**Wayup 인증결제 연동 가이드**

2022.04 WAYUP

ⓒ Copyright ⓒ 2022 WAYUP Co, Ltd. All rightys reserved

본 문서와 정보는 WAYUP 재산입니다. 모든 정보는 WAYUP 소유이며, WAYUP의 사전 동의 없이 본 문서의 어떤 정보도 열람, 복사, 유용, 또는 타인과 공유되어서는 안됩니다.

제,개정 이력

|  |  |  |
| --- | --- | --- |
| 0.1 | 제정 | 2022.04.25 |
| 0.1.5 | 인증결과 ordNo 삭제 | 2024.05.29 |
| 1.1 | 승인코드추가  전표URL추가 | 2024.10.15 |
| 1.2 | 상품명 길이 수정 | 2024.10.15 |
| 1.3 | 승인URL 변경 | 2025.05.13 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# 연동 개요

1.1 개요.

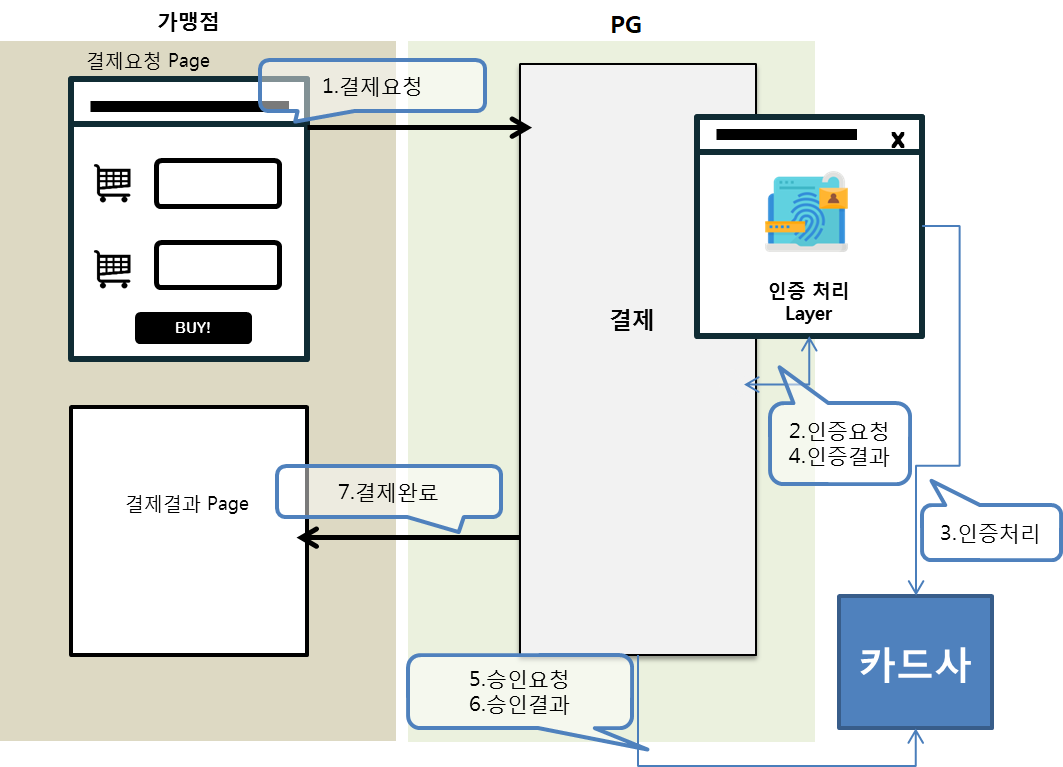
* 본 문서는 WAYUP의 지불결제 서비스를 이용하는 가맹점이 가맹점의 web 페이지에서 WAYUP와 결제 연동을 통해 WAYUP의 결제 서비스를 가맹점의 사용자에게 제공하는 방법을 제공합니다.

.

* 1. 특징.
* WAYUP는 사용자의 PC등에 별도의 모듈 설치(Non-ActiveX)가 없이 사용 가능합니다.
* SHA256을 통한 위변조 방지와 HTTPS(SSL) 통신을 이용하여 보안이 한층 더 강화된 전자지불 시스템을 제공합니다.
* 사용자의 PC 환경에 따라 자동으로 웹표준 결제창을 표시하며, ActiveX 등의 추가적인 설치 없이 Chrome, IE, Firefox, Safari 브라우저에서 사용 할 수 있습니다. (단, 카드사 등 금융기관에서 요구하는 경우에 사용자는 해당 모듈을 설치하여 결제를 이용해야 합니다.)
* 시스템 언어에 관계없이 HTTPS 통신만을 사용하여 쉽게 연동하여 사용할 수 있습니다.

1.3 서비스 구조와 결제 처리 (신용카드)

* 가맹점에서 SHA256을 이용하여 위변조 방지를 포함한 결제데이터를 생성합니다.
* 결제페이지에서 form 데이터를 WAYUP API 로 HTTPS(SSL) Post 액션을 발생시켜 결제요청 데이터로 보냅니다.
* WAYUP는 결제요청을 받아 사용자의 디바이스에 따른 결제창을 표시하고 사용자가 결제(인증/승인)를 진행할 수 있도록 합니다.
* 승인이 완료되면 리턴된 승인결과를 가맹점에 표시합니다



# 설치

2.1 요구사항

* 하드웨어 요구사항 : 특별한 하드웨어 요구사항은 없습니다.
* Web server : SHA256 암호화값의 생성 / httpClient(http Background) 통신이 가능한 웹서버
* DBMS : WAYUP에서는 결제 결과 DATA를 제공해 드리며 DB처리는 가맹점에서 관리하셔야 합니다. 이러한 경우에는 별도의 DBMS가 필요합니다.

2.2 방화벽 설정

결제승인, 결제취소에 대한 통신이 가능하도록 방화벽 설정이 필요합니다.

* 결제승인
  + - 연결대상 : https://api.wayup.co.kr
    - 프로토콜 : HTTPS
    - 연결방향 : INBOUND, OUTBOUND
* 결제취소
* 연결대상 : https://api.wayup.co.kr
* 프로토콜: HTTPS
* 연결방향 : INBOUND, OUTBOUND

2.3 샘플소스

* 샘플소스는 가맹점의 환경에 맞도록 수정하여 사용할 수 있습니다.
* WAYUP API 는 표준 웹 통신만을 사용합니다.
* 결제요청시에는 페이지이동(Form POST Action), API 통신시에는 httpClient 통신을 이용
* HTTPS API Request(httpClient 통신)= httpClient 등의 http Background 통신이 가능한 유틸을 통해서 웹페이지를 요청후 그 결과를 수신
* 샘플 소스 명세

|  |  |  |
| --- | --- | --- |
| 주요 연동 파일 | paySample.jsp | 결제 테스트 파일 |
| payResultSample.jsp | 결제 테스트 승인 및 결과 파일 |
| cancelSample.jsp | 결제취소 테스트 파일 |
| cancelResultSample.jsp | 결제취소 테스트 취소 및 결과 파일 |

* 1. 승인테스트
* 웹 브라우저에서 http://127.0.0.1/paySample.jsp 페이지를 호출합니다.

(localhost에서 진행할 경우 브라우저에 따라 팝업창의 문제가 있을 수 있습니다.)

* 승인 테스트를 위해 필수 값 및 기본 옵션 값을 매뉴얼을 참조 후 확인 후 결제 요청 버튼을 누릅니다.
* 설치 과정 없이 지불 방법을 선택 후 필요한 정보를 기입하고 “확인”을 눌러 진행합니다. (해당 사이트 팝업을 사용으로 하지 않을 시 초기화면으로 로딩 될 수 있습니다. )
* 지불 과정중 필요에 따라 추가적인 모듈을 다운로드 하는 경우도 있습니다. 정상적으로 지불결과 메시지가 나타나는지 확인하십시오.
* **취소 테스트를 위해 거래번호(TID)를 기록해 두십시오.**

# 3. 결제연동

3.1 결제 요청 페이지 작성

3.1.1결제 요청 데이터

* 내부 처리용 데이터 별도 저장
* form 데이터 처리가 필요한 경우 페이지 표시전에 별도로 DB 또는 세션 등에 저장해 두시기 바랍니다.
* merchantKey 값 설정
* 웹 표준 결제 이용시 패스워드 기능을 하는 코드로 MID 번호별로 부여

// 가맹점에 제공된 웹 표준 키(샘플예)

String merchantKey = " BoBwBC4hRuMxAztw9p85L7K+SB7Iswd1tdRwca7xQ2sFftC5nYAFgYkOctQ1ubHzACV0YzaWHJdqWAGZW34kPw==";

* 가맹점 Key 발급방법 : “가맹점정보>>일반정보>>KEY관리” 에서 확인가능
  + 가맹점 Key는 외부에 노출되지 않도록 주의합니다.
* Input 에 결제 요청 데이터 생성
  + Input tag에 name 설정(결제요청 스크립트 실행시 사용됨)
  + 필드명 대소문자 구분

(일부 가맹점에서 필요에 의해 사용자가 변경하는 경우를 제외하고 모두 type=”hidden”을 사용)

<div id=" sampleInput">

<input type=”hidden” name=”mid” value=” demotest0m”/>

<input type=”hidden” name=”price” value=”1004”/>

………..

</div>

**WAYUP는 Form Post 로 결제 요청되며, <form> 태그에 action 속성 설정 / submit 등의 모든 동작은 Import 된 스크립트에 의해서 자동 처리됩니다**.

* 결제 “인증” 요청데이터 필드는 아래와 같습니다. **(샘플: paySample.jsp)**

[TABLE] 기본 요청데이터 필드

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| HTTPS POST content-type : application/x-www-form-urlencoded; charset=utf-8 | | | | | |
| 필드명 | 한글명칭 | 예시[Data Type] | 설명 | 필수  여부 | 크기 (최대) |
| payMethod | 결제 방법 | [String]"" | 결제 방법  (card) | Yes | 10  Byte |
| mid | 상점아이디 | [String] "demotest0m" | 제공된 mid,\* 10자리 고정 | Yes | 10  Byte Fixed |
| trxCd | 에스크로 | [String] “0” | 에스크로 여부(0:고정) | Yes | 1  Byte |
| goodsNm | 상품명 | [String] "컴퓨터" | 상품 이름 | Yes | 50  Byte |
| ordNo | 주문번호 | [String] “1335233672723” | 주문번호 Unique한 | Yes | 40  Byte |
| goodsAmt | 결제금액 | [String] 1004 | 숫자만 입력 | Yes | 15  Byte |
| ordNm | 구매자명 | [String] "홍길동" | 한글/특수기호 입력가능 | Yes | 30  Byte |
| ordTel | 구매자  Mobile | [String] ”01020001234 | 숫자만 허용 | Yes | 20  Byte |
| ordEmail | 구매자  Email | [String] "[buyer@example.com](mailto:buyer@example.com)" | 이메일 형식에 맞도록 |  | 60  Byte |
| userIp | 구매자  아이피 | [String] ”127.0.0.1” or “0:0:0:0:0:0:0:1” | 유저 접속 아이피(IPv4, IPv6 지원) |  | 20  Byte |
| mbsReserved | 상점예약 | [String] "["](mailto:parent@example.com) |  |  | 500  Byte |
| goodsSplAmt | 공급가 | [String] 1004 | 숫자만 입력  대상: ‘전송금액기준’ 설정업체에 한함 |  | 15  Byte |
| goodsVat | 부가세 | [String] 1004 | 숫자만 입력  대상: ‘전송금액기준’ 설정업체에 한함 |  | 15  Byte |
| goodsSvsAmt | 봉사료 | [String] 1004 | 숫자만 입력  대상: ‘전송금액기준’ 설정업체에 한함 |  | 15  Byte |
| channel | 접속타입 | [String] “0001” | 0001:WEB, 0002:mobile, 0003:SMS, 0004: URL |  | 4  Byte |
| period | 제공기간 | [String] “” | default:별도제공기간없음 |  | 21  Byte |
| ediDate | 전문요청일시 | [String] “20210327121212” | 전문요청일시(yyyymmddhhmmss) | Yes | 14  Byte |
| encData | 해시String | [String] “” | Hash256 암호화 | Yes |  |
| notiUrl | NOTI Url | [String] "” | Back으로 결제완료 데이터를 받을 URL |  | N/A |
| charset | 인코딩방식 | [String] “UTF-8” | UTF-8(default), EUC-KR |  | 6  Byte |

**returnUrl :** **샘플기준 payResultSample.jsp 이 실제 결제 “승인” 처리되는 프로그램입니다.**

3.1.2 인증 결과 수신

•결제창을 통해 인증이 완료되면 결과를 가맹점으로 전달 합니다. 인증 결과 데이터 필드는 아래와 같습니다.

[TABLE] 기본 응답데이터 필드

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 필드명 | 한글명칭 | Data Type | 설명 | 크기 (최대) |
| resultCode | 결과코드 | String | [0000 : 정상, 기타 : 실패] | 4 Byte |
| resultMsg | 결과메세지 | String | 성공시 : OK, 실패시 : 기타 오류 메시지 | 100 Byte |
| payMethod | 지불수단 | String | 결제 방법 | 10 Byte |
| tid | 거래번호 | String | 거래 번호 | 30 Byte |
| ediDate | 결제일시 | String | 결제일시 | 14 Byte |
| mid | 상점 아이디 | String | 상점 아이피 | 20 Byte |
| goodsAmt | 거래금액 | String | 결제 금액 | 15 Byte |
| mbsReserved | 상점 예약필드 | String |  | 255 Byte |
| signData | 암호화데이터 | String | 결제 정보 암호화 데이터 |  |

3.1.3 승인 요청**(샘플기준 payResultSample.jsp)**

•결제창을 통해 받은 인증 결과로 승인 요청을 보냅니다. 승인 요청 데이터 필드는 아래와 같습니다.

[TABLE] 승인 요청데이터 필드

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| URL : https://api.wayup.co.kr/pay.do  HTTPS POST content-type : application/x-www-form-urlencoded; charset=utf-8 | | | | |
| 필드명 | 한글명칭 | Data Type | 설명 | 크기 (최대) |
| tid | 거래번호 | String | 거래 번호 | 30 Byte |
| ediDate | 결제일시 | String | 결제일시 | 14 Byte |
| mid | 상점 아이디 | String | 상점 아이피 | 20 Byte |
| goodsAmt | 거래금액 | String | 결제 금액 | 15 Byte |
| charSet | 인코딩 방식 | String |  |  |
| encData | Hash 데이터 | String | Hash256 암호화 |  |
| signData | 암호화데이터 | String | 결제 정보 암호화 데이터 |  |

3.1.4 승인 결과 수신 (결제완료)

•승인이 완료되면 결과를 가맹점으로 전달 합니다. 결제 결과 데이터 필드는 아래와 같습니다.

[TABLE] 결제 승인 기본 응답데이터 필드

|  |  |  |  |
| --- | --- | --- | --- |
| 필드명 | 한글명칭 | 설명 | 크기 (최대) |
| resultCd | 결과코드 | 0000, 3001 : 정상, 기타 : 실패 | 4 Byte |
| resultMsg | 결과메세지 | 성공시 : OK, 실패시 : 기타 오류 메시지 | 100 Byte |
| payMethod | 지불수단 | 결제 방법 | 10 Byte |
| tid | 거래번호 | 거래 번호 | 30 Byte |
| appDtm | 결제일시 | 결제일시 | 14 Byte |
| appNo | 승인번호 | 승인번호 | 20 Byte |
| ordNo | 주문번호 | 주문번호 | 40 Byte |
| goodsName | 결제 상품명 | 결제 상품명 | 50 Byte |
| amt | 거래금액 | 결제결과 금액 | 15 Byte |
| ordNm | 결제자 이름 | 결제자 이름 | 30 Byte |
| fnNm | 제휴사명 | 제휴사명 | 20 Byte |
| cancelYN | 취소여부 | 결제 취소 여부 | 1 Byte |
| appCardCd | 카드 발급사코드 | 카드 발급사코드 | 2 Byte |
| acqCardCd | 카드 매입사코드 | 카드 매입사코드 | 2 Byte |
| quota | 카드 할부기간. | 카드 할부기간. | 2 Byte |
| usePointAmt | 사용 포인트 양 | 사용 포인트 양 | 12 Byte |
| cardType | 카드타입 | 0:신용, 1:체크 | 1 Byte |
| authType | 인증타입 | 01:Keyin, 02:ISP, 03:VISA | 2 Byte |
| mid | 상점아이디 | 제공된 mid | 10 Bytes |
| ediDate | 전문요청일시 | 전문요청일시(yyyymmddhhmmss) | 14 Bytes |
| mbsReserved | 상점예약 | 상점예약필드 | 500 Bytes |
| charSet | 인코딩방식 | UTF-8(default), EUC-KR | 6 Bytes |

* 승인 요청 후 결제결과(responseBody)를 받아 내부처리(DB 저장 등) 하시기 바랍니다.
* 수신시 전송 필드명을 명확히하여 처리하시기 바랍니다. (필드명 대/소문자 구분)
* 승인에 실패하였을 경우 실패시 전달되는 데이터만 전송됩니다.

3.2 결제 취소 요청 페이지 작성

3.2.1결제 취소 요청 데이터

* 내부 처리용 데이터 별도 저장
* form 데이터 처리가 필요한 경우 페이지 표시전에 별도로 DB 또는 세션 등에 저장해 두시기 바랍니다.
* merchandkey 값 설정
* 웹 표준 결제 이용시 패스워드 기능을 하는 코드로 MID 번호별로 부여

// 가맹점에 제공된 웹 표준 키

String merchantKey = " BoBwBC4hRuMxAztw9p85L7K+SB7Iswd1tdRwca7xQ2sFftC5nYAFgYkOctQ1ubHzACV0YzaWHJdqWAGZW34kPw==";

* 가맹점 Key 발급방법 : “가맹점정보>>일반정보>>KEY관리” 에서 확인가능
  + 가맹점 Key는 외부에 노출되지 않도록 주의합니다.
* Input 에 결제 요청 데이터 생성
  + Input tag에 name 설정(결제요청 스크립트 실행시 사용됨)
  + 필드명 대소문자 구분

(일부 가맹점에서 필요에 의해 사용자가 변경하는 경우를 제외하고 모두 type=”hidden”을 사용)

<div id=" sampleInput">

<input type=”hidden” name=”mid” value=”demotest0m”/>

<input type=”hidden” name=”price” value=”1004”/>

………..

</div>

**WAYUP는 Form Post 로 결제 취소 요청되며, <form> 태그에 action 속성 설정 / submit 등의 모든 동작은 Import 된 스크립트에 의해서 자동 처리됩니다**.

* 결제 취소 기본 요청데이터 필드는 아래와 같습니다.

[TABLE] 기본 취소요청데이터 필드

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| URL : https://api.wayup.co.kr/payment.cancel  HTTPS POST content-type : application/x-www-form-urlencoded; charset=utf-8 | | | | | |
| 필드명 | 한글명칭 | 예시  [Data Type], 기본값 | 설명 | 필수  여부 | 크기 (최대) |
| tid | 거래번호 | [String] | 거래 번호 | Yes | 30  Byte |
| ordNo | 주문번호 | [String]  “1335233672723” | 주문번호 |  | 40  Byte |
| canAmt | 취소금액 | [String] 1004 | 숫자만 입력  1달러는 100으로 시작 | Yes | 64  Byte |
| canId | 취소자ID | [String] "testId" | 한글/특수기호 입력가능 |  | 10  Byte |
| canNm | 취소자명 | [String] “홍길동” | 취소자 이름 |  | 30  Byte |
| canMsg | 취소사유 | [String] “고객요청” | 취소사유 | Yes | 100  Byte |
| canIp | 취소 아이피 | [String] ”127.0.0.1” or “0:0:0:0:0:0:0:1” | 취소 아이피(IPv4, IPv6 지원) |  | 20  Byte |
| partCanFlg | 부분취소여부 | [String] ”0 | 0:전체취소, 1:부분취소 | Yes | 20  Byte |
| notiUrl | NOTI Url | [String] "” | 결체 취소 요청을 해서 Back으로 결과를 받을 URL |  | N/A |
| encData | 해시String | [String] “” |  | Yes | 256  Byte |
| ediDate | 전문요청일시 | [String] “20210324121212” | 전문요청일시(yyyymmddhhmmss) | Yes | 14  Byte |
| goodsSplAmt | 공급가 | [String] 1004 | 숫자만 입력  대상: ‘전송금액기준’ 설정업체에 한함 |  | 15  Byte |
| goodsVat | 부가세 | [String] 1004 | 숫자만 입력  대상: ‘전송금액기준’ 설정업체에 한함 |  | 15  Byte |
| goodsSvsAmt | 상품 봉사료 | [String] 1004 | 숫자만 입력  대상: ‘전송금액기준’ 설정업체에 한함 |  | 15  Byte |
| charset | 인코딩방식 | [String] “UTF-8” | UTF-8(default), EUC-KR |  | 6  Byte |

3.2.2 결제취소 결과 수신 (결제취소완료)

•결제 취소가 완료되면 결과를 가맹점으로 전달 합니다. 취소 결과 데이터 필드는 아래와 같습니다.

[TABLE] 결제취소 기본 응답데이터 필드

|  |  |  |  |
| --- | --- | --- | --- |
| 필드명 | 한글명칭 | 설명 | 크기 (최대) |
| resultCd | 결과코드 | [0000 : 정상, 기타 : 실패] | 4 Byte |
| resultMsg | 결과메세지 | 성공시 : OK, 실패시 : 기타 오류 메시지 | 100 Byte |
| payMethod | 지불수단 | 결제 방법 | 10 Byte |
| tid | 거래번호 | 거래 번호 | 30 Byte |
| appDtm | 결제일시 | 결제일시 | 14 Byte |
| appNo | 승인번호 | 승인번호 | 20 Byte |
| ordNo | 주문번호 | 주문번호 | 40 Byte |
| amt | 거래금액 | 결제결과 금액 | 15 Byte |
| cancelYN | 취소여부 | 결제 취소 여부 | 1 Byte |
| oTid | 원거래번호 | 원거래 번호 | 30 Byte |
| ediDate | 전문요청일시 | 전문요청일시(yyyymmddhhmmss) | 14 Bytes |
| mid | 상점아이디 | 제공된 mid | 10 Bytes |
| charset | 인코딩방식 | UTF-8(default), EUC-KR | 6 Bytes |

* 결제 취소 요청 후 취소결과(responseBody)를 받아 내부처리(DB 저장 등) 하시기 바랍니다.
* 수신시 전송 필드명을 명확히하여 처리하시기 바랍니다. (필드명 대/소문자 구분)
* 결제취소에 실패하였을 경우 실패시 전달되는 데이터만 전송됩니다.

# 4. 앱 환경의 적용(안드로이드)

4.1 기본적인 설치 방법

* + 안드로이드 앱 내 WebView (이하 WebView) 에서 WAYUP를 구현하는 경우에 해당됩니다.

4.2 모바일 ISP 연동 (-앱 미설치 체크로직 직접구현 or 자동체크)

* + mobileISP 앱의 기본정보는 아래와 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.
  + WebViewClient를 상속받은 클래스를 구현하시고, shouldOverrideUrlLoading() 을 호출

private class SampleWebView extends WebViewClient { @Override

public boolean shouldOverrideUrlLoading(WebView view, String url) {

…. ]

* + 상기 shouldOverrideUrlLoading() 함수 내에, try{} catch{e} 를 통해, try 내에서는 startActivity(intent) 를 구현하시고, catch Event 발생 시, 앱 스토어로 이동.

[shouldOverrideUrlLoading 부]

private class SampleWebView 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 prefix가 ispmobile 일겨우만 alert를 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 AlertDialo g.Builder(PaymentVie w.this)

.setIcon(android.R.drawable.ic\_dialog\_a lert)

. 알림

. 모바일 ISP 어플리케이션이 설치되어 있지 않습니다 . n 설치를

눌러 진행 해 주십시요 n 취소를 누르면 결제가 취소 됩니다 .")

. 설치 ", new DialogInterface.OnClickListener() {

@Override

public v oid onClick(DialogInterface dialog, int which) {

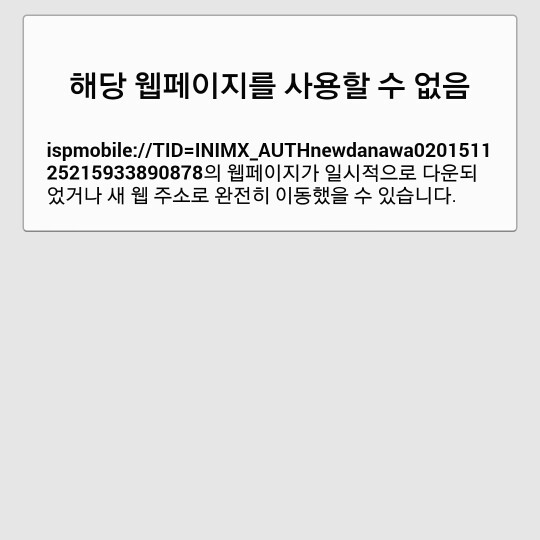
34 / 48 Pub. Date: 2017. 09

|  |
| --- |
| //ISP 설치페이지URL  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)결제를취소하셨습니다." , Toast.LENGTH\_SHORT).show();  finish();  }  }).create();  ...  } |

* + mobileISP 가 단말기에 기 설치되어 있는 경우, mobileISP 가 정상구동
  + mobileISP 가 단말기에 미 설치되어 있는 경우, 설치 후, WAYUP를 다시 띄워주 시면 됩니다. 앞 페이지의 예시[shouldOverrideUrlLoading 부] 의 음영 에 대하여 true 혹은 false 를 설정하는 것은 하기의 표를 참고하십시요.

|  |  |  |  |
| --- | --- | --- | --- |
| apprun\_check | 작동방식 | 앱 미설치 시, 앱스토어 이동 후,결제페이지 | t/f |
| Y | ISP, 계좌이체앱  - intent 작동 | 상태 유지 | true |
| N or 미설정 | ISP, 계좌이체앱  – appScheme 작동 | 하기 그림과 같이  Display 됨 | false |

* shouldOverrideUrlLoading 부] 의 음영 을 true 로 할 경우, WAYUP를 띄운 WebView 는 사라집니다. 따라서, app Scheme 형태로 결제 앱을 호출 할 경우에는 아래와 같이 오류 페이지가 Display 되기 때문에, WebView 를 remove 하는 것이 좋습니 다.



* [shouldOverrideUrlLoading 부] 의 음영을 false 로 할 경우, 결제창 을 띄운 WebView 는 사라지지 않기 때문에, apprun\_check=Y 를 통해 현 결제 페이지가 유지되 는 방식을 사용 하는 것이 좋습니다. 이 방법을 자동체크방식이라 합니다. 단, apprun\_check 옵션을 통해 설치체크로직이 작동되므로, alertIsp 함수는 구현될 필요가 없 습니다. apprun\_check 로직에 대하여 상세히 확인하시려면 ( 결제창 Open (주문정보 전달) – 복합필드 ) 를 확인하여 주십시오. 또한, Intent 호출에 대 하여 예외처리를 반드시 체크하셔야 합니다.

4.3 모바일 ISP 연동 (인증결과 전송)

* ISP 앱에서 인증과정이 완료되면, 다시 WAYUP 결제창으로 돌아와서 ‘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.

4.4 안심클릭 결제 시, 카드사 백신 앱 연동

* BC 계열을 제외한, 나머지 카드사의 결제창을 iframe 내에서 운용하고 있 습니다. 이에, 카드사에서 개별적으로 사용하는 백신 앱의 경우, 가맹점 앱에서도 하기의 유 의사항을 반드시 체크하셔야 합니다.
* WebView 내에서 http와 https URL, 그리고 App Url 을 분기하여 처리합니다. ( shouldOverrideUrlLoading() 처리로직을 하기와 같이 구현 )

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

* 상기의 유의사항을 고려한 샘플 코드는 하기와 같습니다

private class SampleWebViewClient extends WebViewClient { @Override

public boolean shouldOverrideUrlLoading(WebView view, String url) { Log.d("<TEST>","URL : "+url);

/\*

\* URL별로 분기가 필요합니다. 어플리케이션을 로딩하는것과

\* WEB PAGE를 로딩하는것을 분리 하여 처리해야 합니다.

\* 만일 가맹점 특정 어플 URL이 들어온다면

\* 조건을 더 추가하여 처리해 주십시요.

\*/

if( !url.startsWith("http://") && !url.startsWith("https://") && !url.startsWith("javascript:") )

{

Intent intent; try{

intent = Intent.parseUri(url, Intent.URI\_INTENT\_SCHEME); Log.d("<TEST>", "intent getDataString : " + intent.getDataString());

} catch (URISyntaxException ex) {

Log.e("<TEST>", "URI syntax error : " + url + ":" + ex.getMessage()); return false;

}

try{

startActivity(intent);

}catch(ActivityNotFoundException e){

/\* ISP어플이 현재 폰에 없다면 아래 처리에서

\* 알림을 통해 처리하도록 하였습니다

* 삼성카드 및 기타 안심클릭에서는
* 카드사 웹페이지에서 알아서 처리하기때문에
* WEBVIEW에서는 별다른 처리를 하지 않아도 처리됩니다.

\*/

if( url.startsWith("ispmobile://"))

{

//onCreateDialog에서 정의한 ISP 어플리케이션 알럿을 띄워줍니다..URI\_INTENT\_SCHEME); String packageNm = excepIntent.getPackage();

Log.d("<MOBILE>", "excepIntent getPackage : " + packageNm );

excepIntent = new Intent(Intent.ACTION\_VIEW); excepIntent.setData(Uri.parse("market://search?q="+packageNm));

startActivity(excepIntent);

} catch (URISyntaxException e1) {

Log.e("<MOBILE>", "INTENT:// 인입될시 예외 처리 오류 :

" + e1 );

}

}

}

}

else

{

view.loadUrl(url); return false;

}

return true;

}

}

//(ISP 어플리케이션이 없을경우) showDialog(DIALOG\_ISP);

return false;

}else if( url.startWith("intent://")){ //intent 형태의 스키마 처리

try {

Intent excepIntent = Intent.parseUri(url, Intent.URI\_INTENT\_SCHEME); String packageNm = excepIntent.getPackage();

Log.d("<MOBILE>", "excepIntent getPackage : " + packageNm );

excepIntent = new Intent(Intent.ACTION\_VIEW); excepIntent.setData(Uri.parse("market://search?q="+packageNm));

startActivity(excepIntent);

} catch (URISyntaxException e1) {

Log.e("<MOBILE>", "INTENT:// 인입될시 예외 처리 오류 :

" + e1 );

}

}

}

}

else

{

view.loadUrl(url); return false;

}

return true;

}

}

4.5 공인인증 앱 연동 방법

* 안드로이드의 경우, 개별 카드사 앱에서 공인인증서 서명을 할 수 있습니다. 이에, 카드사 창 내에서 호출하는 intent 혹은 app Scheme 를 허용할 수 있도 록 가맹점 앱에서 처리해줘야 합니다. 이는 (안심클릭 결제 시, 카드사 백신 앱 연동) 의 코드를 반영하면 해결됩니다.

4.6 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 경 우, 페이지가 호출되지 않아 인증결과가 전달되지 않을 수 있습니다. 하기의 설정을 확인하십 시오.

|  |  |
| --- | --- |
|  | 코드 |
| Insecurity 페이지 차단 | WebSettings web = paymentView.getSettings();  web.setMixedContentMode(web.MIXED\_CONTENT\_NEVER\_ALLOW); |
| Insecurity 페이지 허용 | WebSettings web = paymentView.getSettings();  web.setMixedContentMode(web.MIXED\_CONTENT\_ALWAYS\_ALLOW); |

* 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 | 코드 |
| hird party cookies 허용 | CookieManager cookieManager = CookieManager.getInstance(); cookieManager.setAcceptCookie(true); cookieManager.setAcceptThirdPartyCookies(sampleWebView,  true); // false 설정 시 오류 발생 |

# 5. 앱 환경의 적용(IOS)

5.1 기본적인 설치방법

* IOS 어플리케이션 내 WebView (이하 WebView) 에서 WAYUP를 구현하는 방법을 안냉합니다.
* WebView 에서 WAYUP를 띄우는 방식은 안드로이드와 방법과 동일합니다. 이에 이번 장 에서는 mobileISP 앱 호출시 주의사항, 카드사 백신 앱 스키마 호출 및 “미설치 시, 앱스토어 이동 이슈” 등의 내용을 주로 다룹니다.

5.2 mobileISP 연동방법

* mobileISP 앱이 종료 된 뒤, 가맹점 앱을 다시 띄우기 위한 조치사항을 안내합니다.
* IOS 는 Android 계열과 다르게도 mobileISP 이 종료된 뒤, 가맹점 앱은 Background 에 머문 채, 바탕화면이 개제됩니다. (IOS의 운영체제 특성에 기반) 이 때문에, mobileISP 앱이 종료되 면서, 가맹점 appScheme 을 호출하도록 구성해야 합니다. 하기와 같이 셋팅 시, 요구사항과 같이 가맹점 앱이 다시 기동됩니다.

|  |  |
| --- | --- |
| P\_RESERVED | &app\_scheme=가맹점스키마명:// |

* 가맹점스키마명 뒤 :// 은 필수로 입력해주셔야 mobileISP 앱 종료 후 가맹점 앱이 호출 됩니다. (Ex. 가맹점 스키마명이 merchant-app 일 경우 app\_scheme= merchant-app:// 로 셋팅해 주시면 됩니다.)

5.3 안심클릭 결제 시, 카드사 백신 앱 연동

* + IOS 환경에서는 카드사에서 별도로 백신을 구동하지 않습니다.

5.4 카드사 앱 연동방법

* + 안심클릭 결제 진행에 필요한 Application (앱카드 등의) 호출이 필요할 경우 아래 샘플코드 를 참고 바랍니다. (해당 샘플은 참고용으로 각 가맹점앱에 맞게 구현하시면 됩니다.)
* WAYUP의 WebView는 IOS 4.X이상의 버전에서 작동하지만 카드사 등의 보안정책 등으로 OS버전에 따른 결제 제약이 있을 수 있습니다.

#pragma mark UIWebViewDelegate

-(BOOL)webView:(UIWebView \*)webView shouldStartLoadWithRequest:(NSURLRequest \*)request navigationType:(UIWebViewNavigationType)navigationType

{

//쿠키 강제 허용

NSHTTPCookieStorage \*cookieSto = [NSHTTPCookieStorage sharedHTTPCookieStorage]; [cookieSto setCookieAcceptPolicy:NSHTTPCookieAcceptPolicyAlways];

//KISPG를 통해 전달되는 URL

NSString \*URLString = [NSString stringWithString:[request.URL absoluteString]];

//URL 을 읽어왔을때 애플 스토어 주소인경우 사파리에 해당 URL 을 넘겨서 앱스토어에서 설치 할수 있도록 유도

BOOL isStoreURL = ([URLString rangeOfString:@"[phobos.apple.com](http://phobos.apple.com/)" options:NSCaseInsensitiveSearch].location != NSNotFound);

BOOL isStoreURL2 = ([URLString rangeOfString:@"[itunes.apple.com](http://itunes.apple.com/)" options:NSCaseInsensitiveSearch].location != NSNotFound);

//앱스토어 이동

if (isStoreURL || isStoreURL2) {

[[UIApplication sharedApplication]openURL:request.URL]; return NO;

}

else if([URLString hasPrefix:@"http"] || [URLString hasPrefix:@"https"] || [URLString hasPrefix:@"about"] ) //일반적인 웹 url 형태인 경우 진행

{

return YES;

}

else{ //그외의 값은 앱스키마로 간주하여 앱 호출

NSURL \*appURL = [NSURL URLWithString:URLString]; //NSString to NSURL

//앱스키마인 경우 앱호출

BOOL bAppScheme = [[UIApplication sharedApplication] canOpenURL:appURL];

if (!bAppScheme) {

//앱이 설치되지 않은 경우 앱스토어로 이동 또는 안내 얼럿 표출 return NO;

}

}

return YES;

}

5.5 쿠키설정

* IOS WebView 에서 호출하고, 안심클릭 계열 서비스를 사용하는 경우, 세 션만료 오류경고가 발생할 수 있습니다. 이에, 하기의 샘플과 같이 쿠키를 허용해야 합니다.

(BOOL)application:(UIApplication \*)application didFinishLaunchingWithOptions:(NSDictionary \*)launchOptions

{

[[NSHTTPCookieStorage sharedHTTPCookieStorage] setCookieAcceptPolicy:NSHTTPCookieAcceptPolicyAlways];

...

return YES;

}

5.6 IOS9버전 Application 구현 시, 주의 사항

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

# 6. 모바일 환경 적용 시 유의사항

6.1네이버/카카오톡 앱 환경 최적화 issue>

* 네이버/카카오 앱에서는 해당 앱 특성에 따라, 당사 결제창을 띄울 때, 새 창(\_blank)을 띄울 경우, 정상적으로 결제가 진행되지 않을 수 있습니다. 따라서 가맹점 플랫폼에서, 당사 결제 창을 띄울 때, 새창이 아닌, \_self 형태로 띄워주시길 권장합니다.

6.2 안드로이드/IOS 앱 USER AGENT 수정

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

6.3 아이폰 3rd party(제3 공급자) 앱브라우저 결제 issue

* iOS 기본 브라우저인 Safari 외 앱브라우저(다음, 네이버, 크롬, 페이스북 등) 에서 ISP 결제 진행 시, ISP 앱에서 인증 완료 후 결제를 진행한 브라우저 앱이 아닌 Safari 브라우저로 돌아 가는 이슈가 있습니다. 브라우저 앱에서 결제를 진행하는 경우, 앱브라우저의 user-agent를 확인하여 P\_RESERVED에 app\_scheme={앱스키마}:// 형태를 설정하시면 해당 이슈가 해결 됩니다
* 확인된 App Scheme

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

# 7. 암호화

* 1. 결제 승인 시 암호화 구성

|  |  |  |
| --- | --- | --- |
| 파라미터 | 암호화 방식 |  |
| encData | Hex(Sha256) | mid + ediDate + goodsAmt + 상점키(merchantKey) |

|  |
| --- |
| String mid = “demotest0m”;  String ediDate = “20210713151115”;  String goodsAmt = “1004”;  String merchantKey = “BoBwBC4hRuMxAztw9p85L7K+SB7Iswd1tdRwca7xQ2sFftC5nYAFgYkOctQ1ubHzACV0YzaWHJdqWAGZW34kPw==”  암호화값 : 4684f4b322ae388e2e4c9a60309c66a472db61140e5cb8da085b1bd6dbb6f0c1 |

* 1. 결제 취소 시 암호화 구성

|  |  |  |
| --- | --- | --- |
| 파라미터 | 암호화 방식 |  |
| encData | Hex(Sha256) | mid + ediDate + canAmt + 상점키(merchantKey) |

|  |
| --- |
| String mid = “demotest0m”;  String ediDate = “20210713151115”;  String canAmt = “1004”;  String merchantKey = “BoBwBC4hRuMxAztw9p85L7K+SB7Iswd1tdRwca7xQ2sFftC5nYAFgYkOctQ1ubHzACV0YzaWHJdqWAGZW34kPw==”  암호화값 : 4684f4b322ae388e2e4c9a60309c66a472db61140e5cb8da085b1bd6dbb6f0c1 |

# 8. 결과 코드

* 1. 주의사항
* 제공 되는 결과 코드에 기재되어 있지 않은 결과 코드는 파트너 업체의 코드를 그대로 전달하고 있습니다. 해당 사항에 포함되는 건은 개인 정보 확인이 필요 할 수 있으며, 사용자(결제자)가 파트너 업체(카드사, 통신사, 은행)를 통해 직접 확인이 필요할 수 있습니다. 필요 시, help@wayup.co.kr 로 TID, ordNo, MID, ediDate를 전달 바랍니다.
  1. 카드 결과 코드

|  |  |
| --- | --- |
| 코드 | 메시지 |
| 0000  또는  3001 | 카드 결제 성공 |
| 3002 | 카드 결제 실패 |
| 3011 | 카드번호 오류 |
| 3012 | 카드가맹점 정보 미확인 |
| 3013 | 카드가맹점 개시 안됨 |
| 3014 | 카드가맹점 정보 오류 |
| 3015 | 사용 중지된 가맹점입니다. |
| 3021 | 유효기간 오류 |
| 3022 | 할부개월 오류 |
| 3023 | 할부개월 한도 초과 |
| 3024 | 50000원 미만은 할부가 불가능 합니다. |
| 3025 | 카드비밀번호 미입력 |
| 3026 | 본인인증 필수값 누락 |
| 3027 | 카드인징 필수값 누락 |
| 3031 | 무이자 할부 카드 아님 |
| 3032 | 무이자 할부 불가 개월수 |
| 3033 | 무이자 할부 가맹점 아님 |
| 3034 | 무이자 할부 구분 미설정 |
| 3041 | 금액 오류(1000원 미만 신용카드 승인 불가) |
| 3051 | 해외카드 미등록 가맹점 |
| 3052 | 통화코드 오류 |
| 3053 | 확인 불가 해외카드 |
| 3054 | 환율전환오류 |
| 3055 | 인증시 달러승인 불가 |
| 3056 | 국내카드 달러승인불가 |
| 3057 | 인증 불가카드 |
| 3061 | 국민카드 인터넷 안전 결제 적용 가맹점 |
| 3062 | 신용카드 승인번호 오류 |
| 3071 | 매입요청 가맹점 아님 |
| 3072 | 매입요청 TID 정보 불일치 |
| 3073 | 기매입 거래 |
| 3081 | 카드 잔액 값 오류 |
| 3091 | 제휴카드 사용불가 가맹점 |
| 3095 | 카드사 실패 응답 |
| 3096 | 외화승인불가VAN |
| 3097 | 외화승인설정오류 |
| 3098 | 해외카드결제불가 |
| 3099 | 해외카드는 할부 결제가 불가합니다. |
| 3100 | 세이브결제 부분취소 불가 |
| 3101 | 타상점 거래취소 불가 |
| 3102 | 거래금액이 입금완료 상태가 아님 |
| 3103 | 부분취소 불가능 가드번호 |
| 3104 | 부분취소 불가능 카드사 가맹점 번호 |
| 3105 | 장바구니 결제 유형 설정 오류 |
| 3106 | 장바구니 서브 거래내역 없음 |

* 1. 취소 결과 코드

|  |  |
| --- | --- |
| 코드 | 메시지 |
| 0000  또는  2001 | 취소 성공 |
| 2002 | 취소 진행중 |
| 2003 | 취소 실패 |
| 2004 | 이미 취소된 거래 |
| 2010 | 취소 요청금액 0원 이하 |
| 2011 | 취소 금액 불일치 |
| 2012 | 취소 해당거래 없음 |
| 2013 | 취소 완료 거래 |
| 2014 | 취소 불가능 거래 |
| 2015 | 기 취소 요청 |
| 2016 | 취소 기한 초과 |
| 2017 | 취소 불가 가맹점 |
| 2018 | 신용카드 매입후 취소 불가능 가맹점 |
| 2019 | 타 가맹점 거래 취소 불가 |
| 2020 | 망상취소 허용시간 초과 |
| 2021 | 매입전취소 |
| 2022 | 매입후취소 |
| 2023 | 취소 한도 초과 |
| 2024 | 취소패스워드 불일치 |
| 2025 | 취소패스워드 미 입력 |
| 2026 | 입금액보다 취소금액이 큽니다. |
| 2027 | 에스크로 거래는 구매 또는 구매거절 상태에만 취소가능 |
| 2028 | 부분취소 불가능 가맹점 |
| 2029 | 부분취소 불가능 결제수단 |
| 2030 | 해당결제수단 부분취소 불가 |
| 2031 | 전체금액취소 불가 |
| 2032 | 취소금액이 취소가능금액보다 큼 |
| 2033 | 부분취소 불가능금액, 전체취소 이용바람 |
| 2034 | 해당거래 부분취소 불가능. 매입완료 후 가능 |
| 2035 | 채권방지 설정으로 취소불가. PG사로 문의바람 |
| 2036 | 카드정보 불일치 |
| 2037 | 취소요청금액과 장바구니 취소금액 불일치 |
| 2038 | 장바구니 거래로 취소요청 바람(일반취소 불가) |
| 2039 | 일반거래로 취소요청 바람(장바구니 취소 불가) |
| 2040 | 장바구니 취소 정보가 없음 |
| 2041 | 익월 취소 불가능한 거래입니다. |
| 2042 | 에스크로 거래는 구매 상태에만 환불 가능 |

# 9. 카드사 코드

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 코드 | 카드사 |  | 코드 | 카드사 |
| 01 | 비씨 |  | 27 | 해외다이너스 |
| 02 | 국민 |  | 28 | 해외AMAX |
| 03 | 하나 |  | 29 | 해외JCB |
| 04 | 삼성 |  | 30 | 해외 |
| 06 | 신한 |  | 32 | 우체국 |
| 07 | 현대 |  | 33 | MG새마을체크 |
| 08 | 롯데 |  | 34 | 중국은행체크 |
| 11 | 시티 |  | 38 | 은련 |
| 12 | NH농협 |  | 41 | 신협 |
| 13 | 수협 |  | 42 | 저축은행 |
| 15 | 우리 |  | 43 | KDB산업 |
| 21 | 광주 |  | 44 | 카카오뱅크 |
| 22 | 전북 |  | 45 | 케이뱅크 |
| 23 | 제주 |  | 46 | 카카오머니 |
| 25 | 해외비자 |  | 47 | 강원 |
| 26 | 해외마스터 |  | 48 | UNIONPAY |

# 10. 신용카드 전표 URL

# https://api.wayup.co.kr/receipt?tid={tid}&appNo={appNo}

|  |  |  |  |
| --- | --- | --- | --- |
| 필드명 | 한글명칭 | 설명 | 크기 (최대) |
| tid | 거래번호 | 거래 번호 | 30 Byte |
| appNo | 승인번호 | 승인번호 | 20 Byte |