 환경에서의 쿠키허용 issue

이니페이모바일™ 을 정상적으로 사용하기 위해서는 사용자의 브라우저에서 쿠키를 허용하는 상태여야 합니다

Android	기본	설정 > 개인정보 보호 및 보안 > 쿠키허용 체크활성
	권장(필수)	설정 > 개인정보 보호 및 보안 > 쿠키허용 체크활성
IOS	기본	설정 > Safari > 쿠키차단 > 내가 방문한 웹사이트에서 허용
	권장(필수)	설정 > Safari > 쿠키차단 > 항상 허용

※ IOS 는 버전에 따라 메뉴 명이 상이할 수 있습니다.

### 주의사항 2) 안드로이드/IOS 앱 USER AGENT 수정

신용카드 결제 시 각 카드사 ACS 페이지에서는 사용자의 USER AGENT 를 검증하고 있습니다.  
( 검증방식은 카드사별 상이 ) 가맹점 앱에서 USER AGENT 를 임의로 변경하는 경우, 카드사 인증실패가 발생할 수 있으니 이점 유의 바랍니다.

### 주의사항 3) 아이폰 3rd party(제 3 공급자) 앱브라우저 결제 issue

iOS 기본 브라우저인 Safari 외 앱브라우저(다음, 네이버, 크롬, 페이스북 등) 에서 ISP, 계좌이체 결제 진행 시, ISP 앱에서 인증 완료 후 결제를 진행한 브라우저 앱이 아닌 Safari 브라우저로 돌아가는 이슈가 있습니다.

해당 브라우저 앱에서 결제를 진행하는 경우, 앱브라우저의 user-agent 를 확인하여 P\_RESERVED 에 app\_scheme={앱스키마};// 형태를 설정하시면 해당 이슈가 해결 됩니다.

ex) 카카오톡 앱에서 결제 진행 시

신용카드	P_RESERVED=twotrs_isp=Y&block_isp=Y&app_scheme=kakaotalk://&
계좌이체	P_RESERVED=twotrs_bank=Y&iosapp=Y&app_scheme=kakaotalk://& P_RETURN_URL=kakaotalk://

## ◆ 확인된 App Scheme (2018.12.31 기준)

다음	daumapps://open
네이버	naversearchapp://
크롬	googlechromes://
페이스북	fb://
카카오톡	kakaotalk://

**주의사항 4) 삼성페이 연동 시 유의사항**

① 가맹점 앱에서 USER-AGENT 정보를 제어하고 있는 경우, 삼성페이가 정상적으로 실행되지 않을 수 있어, 당사에서는 가맹점 앱 내 USER-AGENT 변경을 권장하지 않고 있습니다.

② 삼성페이 앱스키마 ( samsungpay:// ) 에 대해 가맹점 앱에서 정상 실행할 수 있도록 구현되어 있어야 합니다. 해당 내용과 관련하여는 본 매뉴얼 1.A 내 ispmobile:// 에 대한 예제를 참고하여 samsungpay:// 스키마 호출 시에 대한 처리가 앱에서 지원되어야합니다.