EventServer API 명세서

Thank you for choosing CASTIS.

This document is the CASTIS EventServer

Interface Specificastions.



Revision Date

2015-03-04

# 문서 작성 이력

|  |  |  |
| --- | --- | --- |
| Version | Description | Date |
| 0.0.19 | * SubscriberEnrollDetail   + couponInfo 값 추가 | 2015-03-04 |
| 0.0.18 | * DoEnrollPorcessByService   + 요청 파라메터 price 필수 값 삭제 | 2015-03-02 |
| 0.0.17 | * getReservationInfo API 추가 * doEnrollProcessByPurchase   + 예약정보 구하는 기능 추가   + EnrollServer로 예약 API 호출 추가 * DoEnrollPorcessByService   + 예약정보 구하는 기능 추가   + EnrollServer로 예약 API 호출 추가 * doManualEnroll   + doManualEnrollStart에서 이름 변경   + 예약된 정보 확인기능 추가   + EnrollServer로 예약 confirm API호출로 변경 * PortalMenu 정보   + listSize 추가 * getWinnerList API   + 페이징 관련 반환 정보 추가 | 2015-02-27 |
| 0.0.16 | * getAvailableEventList API 추가 * getEventActionTargetList API 추가 * getWinnerList API 추가 * getEventInfo   + eventActionTargetLIst API 추가로 응답에서 삭제   + winnerList API 추가로 응답에서 삭제   + 구매/시청 관련 요청 변수 삭제   + 당첨자 보기 버튼 활성화 여부 추가 * doEnrollProcessByService   + resultCount 삭제 * doEnrollProcessByPurchase   + resultCount 삭제 | 2015-02-25 |
| 0.0.15 | * Event 정보   + profile 1 에 당첨자발표일(winnerAnnouncedTime) 항목 추가 * SubscriberWinInfo 정보   + Phone 항목 추가. * targetId의 “productId|goodId” 형식 중 “goodId” 없을 경우 “productId”만 사용하도록 변경   + EventViewTarget 정보> targetId   + EventActionTarget 정보 > targetId   + getEventListForViewTarget API > targetId   + getEventListForViewTargetPopup API > targeted * getEventListForActionTarget API   + 처리 내용의 정렬처리 설명 삭제 * getEnrolledEventListForActionTarget API   + API 명칭 변경 🡪 getEnrolledEventListForAction * doManualEnrollStart API   + 처리내용의 참여대상정보 및 참여조건 변경. 요청변수에 구매/시청 정보의 동시 입력 가능하도록   + 요청변수에 price 항목 추가 * doEnrollProcessByPurchase API   + 요청변수에 price 항목 추가 * doEnrollProcessByService API   + 요청변수에 price 항목 추가 * getMyEventList API   + 처리내용 중 불필요한 설명 삭제   + 지원 프로파일을 2,3로 변경 * getEventInfo API   + 처리내용 중     - 이벤트 대상 목록 정보 처리설명 수정     - 연관 이벤트 목록 처리 내용 삭제     - 기타 불필요한 처리 내용 설명 삭제   + 응답 항목의 evetActionTargetList 항목 설명 중 실행대상의 라이선스 체크 내용 추가 | 2015-02-22 |
| 0.0.14 | * doPrizeReceipt API   + 처리내용을 중복당첨 시를 고려하여 수정 * doWinConfirm API   + 처리내용을 중복당첨 시를 고려하여 수정 | 2015-02-11 |
| 0.0.13 | * getEventListForActionTarget API   + 프로파일 1 지원하도록 변경   + 요청 파라미터에 imagePosition 항목 추가 * getEventListForViewTargetPopup API   + 요청 파라미터 imagePosition 항목에 코드 4 추가 | 2015-02-10 |
| 0.0.12 | * doManualEnrollStart API   + 시청/구매 이벤트에 따른 처리 내용 변경   + 중복참여 조건 변경   + 요청변수 필수 값 변경 * doEnrollProcessByPurchase API   + 이벤트 상태, 자동참여 여부에 따른 참여가능 이벤트 조건 설명 수정 * doEnrollProcessByService API   + 이벤트 상태, 자동참여 여부에 따른 참여가능 이벤트 조건 설명 수정 * getPortalMenuEventList API   + portalMenuId 요청 변수 필수 입력 값으로 변경 * getPortalMenuNoticeList API   + portalMenuId에 대한 type 체크 내용 삭제 | 2015-02-05 |
| 0.0.11 | * 연동 규격 및 규칙 명세 추가 * API 개발 시 주의 사항 명세 추가 * 주요 정보 변경 사항   + EventActionTarget     - Title, price 항목 추가   + RelatedEvent     - 프로파일 삭제   + PortalMenu 정보 추가   + Notice 정보 추가 * API 변경 사항   + getEventListForViewTarget API     - 처리내용의 정렬 기준 설명 문구 삭제     - 요청변수 targetId 설명 내용 중 productType관련 설명 문구 삭제     - API 설계 배경 설명문구 추가.   + getEvemtForViewTargetPopup API     - assetId, productId, goodId, productType 항목 삭제     - targetType, targetId 항목 추가     - targetId 항목의 설명 내용 중 productType관련 설명 문구 삭제     - imagePosition 코드 변경 및 기본값 변경.   + getEventListForActionTarget API     - 요청변수에서 productType 삭제     - goodId 항목의 설명 내용 중 productType관련 설명 문구 삭제   + doEnrollStart API 삭제   + doEnrollProcess API 삭제   + doManualEnrollStart API 추가     - 요청변수에 productType 추가     - 수동참여 이벤트인지 확인하는 처리내용 추가     - purchaseDomain, serviceDomain 요청 파라미터 설명 추가   + doEnrollProcessByPurchase API 추가     - 요청변수에 productType 추가     - purchaseDomain 요청 파라미터 설명 추가   + doEnrollProcessByService API 추가     - serviceDomain 요청 파라미터 설명 추가   + getPortalMenuList API 추가   + getPortalNoticeList API 추가     - 공지사항 조회 기간 설명 수정.   + getMyEventList API     - 처리내용 > 우선순위 설명 추가     - 처리내용 > 각 이벤트 별 포함 할 정보 설명 삭제     - 요청 변수의 soId, stbModelName 항목 삭제     - Profile 기본값 3으로 변경     - 요청 변수의 sortType에 대한 설명 문구 삭제   + getEventInfoById 🡪 getEventInfo 로 API 명 변경     - 처리 내용 중 연관이벤트 목록, 개인정보, 참여정보, 당첨자목록, 당첨관련 내용 설명 삭제     - 요청 변수의 profile 기본값 3으로 변경     - 요청변수에서 productType 삭제     - 응답 항목의 ralatedEventList 삭제   + getRelatedEventListForActionTarget API 추가   + getUncheckedWinEventList API 삭제   + getUncheckedWinEventCount API 추가   + doPrizeReceipt API     - 요청상황, 처리내용 설명 수정     - 요청 변수에 phone 항목 추가   + doWinConfirm API 추가   + getEnrollEventCount API 삭제   + getEventList API     - 처리내용 > 정렬 기준 내용 수정     - 요청 변수       * autoEnroll 삭제       * 지원 profile에서 1 삭제       * sortType중 id, createTime, displayTime 관련 정렬 기준 삭제 | 2015-02-04 |
| 0.0.10 | * 요청변수 중 에셋, 상품 정보 동시에 받는 경우 처리방법 추가 설명   + getEventForViewTargetPopup   + getEventListForActionTarget   + getEnrolledEventListForActionTarget   + doEnrollStart   + doEnrollProcess   + getEventInfoById * getEventInfoById   + 구매정보, 시청정보를 요청변수에서 삭제 | 2015-01-28 |
| 0.0.9 | * Event 정보의 일부를 분리하여 SubscriberEnrollInfo, SubscriberWinInfo 정보 추가 * EnrollDetail 정보 이름을 SubscriberEnrollDetail로 변경 * RelatedEvent 정보 추가 * getEventList API 추가 * 2.12.1 이벤트 진행에 따른 상태에서 발표대기 상태 삭제 * Event 정보 변경   + 노출기간, 진행기간 값에서 초 삭제.(분 까지만 표시)   + Status 에서 60:발표대기 상태 삭제   + creationTime, lastModifyTime 추가   + imageUrl 설명 수정.(요청 없을 경우 기본이미지로 상세화면 이미지 적용)   + closeBtnRedirectCategoryId 프로파일을 1 🡪 0으로 변경   + drawReservationTime 프로파일 변경 2 🡪 1, 값을 분단위까지만 표현   + isCollectPhoneNumberForEnroll 프로파일 변경 2 🡪 1   + drawExecutionTime 프로파일 변경 2 🡪 1, 값의 날짜 형식 지정   + 기존의 참여 정보 항목 삭제하고 SubscriberEnrollInfo 정보 추가   + 기존의 당첨 정보 항목 삭제하고 SubscriberWinInfo 정보 추가 * SubscriberEnrollDetail 정보 변경   + goodId 값 없는 경우 처리 방법 명시   + enrollTime 값의 날짜 형식 지정 * EventViewTarget 정보   + eventList 의 지원 프로파일을 0 만 지원하도록 변경 * getEventListForActionTarget   + goodId, productType 값의 상황에 따른 필수 여부 추가 * doEnrollStart   + eventId 타입을 Long 🡪 Int 로 변경 * doEnrollProcess   + 일부 항목의 타입을 String 으로 지정   + goodId, productType 값의 상황에 따른 필수 여부 추가   + productType 코드 명시   + purchaseTime, serviceTime 의 형식 지정   + 요청 항목의 purchaseTime, serviceTime 항목 삭제 * getPortalEventList   + 지원 프로파일을 0,1,2,3 으로 변경 * getEventInfoById   + 요청변수의 eventId 타입을 Long 🡪 Int로 변경   + goodId, productType 값의 상황에 따른 필수 여부 추가   + 응답항목에 subscriberEnrollDetailList, eventActionTargetList, relatedEventList, winnerList 추가 | 2015-01-23 |
| 0.0.8 | * EnrollDetail 정보   + goodId 값 없는 경우 처리 방법 명시 * EventActionTarget 정보   + goodId 값 없는 경우 처리 방법 명시 * EventViewTarget 정보   + goodId 값 없는 경우 처리 방법 명시 * getEventListForViewTarget   + 상세팝업 정보 관련 내용 삭제   + 처리조건 중 가입자 내용 삭제하고 SO, STB조건만 반영   + 참여이벤트 옵션 조건 삭제   + 요청 변수     - subscriberId, enrollIncluded 항목 삭제     - imagePosition 값 없을 경우 설명 추가     - targetId 값 중 goodId 값 없는 경우 처리 방법 명시 * getEventForViewTargetPopup API 추가 * getEventListForActionTarget   + 요청 변수     - goodId 에 대한 설명 및 필수조건 변경     - productType 종류 명시 * doEnrollProcess   + 요청 변수     - productType 종류 명시     - goodId 값이 없는 경우 처리 방법 명시 * getPortalEventList   + 용어 변경. “카테고리” 🡪 “이벤트포털 메뉴”   + 처리 내용 중 이미지 정보 내용 추가   + 정렬 기준 설명 수정   + 로그 기록 내용 삭제   + 요청 변수     - categotyId 🡪 portalMenuId 로 변경 및 설명 변경 * getMyEventList   + 처리 조건에 포함 이미지 정보 내용 추가   + 로그 기록 내용 삭제 * getEventInfoById   + 처리 내용 중 이미지 정보 내용 추가   + 로그 기록 내용 삭제   + 요청 변수     - productType 에 상품 종류 명시     - goodId 값이 없는 경우 처리 방법 명시 | 2015-01-21 |
| 0.0.7 | * 요청변수의 타입이 bool인 경우 입력값을 숫자형이 아닌 true/false 값으로 변경 * 응답항목의 errorString 타입 Boolean 🡪 String 으로 변경 * 에셋아이디, 상품아이디 를 요청변수로 입력 받는 경우 assetId, productId 둘 중 하나는 필수 값으로 변경. productType 값 필수입력으로 요청 변수 추가   + 관련 API     - getEventListForActionTarget     - doEnrollProcess     - getEventInfoById | 2015-01-16 |
| 0.0.6 | * 2. 주요 용어 정리 추가 * 3.1 Event 정보   + 설명 및 프로파일 설명 추가.   + 각 항목 이름, 타입, 설명, 프로파일 변경 * 3.2 EnrollDetail 정보   + targetType 종류 대문자로 변경   + enrollStatus 타입 Int형으로 변경, 상태 코드 수정 * Target 정보 삭제 * 3.3 EventActionTarget 정보 추가 * 3.4 EventViewTarget 정보 추가 * 3.5 Image 정보 삭제 * 3.6 Winner 정보의 name, phone 값의 \*표 처리 설명 추가. * 4.1 getEventListForObject 🡪 getEventListForViewTarget 으로 API 명 변경   + 설명 수정   + 요청 상황 수정   + 기본 처리 조건 수정   + 옵션 조건 수정   + 제외조건 삭제   + 정렬기준 수정   + 요청변수 이름, 타입, 설명 필수 여부, 기본값 등 수정   + 응답 항목 항목 삭제 및 수정 * 4.2 getEventListForAction 🡪 getEventListForActionTarget 으로 API 명 변경   + 기본 처리 조건 수정   + 정렬 기준 수정   + 요청변수 이름, 타입, 설명 필수 여부, 기본값 등 수정   + 응답 항목 항목 삭제 및 수정 * 4.3 getEnrollCompletedEventListForActoin 🡪 getEnrolledEventListForActionTarget 로 API 명 변경   + 기본 처리 조건 수정   + 제외 조건 삭제   + 요청변수 이름, 타입, 설명 필수 여부, 기본값 등 수정 * getServiceEventList 🡪 getPortalEventList 로 API 명 변경   + 처리 내용 수정   + 요청변수 및 응답항목 수정 | 2015-01-14 |
| 0.0.5 | * 이벤트 정보의 프로파일 변경 및 설명 추가 * getEventListForObject API   + 요청변수 중 enrollIncluded 항목 삭제   + API의 처리 내용 중 참여조건 포함여부 항목 삭제 | 2015-01-12 |
| 0.0.4 | * 모든 API 요청변수 중 아래 항목 수정   + externalSoId -> soId   + exposureGroupId -> stbModelName   + subscriberDomain -> 삭제   + subscriberId 설명 수정   + externalSubscriberId -> 삭제 * 3.1 Event 정보 변경   + ui\_event\_type -> uiEventType   + enrollCount 항목의 profile 2 -> 1로 변경   + enrollActionType 추가   + enrollCompleteCount 추가   + enrollCompletePrice 추가   + draw\_time -> drawExecutionTime   + is\_prize\_confirm -> prizeReceiptStatus   + is\_collect\_phone\_number\_for\_enroll -> isCollectPhoneNumberForEnroll   + phone\_number -> phoneNumber   + prize\_type -> prizeType. 설명의 type 중 tv\_coin -> tvcoin   + prize\_detail -> prizeDetail   + is\_collect\_phone\_number\_for\_prize -> isCollectPhoneNumberForPrize   + draw\_reservation\_time -> drawReservationTime   + enroll\_detail -> enrollDetail   + winner\_display -> winnerDisplay * 3.2. Enroll\_Detail -> EnrollDetail 로 이름 변경 * 3.2 EnrollDetail 정보 항목 수정   + target\_type -> targetType   + target\_id -> 삭제   + assetId 추가   + productId 추가   + goodId 추가   + target\_name -> targetName   + enroll\_time -> enrollTime   + target\_price -> price * 4.1. getEventListForObject API의 처리 내용 추가   + 상품 소진 시 이벤트 종료 관련 항목 추가   + 4.1.2 요청 주소 샘플 추가   + 요청 변수 변경     - targetList 삭제     - targetTypeList 추가     - targetIdList 추가   + 4.1.5 요청과 응답 예 추가 * 4.2. getEventListForAction API의 처리 내용 추가   + 상품 소진 시 이벤트 종료 관련 항목 추가   + 4.2.2 요청 주소 샘플 추가   + 4.2.5 요청과 응답 예 추가 * 4.5. doEnrollProcess API의 처리 내용 추가   + 상품 소진 시 이벤트 종료 관련 항목 추가 * 4.7 getMyEventList 처리 내용에 방문로그 처리 추가 * 4.8 getEventInfoById 처리 내용에 방문로그 처리 추가 * 4.10 doPrizeReceipt API 추가 * 4.11 getEnrollEventCount API 추가 | 2015-01-08 |
| 0.0.3 | * event 정보 변경   + enroll\_detail 추가   + draw\_reservation\_time 추가   + draw\_time 추가   + winner\_display 추가   + status 에 31:상품소진 상태 추가   + prize --> prize\_type로 변경   + prize\_detail 추가   + prize\_collect\_phone\_number 추가   + is\_prize\_confirm 추가 * enroll\_detail 정보 추가 * getEventListForObject, getEventListForAction API 설명 추가 * getEventListForObject API 처리내용의 D항목 삭제 * getEnrollCompletedEventListForAction API의 처리내용에 제외 조건 추가 * getServiceEventList API에 category 요청변수 추가 * getMyEvent API 정렬 조건 변경 * getEventInfoById API의 참여정보 상세 내용 추가, 연관된 다른 이벤트 목록 정렬 순서 설명 추가 * 참여 이벤트 수 조회 API 추가 * 공지사항 관련 API 삭제 | 2015-01-07 |
| 0.0.2 | * 주요 정보 및 프로파일 추가 * 전체 API 수정 | 2014-12-22 |
| 0.0.1 | * 초안 작성 | 2014-12-22 |

# 목 차

[문서 작성 이력 1](#_Toc412472463)

[목 차 11](#_Toc412472464)

[1. 개요 13](#_Toc412472465)

[2. 연동 규격 및 규칙 14](#_Toc412472466)

[2.1. 요청 Method 14](#_Toc412472467)

[2.2. 응답 Data Type 14](#_Toc412472468)

[2.3. 한글 Encoding 방식 14](#_Toc412472469)

[2.4. 공통 파라미터 설명 14](#_Toc412472470)

[2.4.1. 공통 요청 파라미터 14](#_Toc412472471)

[2.4.2. 공통 응답 파라미터 14](#_Toc412472472)

[3. API 개발 시 주의 사항 16](#_Toc412472473)

[3.1. 대소문자 구분 규칙 16](#_Toc412472474)

[3.2. Validation 처리 규칙 16](#_Toc412472475)

[3.3. 각 API 별 공통 처리 16](#_Toc412472476)

[3.4. 외부 연동 시 resultCodt 처리 16](#_Toc412472477)

[3.5. 시간 비교 16](#_Toc412472478)

[3.6. 라이선스 체크 17](#_Toc412472479)

[4. 주요 용어 정리 18](#_Toc412472480)

[4.1. 가입자(SubscriberId) 18](#_Toc412472481)

[4.2. 지역 아이디(SoId) 18](#_Toc412472482)

[4.3. 이벤트 표현 대상(EventViewTarget) 18](#_Toc412472483)

[4.4. 이벤트 실행 대상(EventActionTarget) 18](#_Toc412472484)

[4.5. 이벤트 노출 대상 가입자 18](#_Toc412472485)

[4.6. 연관 이벤트(RelatedEvent) 19](#_Toc412472486)

[4.7. 이벤트 타입 19](#_Toc412472487)

[4.8. 응모(Enroll), 참여의사표시, 응모완료(EnrollComplete) 19](#_Toc412472488)

[4.9. 추첨(Draw) 20](#_Toc412472489)

[4.10. 당첨(Win), 당첨자(Winner) 20](#_Toc412472490)

[4.11. 혜택(Prize), 즉시지급혜택, 수령확인(PrizeReceipt), 수령취소 20](#_Toc412472491)

[4.12. 이벤트 상태 21](#_Toc412472492)

[4.12.1. 이벤트 진행에 따른 상태(Event Status) 21](#_Toc412472493)

[4.12.2. 이벤트 참여 상태(Enroll Status) 21](#_Toc412472494)

[5. 주요 정보 23](#_Toc412472495)

[5.1. Event 23](#_Toc412472496)

[5.2. SubscriberEnrollInfo 25](#_Toc412472497)

[5.3. SubscriberWinInfo 25](#_Toc412472498)

[5.4. SubscriberEnrollDetail 26](#_Toc412472499)

[5.5. EventViewTarget 26](#_Toc412472500)

[5.6. EventActionTarget 27](#_Toc412472501)

[5.7. RelatedEvent 28](#_Toc412472502)

[5.8. Winner 29](#_Toc412472503)

[5.9. PortalMenu 29](#_Toc412472504)

[5.10. Notice 29](#_Toc412472505)

[6. 기능 및 인터페이스 목록 30](#_Toc412472506)

[6.1. 이벤트 아이콘, 상세팝업 관련 API 30](#_Toc412472507)

[6.1.1. getEventListForViewTarget 30](#_Toc412472508)

[6.1.2. getEventForViewTargetPopup 32](#_Toc412472509)

[6.2. Toast팝업, 시청종료화면 관련 API 35](#_Toc412472510)

[6.2.1. getEventListForActionTarget 35](#_Toc412472511)

[6.2.2. ~~getEnrolledEventListForActionTarget~~ 🡪 getEnrolledEventListForAction 38](#_Toc412472512)

[6.3. 이벤트 참여 관련 API 39](#_Toc412472513)

[~~6.3.1.~~ ~~doEnrollStart~~ 39](#_Toc412472514)

[~~6.3.2.~~ ~~doEnrollProcess~~ 39](#_Toc412472515)

[6.3.3. doManualEnrollStart 39](#_Toc412472516)

[6.3.4. doEnrollProcessByPurchase 42](#_Toc412472517)

[6.3.5. doEnrollProcessByService 44](#_Toc412472518)

[6.4. 이벤트 포털 관련 API 47](#_Toc412472519)

[6.4.1. getPortalMenuList 47](#_Toc412472520)

[6.4.2. getPortalEventList 48](#_Toc412472521)

[6.4.3. getPortalNoticeList 50](#_Toc412472522)

[6.4.4. getMyEventList 51](#_Toc412472523)

[6.4.5. getEventInfo 53](#_Toc412472524)

[6.4.6. getRelatedEventListForActionTarget 56](#_Toc412472525)

[~~6.4.7.~~ ~~getUncheckedWinEventList~~ 58](#_Toc412472526)

[6.4.8. getUncheckedWinEventCount 58](#_Toc412472527)

[6.5. 당첨자 관련 API 59](#_Toc412472528)

[6.5.1. doPrizeReceipt 59](#_Toc412472529)

[6.5.2. doWinConfirm 61](#_Toc412472530)

[~~6.5.3.~~ ~~getEnrollEventCount~~ 62](#_Toc412472531)

[6.6. 외부 연동을 위한 일반 이벤트 API 62](#_Toc412472532)

[6.6.1. getEventList 62](#_Toc412472533)

[7. 결과 코드 (Result Code) 67](#_Toc412472534)

# 개요

외부 시스템 혹은 클라이언트에 이벤트 관련 정보를 제공하고 이벤트 참여 및 당첨 관련 정보를 처리하는 API를 제공함

# 연동 규격 및 규칙

## 요청 Method

아래 방식의 요청(request)을 지원한다.

* GET
* POST

## 응답 Data Type

http 프로토콜을 통해 주고받는 메시지의 본문(body) 형식은 다음과 같다

* XML
* JSON

api\_name**.json** 요청 시 jsonString 으로 응답, api\_name**.xml** 요청 시 xml String 응답.

## 한글 Encoding 방식

입출력 값의 한글은 UTF-8로 인코딩하여 전송.

## 공통 파라미터 설명

### 공통 요청 파라미터

모든 요청(request)은 공통적으로 아래 파라미터(parameter)를 포함한다.

|  |  |  |
| --- | --- | --- |
| 이름 | 설명 | 필수 여부 |
|  |  |  |

### 공통 응답 파라미터

모든 응답(response)은 공통적으로 아래 세 가지의 파라미터(parameter)를 포함한다.

|  |  |
| --- | --- |
| 이름 | 설명 |
| resultCode | 요청에 대한 결과값. 별도로 정의한 숫자를 사용한다 |
| erroString | resultCode가 100(Success)가 아닌 경우 에러 내용을 명시한다 |

# API 개발 시 주의 사항

## 대소문자 구분 규칙

* 요청 파라미터 변수명은 대소문자 구분. camelCase(AaaBbbCcc) 형식.
* 요청 파라미터의 코드 값 등은 대소문자 상관없이 처리.

## Validation 처리 규칙

모든 입력 파라미터 값에 대하여 아래와 같은 validation 처리를 해야 함.

* 필수 입력 파라미트 누락 여부
* 숫자 형 파라미터의 입력 값이 숫자인지
* 미리 정의 된 코드의 경우 지원하는 코드 값인지
* 미리 정의된 형식이 있는 경우 형식에 맞는지

## 각 API 별 공통 처리

* Request, Response 에 대한 로그
* Error 로그
* API 별 처리 시간

## 외부 연동 시 resultCodt 처리

* 외부 컴포넌트와 연동 시 외부 컴포넌트에서 받은 결과 코드를 그대로 bypass 하지 말 것.
* 반드시 EventServer 내부의 결과코드로 매핑하여 처리해야 함.

## 시간 비교

* 시간을 비교하는 경우에는 반드시 서버의 시간을 기준으로 비교.
* DB 상의 시간 값을 시간으로 비교할 때도 서버의 시간을 DB로 넘겨서 비교 처리.
* 시간 비교 시 시간 값의 단위를 어디까지 비교할 것인지 주의 할 것.
  + 분 단위까지 비교 시 분단위 값 까지만 가지고 비교
* MySQL의 DateTime 형식은 초 단위 까지만 저장되며 밀리초(millisecond) 값까지 넣는 경우 반올림되어 초 단위가 변경되므로 주의 할 것.

## 라이선스 체크

* 특별한 비교 기준 시간이 없는 경우 항상 현재 서버시간을 기준으로 라이선스 체크
* 이벤트 조회 시 라이선스를 체크
  + 진행 중 여부와 상관 없는 경우는 displayTime 을 기준으로 체크
  + 진행 중 이벤트만 조회 해야 할 때는 actionTime 을 기준으로 체크
* ActionTarget (실행대상) 을 조회 시 라이선스 체크
  + ActionTarget 에셋과 상품의 licenseTime을 기준으로 체크

# 주요 용어 정리

## 가입자(SubscriberId)

가입자는 이벤트에 참여하는 STB 단위의 가입자를 기준으로 한다.

가입자를 구분하는 아이디는 subscriberId 이며 실제 값은 EPG/쿠폰/xPUSH 등에서는 CAS\_ID 값, VOD 에서는 customerId 값에 해당한다.

## 지역 아이디(SoId)

so id는 가입자가 속한 so 지역을 구분하기 위한 아이디이다. VOD 백오피스에서는 externalSoId 라는 값이 이에 해당한다. EPG, 쿠폰 시스템에서는 soId가 이에 해당한다.

## 이벤트 표현 대상(EventViewTarget)

이벤트 표현 대상은 이벤트가 있는지를 표현할 대상을 말하며, 이벤트 매니저에서 대상으로 직접 설정한 에셋, 카테고리, 상품을 말한다.

A카테고리를 이벤트 대상으로 지정한 경우 이벤트 아이콘은 A카테고리 자체에 표시되며, 상세팝업도 A카테고리 진입 시 볼 수 있다. A카테고리가 이벤트 표현 대상이다. A카테고리 내에 있는 에셋, 상품, 카테고리 등은 표현 대상 아니므로 이벤트 아이콘이나 상세팝업이 없다.

## 이벤트 실행 대상(EventActionTarget)

이벤트 실행 대상은 이벤트에 참여할 수 있는 실제 대상 에셋, 상품을 말하며, 이벤트 매니저에서 대상으로 설정한 에셋, 카테고리, 상품 정보를 이용하여 실제로 참여할 수 있는 에셋, 상품 정보로 변환해 놓는다. 또한 여기서 말하는 에셋은 FOD/RVOD만을 대상으로 하며, FOD/RVOD 정보 없이 다른 상품에만 연결된 에셋(예를 들어 패키지 상품에만 연결된 에셋)은 이벤트 실행 대상에서 제외된다.

A카테고리 안에 에셋 001, 002, 003 가 있는 경우 이벤트에 A카테고리를 대상으로 설정하면 이벤트 표현대상은 A카테고리 이지만, 이벤트 실행 대상은 카테고리 내에 있는 에셋 001, 002, 003이 된다.

## 이벤트 노출 대상 가입자

이벤트 노출 대상 가입자는 이벤트 정보를 볼 수 있는 가입자를 뜻한다. 어떤 컨텐츠를 대상으로 이벤트를 진행하더라도 이벤트 노출 대상 가입자에게만 해당 컨텐츠의 이벤트 정보가 제공되고 참여 가능하다. 현재는 (특정 SO + 특정 STB 모델) 조합으로 이벤트 노출 대상 가입자를 설정 한다.

향후에는 구매/시청/가입일 정보 등의 보다 다양한 조건을 이용하여 노출 대상을 구분할 수 있는 기능을 제공할 예정이다.

## 연관 이벤트(RelatedEvent)

특정 컨텐츠(에셋/상품/카테고리)가 이벤트 대상으로 설정된 다른 이벤트를 말한다. 에셋A가 이벤트 E1, E2, E3에 이벤트 대상으로 설정된 경우 에셋A의 연관 이벤트는 E1, E2, E3가 된다.

## 이벤트 타입

* 경품 이벤트
  + 경품 이벤트로 설정한 이벤트. 1회의 응모만으로 참여 완료되어 추첨 대상이 되는 이벤트.
* **목표달성형 이벤트**  
  2회 이상 응모 행위를 통하여 목표를 달성해야 추첨 대상이 되는 이벤트. 목표달성형 이벤트는 아래와 같은 이벤트 타입이 있다.
  + 스탬프 : 정해진 수 만큼 스탬프를 찍어야 참여 완료되며 혜택이 지급되는 이벤트
  + N+1 : N번 참여 시 참여완료 되며 +1의 혜택이 지급 되는 이벤트
  + 합산페이백 : 정해진 금액만큼 구매를 해야 참여 완료되며, 할인쿠폰이나 TV코인등으로 페이백이 지급되는 이벤트
* 자동/수동 이벤트
  + 자동 이벤트 : 별도의 참여의사 표시 없이 이벤트 대상을 구매/시청만 하면 바로 참여되는 이벤트. 경품이벤트 중 자동 참여로 설정된 이벤트
  + 수동 이벤트 : 이벤트 대상의 구매/시청과는 별도로 이벤트에 참여하겠다고 직접 의사를 밝혀야 하는 이벤트. 구매/시청 전이나 구매/시청 후 시청종료 즉시 이벤트포털을 통하여 참여의사를 밝혀야 참여가 된다. 경품 이벤트 중 수동 참여로 설정된 이벤트, 목표달성형 이벤트가 이에 해당한다.

## 응모(Enroll), 참여의사표시, 응모완료(EnrollComplete)

* 응모/참여(Enroll)
  + 응모, 참여 용어는 동일한 의미로 사용한다.
  + 응모/참여는 이벤트에 응모/참여 하는 행위 자체를 뜻한다. 경품 이벤트의 경우 1회 응모만 가능하지만, 향후 목표달성형 이벤트의 경우 여러 번 응모하는 행위가 가능하다.
* 참여의사표시
  + 참여의사표시는 수동 이벤트(경품 수동 이벤트, 목표달성형 이벤트)에 대하여 가입자가 이벤트에 참여하겠다고 의사를 밝힌 상태를 말한다. 반드시 이벤트포털 상세화면에서 참여하는 행위를 직접 해야만 한다.
  + 응모완료/참여완료 응모를 통해 당첨 대상에 포함된 상태를 말한다. 경품 이벤트의 경우 1회 응모만으로 응모완료가 되며, 목표달성형 이벤트의 경우 목표치를 채웠을 때 응모완료가 된다.
  + 위 내용으로 볼 때 참여자는 참여의사표시만 한 참여자, 1회 응모를 했지만 목표를 채우지 못한 참여중인 참여자, 목표를 모두 채운 참여 완료한 참여자가 있으며, 참여 완료한 가입자만 추첨대상이 된다.
* 응모완료/참여완료
  + 응모완료는 경품응모 이벤트의 1회의 응모만으로 응모완료가 되며, 목표달성형 이벤트에서는 목표 수치를 달성한 경우에 응모완료가 된다.

## 추첨(Draw)

추첨은 응모완료 한 가입자 중에 혜택(상품)을 받을 가입자를 선정하는 행위를 말한다.

실시간 추첨은 응모완료가 되는 즉시 혜택을 받을지 여부가 결정되는 방식이다.

일괄 추첨은 이벤트 진행이 종료 된 이후 모든 참여자를 대상으로 한꺼번에 혜택 받을 가입자를 선정하는 방식이다.

수동 추첨은 이벤트 진행이 종료 된 이후 운영자가 직접 혜택 받을 가입자를 선정하는 방식이다.

## 당첨(Win), 당첨자(Winner)

당첨은 추첨을 통해 혜택을 받을 대상으로 선정된 상태를 말한다.

당첨된 가입자를 당첨자라 한다.

## 혜택(Prize), 즉시지급혜택, 수령확인(PrizeReceipt), 수령취소

혜택(상품, 경품)은 당첨자에게 제공할 선물을 말한다. TV코인, 할인쿠폰, 혹은 실물경품(상품권, 여행권 혹은 자전거, TV 등)을 제공할 수 있다.

즉시지급 혜택은 TV코인, 할인쿠폰 등 당첨되면 즉시 제공받은 혜택을 말한다. 즉시지급 혜택은 수령확인이 별도로 필요하지 않아도 지급된다.

수령확인은 당첨자가 혜택 내용을 확인하고 받겠다고 의사를 밝히는 것을 말한다.

수령취소는 혜택 받을 내용을 확인하지 않거나, 받기를 거부한 것을 말한다. 실물경품의 경우 제세공과금을 당첨자가 납부해야 하는데 이를 거부하는 경우 수령취소가 된다. 또는 수령확인이 필요한 혜택을 받는 당첨자가 수령확인 행위를 일정 기간 안에 하지 않으면 수령취소가 된다.

## 이벤트 상태

이벤트 상태 정보는 DB에서 모두 관리하지 않는다. 노출기간, 진행기간에 따른 상태는 현재 시간을 기준으로 판단해야 하며, 참여, 추첨, 발표, 상품소진, 진행중지, 당첨여부, 참여상태 등은 별도로 관리되는 각각의 상태에 따라 판단한다.

### 이벤트 진행에 따른 상태(Event Status)

* 진행 전 : 이벤트 노출 시작 시간 ~ 이벤트 진행 시작 시간
* 진행 중 : 이벤트 진행 시작 시간 ~ 이벤트 진행 종료 시간
* 마감임박 : 이벤트 진행 종료 시간으로부터 정해진 이전 시간 ~ 이벤트 진행 종료 시간
  + 마감임박 시간이 3일 이라면 (이벤트 진행 종료 시간 – 3일) ~ 이벤트 진행 종료 시간
* 진행중지 : 이벤트 진행 중에 운영자가 진행중지 기능을 통하여 중지시킨 상태이다. 노출기간 동안에는 이벤트포털에서 노출 된다.
* 추첨대기 : 추첨일이 설정되어 있는 경우 이벤트 진행 종료 시간이 지나면 추첨대기 상태가 된다. 추첨일이 미설정되어 있는 실시간 이벤트의 경우 진행 종료가 되면 자동으로 추첨 완료 상태가 된다. 당첨자들은 아직 당첨여부를 확인할 수 없다.
  + 추첨완료 : 모든 혜택에 대하여 추첨이 완료된 상태에서 상품수가 ALL인 혜택을 제외한 나머지 혜택에 대한 당첨자들의 개인정보가 모두 입력되지 않은 상태이다. 본인의 당첨여부 및 혜택을 확인할 수 있으나, 당첨자 목록은 조회할 수 없다.
* 상품소진완료 : 이벤트 등록 시 “참여 혜택 당첨 완료 시, 이벤트 종료”가 체크하고 실시간 추첨으로 진행되는 경우에 이벤트 진행 중에도 상품이 모두 소진 되는 상황이 생길 수 있다. 이 경우에는 상품소진 상태가 되며 이벤트가 마감되어 더 이상 참여 할 수 없다. 단 이벤트포털에서 노출은 가능하다.
* 발표대기 : 상품수가 ALL인 혜택을 제외한 나머지 혜택(상품수가 N개로 정해진)의 당첨자들에 대한 모든 개인정보가 입력이 완료된 상태. 아직 발표하지는 않은 상태이다.
* 발표완료 : 발표대기 상태에서 당첨자 발표버튼을 눌러서 발표한 상태. 이벤트포털 상세화면에서 당첨자목록 버튼을 통하여 당첨자 목록을 조회할 수 있다.

### 이벤트 참여 상태(Enroll Status)

* 미참여 : 이벤트에 참여하지 않음
* 참여 의사 표시 : 이벤트에 참여 의사만 밝히고 실제 참여 횟수는 0인 상태
* 참여 중 : 참여 의사를 밝힌 후 1회 이상 참여했으며 목표를 달성하지 못한 상태
* 참여 완료 : 목표를 달성한 상태.

# 주요 정보

이벤트 서버에서 제공하는 주요 정보

## Event

이벤트 정보는 프로파일을 지원한다. 프로파일은 각 API별로 혹은 동일 API에서도 상황에 따라 받고 싶은 이벤트 정보의 수준이 다른 경우에 어느 정도의 정보를 제공할지를 결정하는 요소이다. 프로파일에 따라 간단한 정보부터 전체 정보까지 조회 할 수 있다. 각 API별로 지원하는 프로파일 수준은 다를 수 있다.

프로파일은 0~2까지 총 3가지가 있으며 각 프로파일 별 용도는 아래와 같다.

* 0 : 이벤트 최소 정보 제공.
  + 이벤트 아이콘 조회, 재생 중 Toast 팝업, 시청종료 배너 등 최소한의 이벤트 정보만 조회하는 경우
* 1 : 이벤트 추가 정보 제공.
  + 이벤트 상세 팝업
* 2 : 가입자의 이벤트 참여 정보 제공.
  + 이벤트포털 목록 등
* 3 : 가입자의 이벤트 당첨 및 혜택 정보 제공
  + 이벤트포털 목록 등

|  |  |  |  |
| --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 프로파일 |
| eventId | Int | 이벤트 아이디 | 0 |
| title | String | 이벤트 타이틀 | 0 |
| description | String | 설명 | 0 |
| eventType | String | UI 이벤트 타입   * 0 : 경품응모 * 1 : 스탬프 * 2 : N+1 * 3 : 합산페이백 | 0 |
| isAutoEnroll | Bool | 자동참여 여부 | 0 |
| displayStartTime | String | 노출 시작 일시. yyyy-MM-dd HH:mm~~:ss~~ | 0 |
| displayEndTime | String | 노출 종료 일시. yyyy-MM-dd HH:mm~~:ss~~ | 0 |
| actionStartTime | String | 진행 시작 일시. yyyy-MM-dd HH:mm~~:ss~~ | 0 |
| actionEndTime | String | 진행 종료 일시. yyyy-MM-dd HH:mm~~:ss~~ | 0 |
| status | Int | 이벤트 상태   * 10 : 이벤트 전 * 20 : 이벤트 중 * 30 : 마감임박 D-3 * 40 : 추첨 대기(진행기간 종료 후 추첨 전까지) * 50 : 추첨 완료(ALL 이외 나머지 상품의 당첨자가 모두 결정 된 경우) * 51 : 상품소진완료(“참여 혜택 당첨 완료 시, 이벤트 종료“ 설정된 이벤트의 상품이 소진 된 경우) * ~~60 : 발표 대기(추첨 완료 후 ALL 이외 나머지 모든 당첨자 개인정보가 입력된 상태)~~ * 70 : 발표 완료(발표 버튼을 누른 경우) * ~~-10~~ 🡪 80 : 진행 중지 | 0 |
| creationTime | String | 이벤트 생성일. yyyy-MM-dd HH:mm:ss | 0 |
| lastModifyTime | String | 이벤트 최종 수정일. yyyy-MM-dd HH:mm:ss | 0 |
| imageUrl | String | 이미지 Url.   * 요청 변수로 입력 받은 position의 이미지 url * 요청 값 없을 경우 이벤트포털 상세이미지가 기본 값 | 1 |
| closeBtnRedirectCategoryId | String | 이벤트포털 상세화면에서 닫기 버튼을 선택할 때 이동할 카테고리 아이디 | 1 |
| enrollActionType | String | 참여 방법.   * 0 : 구매 * 1 : 시청 | 1 |
| enrollCompleteCount | Int | 목표 달성 기준 횟수 | 1 |
| enrollCompletePrice | Int | 목표 달성 기준 금액 | 1 |
| isCollectPhoneNumber ForEnroll | Bool | 전화번호 수집 여부. true/false | 1 |
| drawReservationTime | String | 추첨 예정일. yyyy-MM-dd HH:mm~~:ss~~ | 1 |
| drawExecutionTime | String | 추첨 실행일(추첨예정일 없는 경우는 실시간 추첨이므로 실행일 없음). yyyy-MM-dd HH:mm:ss | 1 |
| winnerAnnouncedTime | String | 당첨자 발표일. yyyy-MM-dd HH:mm | 1 |
|  |  |  |  |
| subscriberEnrollInfo | *SubscriberEnrollInfo* | 가입자의 참여 정보 | 2 |
|  |  |  |  |
| subscriber WinInfo | *SubscriberWinInfo* | 가입자의 당첨 및 혜택 정보 | 3 |
|  |  |  |  |

## SubscriberEnrollInfo

가입자의 참여 정보.

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| totalEnrollCount | Int | 총 참여 횟수 |
| totalEnrollPrice | Int | 총 참여 금액 |
| enrollStatus | String | 참여 상태   * 0 : 미참여 * 1 : 참여 의사 표시 * 2 : 참여 중 * 9 : 참여 완료 |
| lastEnrollTime | String | 마지막 참여 시간. yyyy-MM-dd HH:mm:ss |
| isWinner | Bool | 당첨 여부 |

## SubscriberWinInfo

가입자의 당첨 및 혜택 정보.

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| isCollectPhoneNumberForPrize | Bool | 혜택 수령 시 번호 수집 여부 |
| prizeType | String | 혜택 종류   * 0 : TV\_COIN * 1 : COUPON * 2 : MANUAL |
| prizeDetail | String | 혜택 상세 내용 |
| winConfirmStatus | String | 당첨 확인 여부.   * 0 : 미확인 * 1 : 수령확인 |
| prizeReceiptStatus | String | 경품 수령확인 여부.   * 0 : 미확인 * 1 : 수령확인 * 2 : 수령취소 |
| phone | String | 수집된 전화번호. 010-\*\*\*\*-1234 형식으로 가운데 4자리는 별표. |
| couponInfo | String | "쿠폰명|이용타입|금액|할인율|쿠폰만료일시" |

## SubscriberEnrollDetail

가입자의 참여 상세 내역.

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| targetType | String | 이벤트 실행 대상 종류(ASSET, PRODUCT) |
| assetId | String | Asset 아이디(providerId|assetId) |
| productId | String | 상품 Product ID |
| goodId | String | 상품 Good ID. 값이 없을 경우 빈 값(“”)으로 처리.   * RVOD, SVOD, Package 상품 타입인 경우 “productId|goodId” 쌍으로 상품 ID를 표현하므로 goodId 값이 반드시 있음. * SVODPackage, Bundle 상품 타입인 경우 goodId 없음. goodId 값 없는 경우 빈 값(“”)으로 처리. |
| targetName | String | 대상 이름   * Asset의 타이틀 * Product의 상품명 |
| enrollTime | String | 참여 일시(yyyy-MM-dd HH:mm:ss) |
| price | Int | 구매 가격 |

## EventViewTarget

EventViewTarget은 이벤트 표현 대상 정보이며, 이벤트 매니저에서 설정한 이벤트 대상 에셋/상품/카테고리 정보이다. 이벤트에 카테고리를 설정한 경우 해당 카테고리 자체가 EventViewTarget이 된다.

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| targetType | String | 요청 대상의 타입   * ASSET : 에셋 * PRODUCT : 상품 * CATEGORY : 카테고리 |
| targetId | String | 타입에 따른 target ID   * targetType이 ASSET인 경우 “providerId|assetId” 쌍으로 이루어진 에셋 아이디 * targetType이 CATEGORY인 경우 카테고리 아이디 * targetType이 PRODUCT인 경우   + FOD, RVOD, SVOD 상품은 “productId|goodId” 형식   + SVODPackage, Bundle 상품은 productId 형식.   + ~~SVODPackage, Bundle 상품은 product\_id 형식. (goodId 없으면 빈 값(“”)으로 처리)~~ |
| eventList | *Event[]* | 이벤트 표현 대상에 연관된 이벤트 목록.   * 여기에 포함되는 이벤트 정보는 프로파일 0~~, 1~~만 지원함. |

## EventActionTarget

EventActionTarget은 이벤트 실행 대상 정보이며 이벤트에 구매/시청 행위를 통해 이벤트에 직접 참여할 수 있는 실제 대상을 말한다. 이벤트에 대상에 카테고리가 설정 된 경우 EventActionTarget은 선택한 카테고리 내에 있는 각 FOD/RVOD 들이 된다.

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| targetType | String | 요청 대상의 타입   * ASSET : 에셋 * PRODUCT : 상품 * CATEGORY : 카테고리 |
| targetId | String | 타입에 따른 target ID   * targetType이 ASSET인 경우 “providerId|assetId” 쌍으로 이루어진 에셋 아이디 * targetType이 CATEGORY인 경우 카테고리 아이디 * targetType이 PRODUCT인 경우   + FOD, RVOD, SVOD 상품은 “productId|goodId” 형식 * SVODPackage, Bundle 상품은 “productId” 형식. * ~~SVODPackage, Bundle 상품은 “productId|” 형식.(goodId 없으면 빈 값(“”)으로 처리)~~ |
| title | String | 에셋 타이틀. 상품명 |
| price | Int | 가격 |
| redirectCategoryId | String | 대상이 상품인 경우 연결된 카테고리 아이디 |

## RelatedEvent

ActionTarget 과 연관된 이벤트 정보.

|  |  |  |  |
| --- | --- | --- | --- |
| 이름 | 타입 | 설명 | ~~프로파일~~ |
| eventId | Int | 이벤트 아이디 | ~~0~~ |
| title | String | 이벤트 타이틀 | ~~0~~ |
| description | String | 설명 | ~~0~~ |
| eventType | String | UI 이벤트 타입   * 0 : 경품응모 * 1 : 스탬프 * 2 : N+1 * 3 : 합산페이백 | ~~0~~ |
| isAutoEnroll | Bool | 자동참여 여부 | ~~0~~ |
| displayStartTime | String | 노출 시작 일시. yyyy-MM-dd HH:mm | ~~0~~ |
| displayEndTime | String | 노출 종료 일시. yyyy-MM-dd HH:mm | ~~0~~ |
| actionStartTime | String | 진행 시작 일시. yyyy-MM-dd HH:mm | ~~0~~ |
| actionEndTime | String | 진행 종료 일시. yyyy-MM-dd HH:mm | ~~0~~ |
| status | Int | 이벤트 상태   * 10 : 이벤트 전 * 20 : 이벤트 중 * 30 : 마감임박 D-3 * 40 : 추첨 대기(진행기간 종료 후 추첨 전까지) * 50 : 추첨 완료(ALL 이외 나머지 상품의 당첨자가 모두 결정 된 경우) * 51 : 상품소진완료(“참여 혜택 당첨 완료 시, 이벤트 종료“ 설정된 이벤트의 상품이 소진 된 경우) * ~~60 : 발표 대기(추첨 완료 후 ALL 이외 나머지 모든 당첨자 개인정보가 입력된 상태)~~ * 70 : 발표 완료(발표 버튼을 누른 경우) * ~~-10~~ 🡪 80 : 진행 중지 | ~~0~~ |
| enrollStatus | String | 참여 상태   * 0 : 미참여 * 1 : 참여 의사 표시 * 2 : 참여 중 * 9 : 참여 완료 | ~~1~~ |

## Winner

이벤트 정보에 포함되는 당첨자 정보. 혜택 우선순위 오름차순, 이름 오름차순 정렬

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| name | String | 당첨자 이름.   * 김\*동 형식으로 가운데 글자 \* 처리, 이름이 외자이면 박\* 형식으로 이름만 \* 처리 |
| phone | String | 당첨자 연락처.   * 010-\*\*\*\*-1234 형식으로 국번과 마지막 번호4자리 제외한 나머지 번호 \* 처리 |
| prize | String | 혜택 정보 |

## PortalMenu

이벤트포털 메뉴 정보

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| menuId | Int | 이벤트포털 메뉴 아이디 |
| name | String | 메뉴 명 |
| type | String | 메뉴 타입   * 0 : Main * 1 : Event * 2 : Notice * 9 : MyEvent |
| listSize | Int | 메뉴에 속한 리스트의 개수 |

## Notice

공지사항 정보

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| noticeId | Int | 공지사항 아이디 |
| Title | String | 공지사항 제목 |
| description | String | 공지사항 내용 |
| creationTime | String | 최초 등록일. yyyy-MM-dd HH:mm:ss 형식 |

## EnrollReservationInfo

공지사항 정보

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| serviceId | String | 시청 아이디 |
| serviceDomain | String | 시청 도메인. 시청한 콘텐츠 정보를 관리하는 도메인 정보. C3, C7 등등 |
| asseId | String | 에셋 아이디.  providerId|assetId 형식 |
| purchaseId | String | 구매 아이디 |
| purchaseDomain | String | 구매 도메인. 구매한 상품 정보를 관리하는 도메인 정보. C3, C7 등등 |
| productId | String | 상품 productId |
| goodId | String | 상품 goodId. goodId 값이 없는 경우 빈 값(“”)으로 처리. |
| productType | String | 상품 타입.   * 0 : FOD * 1 : RVOD * 2 : PACKAGE * 3 : BUNDLE * 4 : SVOD * 5 : SVOD\_PACKAGE   참여 로그를 위해 수집하는 정보. 참여처리 내용에는 무관함. |
| isUsedFreeCoupon | Bool | 무료 쿠폰 사용 여부 true/false |
| price | Int | 가격 |

# 기능 및 인터페이스 목록

## 이벤트 아이콘, 상세팝업 관련 API

### getEventListForViewTarget

#### 상세 설명

1. 설명
   1. 이벤트 표현 대상(에셋/상품/카테고리) 포스터의 이벤트 아이콘 정보 조회
   2. **지역, STB 기준 노출대상 이벤트 정보를 조회한다. wHAS에서 이벤트 아이콘 정보 조회 시 캐슁 처리를 위해서 가입자 관련 정보를 제외한 이벤트 정보만 조회하는 API로 만듬.**
2. 요청 상황
   1. 에셋, 카테고리, 상품 목록 조회 시 각 대상에 이벤트 아이콘 표시 정보 조회 시
3. 처리내용
   1. 기본 처리 조건
      1. 경품응모 이벤트(목표달성형 이벤트 제외)
      2. 진행중, 마감임박 상태인 이벤트
      3. SO, STB 종류가 노출 대상에 포함된 이벤트
      4. 조회 대상이 이벤트 표현 대상에 포함된 이벤트.
         1. 카테고리로 조회 시 해당 카테고리에 대해서만 이벤트 정보 조회하며, 카테고리 내 에셋에 대해서는 조회하지 않음.
      5. 진행중지, 상품소진완료 상태가 아닌 이벤트
         1. normalEvent 테이블의 stopStatus = 0
      6. 정렬
   2. 정렬 기준
      1. 이벤트 유형 : 수동 경품 > 자동 경품(isAutoEnroll desc)
      2. 동일 유형일 경우 : 최초 등록순(creationTime ASC)

#### 요청 주소

http://serverIP:port/EventPortalServer/getEventListForViewTarget.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| ~~targetList~~ | ~~Target[]~~ | ~~대상 목록~~ | ~~●~~ |  |
| targetType | String | 대상 타입(ASSET, CATEGORY, PRODUCT).   * targetId와 쌍을 이룬다. | ● |  |
| targetId | String | 대상 아이디.   * targetType = PRODUCT인 경우   + FOD, RVOD, SVOD 상품은 “productId|goodId” 형식   + SVODPackage, Bundle 상품은 “productId” 형식.   + ~~SVODPackage, Bundle 상품은 “product\_id|” 형식. goodId 값 없는 경우 빈 값(“”)으로 처리.~~   + ~~따라서 goodId없는 targetId의 경우 productType이 SVODPackage, Bundle 상품 중에서만 조회해야 함.~~ * 조회하고자 하는 target이 여러 개이면 targetType, targetId를 target 수 만큼 전송하면 되고, targetType, targetId 쌍의 순서는 동일해야 한다. * n번째 targetType과 n번째 targetId를 묶어서 하나의 target으로 보고 처리한다. | ● |
| targetCount | Int | 요청변수로 넘어오는 target의 총 개수.   * targetCount 값과 targetType 수, targetId 수가 다르면 예외처리. | ● |  |
| soId | String | 가입자의 지역 정보.   * VOD시스템의 externalSoId 값 | ● |  |
| stbModelName | String | 셋탑 모델 명 | ● |  |
| profile | Int | 조회하고자 하는 이벤트 정보의 수준.   * **(지원 프로파일 : 0)** |  | 0 |
| ~~imagePosition~~ | ~~String~~ | ~~필요한 이미지의 위치 정보. 값이 없을 경우 이미지 정보 제외.~~   * ~~0 : 이벤트포털 상세화면 이미지~~ * ~~1 : 이벤트포털 기본 리스트 배너 이미지~~ * ~~2 : 이벤트포털 특정 카테고리 리스트 배너 이미지~~ * ~~3 : 상세팝업 이미지~~ |  |  |
| sortType | String | 각 타겟의 결과에 포함된 이벤트 목록의 정렬 순서.   * manualEnrollFirst : 수동참여형 우선 (수동 경품 > 자동 경품. 동일 유형일 경우 최초 생성순) * creationTimeDesc : 최근 생성순 |  | manualEnrollFirst |
| eventListSize | Int | 각 target 별 이벤트 목록 크기.   * 0 : 전체목록. * N : 각 target 별로 N개의 이벤트 목록만 조회 |  | 0 |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| eventViewTargetList | *EventView Target[]* | 이벤트 표현 대상 목록.   * 각 대상 별로 연관 이벤트 목록 포함. |

#### 요청과 응답 예

|  |
| --- |
|  |
|  |

예. getEventListForViewTarget – JSON 응답

### getEventForViewTargetPopup

#### 상세 설명

1. 설명
   1. 이벤트 표현 대상(에셋/상품/카테고리) 상세 팝업 정보 조회
2. 요청 상황
   1. 특정 에셋, 상품, 카테고리를 선택 할 때 관련된 이벤트 상세팝업 정보 조회 시
3. 처리내용
   1. 기본 처리 조건
      1. 경품응모 이벤트(목표달성형 이벤트 제외)
      2. 진행중, 마감임박 상태인 이벤트
      3. 가입자가 노출 대상에 포함된 이벤트
         1. SO, STB 종류가 노출대상에 포함 된 이벤트
      4. 조회 대상이 이벤트 표현 대상에 포함된 이벤트.
         1. 카테고리로 조회 시 해당 카테고리에 대해서만 이벤트 정보 조회하며, 카테고리 내 에셋에 대해서는 조회하지 않음.
      5. 진행중지, 상품소진완료 상태가 아닌 이벤트
         1. normalEvent 테이블의 stopStatus = 0
      6. 참여하지 않은 이벤트(참여의사표시, 참여중, 참여완료 제외)
      7. 에셋, 상품 정보가 함께 들어오는 경우 에셋 관련 이벤트 + 상품 관련 이벤트 합집합 목록에서 중복 제거한 결과 반환
      8. 정렬 기준
         1. 이벤트 유형 : 수동 경품 > 자동 경품(isAutoEnroll desc)
         2. 동일 유형일 경우 : 최초 등록순(creationTime ASC)
      9. **정렬 한 결과 중 첫 번째 이벤트 정보만 반환**

#### 요청 주소

http://serverIP:port/EventPortalServer/getEventForViewTargetPopup.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| ~~assetId~~ | ~~String~~ | ~~에셋 아이디~~ | ~~●~~  ~~assetId, productId 둘 중 하나는 필수~~ |  |
| ~~productId~~ | ~~String~~ | ~~상품의 productId~~ |  |
| ~~goodId~~ | ~~String~~ | ~~상품의 goodId. 값이 없는 경우 빈 값(“”) 으로 처.~~ | ~~●~~  ~~productId가 있을 경우 필수~~ |  |
| ~~productType~~ | ~~String~~ | ~~상품 타입.~~   * ~~SVOD~~ * ~~SVOD\_PACKAGE~~ * ~~BUNDLE~~ * ~~PACKAGE~~ * ~~FOD~~ * ~~RVOD~~ | ~~●~~  ~~productId가 있을 경우 필수~~ |  |
| targetType | String | 대상 타입(ASSET, CATEGORY, PRODUCT).   * targetId와 쌍을 이룬다. | ● |  |
| targetId | String | 대상 아이디.   * 에셋의 경우 “providerId|assetId” 형식 * targetType = PRODUCT인 경우   + FOD, RVOD, SVOD 상품은 “productId|goodId” 형식   + SVODPackage, Bundle 상품은 “productId” 형식.   + ~~SVODPackage, Bundle 상품은 “product\_id|” 형식. goodId 값 없는 경우 빈 값(“”)으로 처리.~~   + ~~따라서 goodId없는 targetId의 경우 productType이 SVODPackage, Bundle 상품 중에서만 조회해야 함.~~ | ● |  |
| subscriberId | String | 가입자 아이디 | ● |  |
| soId | String | 가입자의 지역 정보.   * VOD시스템의 externalSoId 값 | ● |  |
| stbModelName | String | 셋탑 모델 명 | ● |  |
| profile | Int | 조회하고자 하는 이벤트 정보의 수준.   * **(지원 프로파일 : 0, 1)** |  | 1 |
| imagePosition | String | 필요한 이미지의 위치 정보. 값이 없을 경우 이미지 정보 제외.   * ~~0 : 이벤트포털 상세화면 이미지~~ * ~~1 : 이벤트포털 기본 리스트 배너 이미지~~ * ~~2 : 이벤트포털 특정 카테고리 리스트 배너 이미지~~ * ~~3 : 상세팝업 이미지~~ * 1 : 이벤트포탈 기본 리스트 이미지 * 2 : 상세팝업 이미지 * 3 : 홈 메뉴 배너 이미지 * 4 : VOD종료 이벤트 배너 이미지 |  | 2 |
| sortType | String | 각 타겟의 결과에 포함된 이벤트 목록의 정렬 순서.   * manualEnrollFirst : 수동참여형 우선 (수동 경품 > 자동 경품. 동일 유형일 경우 최초 생성순) * creationTimeDesc : 최근 생성순 |  | manualEnrollFirst |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| event | *Event* | 이벤트 정보 |

#### 요청과 응답 예

|  |
| --- |
|  |
|  |

예. getEventListForViewTargetPopup – JSON 응답

## Toast팝업, 시청종료화면 관련 API

### getEventListForActionTarget

#### 상세 설명

1. 설명
   1. VOD 시청 중 Toast 팝업, 시청종료화면 이벤트 배너 용 이벤트 목록 조회
   2. 이벤트 실행 대상(상품, 에셋)을 통해 참여 가능 한 이벤트 목록 조회
2. 요청 상황
   1. VOD 시청 화면의 Toast 팝업에서 보여줄 현재 시청중인 컨텐츠의 에셋/상품 정보로 참여 가능한 이벤트 정보 정보 조회 시. 프로파일 “0”으로 조회
   2. VOD 시청 종료 화면의 이벤트 배너에 연결 할 참여 가능 한 이벤트 정보 조회 시. 프로파일 “1”로 조회(이미지는 VOD종료 이벤트 배너 이미지로 요청)
3. 처리내용
   1. 기본 처리 조건
      1. 가입자가 노출 대상인 이벤트
      2. 대상이 이벤트 대상인 이벤트
      3. 진행 중 상태인 이벤트
      4. 진행중지 상태가 아닌 이벤트
      5. 미참여 이벤트
      6. 진행중지, 상품소진완료 상태가 아닌 이벤트
         * 1. normalEvent 테이블의 stopStatus = 0
      7. 에셋, 상품 정보가 함께 들어오는 경우 에셋 관련 이벤트 + 상품 관련 이벤트 합집합 목록에서 중복 제거한 결과 반환
   2. 정렬 처리~~기준~~
      1. ~~목표달성형 > 수동 경품 > 자동 경품~~
      2. ~~동일 유형일 경우 : 최초 등록순(creationTime ASC)~~

#### 요청 주소

http://serverIP:port/EventPortalServer/getEventListForActionTarget.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| assetId | String | 애셋 ID | ●  assetId, productId 둘 중 하나는 필수. 둘 다 들어오는 경우 에셋, 상품에 대하여 한꺼번에 처리 |  |
| productId | String | 상품 Product ID |  |
| goodId | String | 상품 Good ID   * RVOD, SVOD, Package 상품 타입인 경우 <productId, goodId> 쌍으로 상품 ID를 표현하므로 goodId 값이 반드시 있어야 한다. * SVODPackage나 Bundle 상품 타입인 경우 goodId 는 빈값(””)으로 입력. * ~~따라서 goodId없는 targetId의 경우 productType이 SVODPackage, Bundle 상품 중에서만 조회해야 함.~~ | ●  productId가 있을때는 필수 |  |
| ~~productType~~ | ~~String~~ | ~~상품 타입.~~   * ~~SVOD~~ * ~~SVOD\_PACKAGE~~ * ~~BUNDLE~~ * ~~PACKAGE~~ * ~~FOD~~ * ~~RVOD~~ | ~~●~~  ~~productId가 있을 때는 필수~~ |  |
| subscriberId | String | 가입자 아이디 | ● |  |
| soId | String | 가입자의 지역 정보. externalSoId 값 | ● |  |
| stbModelName | String | 셋탑 모델 명 | ● |  |
| profile | Int | 결과 정보의 수준.   * **(지원 프로파일 : 0, 1)** * **Toast 팝업 조회 시 : 0** * **VOD 종료 이벤트 배너 조회 시 : 1** |  | 0 |
| imagePosition | String | 필요한 이미지의 위치 정보. 값이 없을 경우 이미지 정보 제외.   * 1 : 이벤트포탈 기본 리스트 이미지 * 2 : 상세팝업 이미지 * 3 : 홈 메뉴 배너 이미지 * 4 : VOD종료 이벤트 배너 이미지 |  | 4 |
| sortType | String | * default   + 목표달성형 > 수동 경품 > 자동 경품   + 동일 유형일 경우 : 최초 등록순(creationTime ASC) |  | default |
| eventListSize | Int | 결과로 받을 이벤트 목록 개수   * 0 : 전체목록. * N : N개의 이벤트 목록만 조회. |  | 0 |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| totalCount | Int | 총 이벤트 수 |
| eventList | *event[]* | 이벤트 정보 목록 |

#### 요청과 응답 예

|  |
| --- |
|  |
|  |

예. getEventListForAction – JSON 응답

### ~~getEnrolledEventListForActionTarget~~ 🡪 getEnrolledEventListForAction

#### 상세 설명

1. 설명
   1. 구매 아이디, 시청 아이디로 참여하여 참여 완료 된 이벤트 조회
   2. Toast 팝업 용
2. 요청 상황
   1. 재생 중 toast 팝업에서 보여줄 이벤트 중 바로 전에 구매 혹은 시청 요청으로 참여한 ~~완료한~~ 이벤트 조회
3. 처리내용
   1. 기본 처리 조건
      1. 요청 받은 구매 아이디, 시청 아이디로 가입자가 참여한 이벤트
      2. 정렬순서
         1. 목표달성형 > 수동 경품 > 자동 경품
         2. 동일 유형일 경우 : 최초 등록순(creationTime ASC)
   2. 제외 조건
      1. 타임세일 이벤트
         1. event.type=1

#### 요청 주소

http://serverIP:port/EventPortalServer/getEnrolledEventListForAction.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| purchaseDomain | String | 구매 도메인 | ●  구매정보, 시청 정보 중 하나는 필수 |  |
| purchaseId | String | 구매 아이디 |  |
| ServiceDomain | String | 시청 도메인 |  |
| ServiceId | String | 시청 아이디 |  |
| subscriberId | String | 가입자 아이디 | ● |  |
| profile | Int | 결과 정보의 수준.   * **(지원 프로파일 : 0)** |  | 0 |
| sortType | String | default : 목표달성형 > 수동 경품 > 자동 경품 |  | default |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| totalCount | Int | 총 이벤트 수 |
| EventList | *event[]* | 이벤트 정보 목록 |

#### 요청과 응답 예

## 이벤트 참여 관련 API

### ~~doEnrollStart~~

참여 의사 표시 API 삭제

### ~~doEnrollProcess~~

참여 처리 API 삭제

### doManualEnroll ~~<- doManualEnrollStart~~

#### 상세 설명

1. 설명
   1. 수동이벤트 참여 시작 및 참여 처리
2. 주의사항
   1. ~~STB UI에서 이 API를 요청한 후 결과에서 당첨이 된 경우에는 당첨확인 API(doWinConfirm) 를 이용하여 당첨확인 요청을 다시 해야 함.~~
3. 요청 상황
   1. 시청종료 후 수동참여 이벤트의 이벤트 포털 상세 화면에서 참여 시
4. 처리 내용
   1. eventId에 해당하는 이벤트에 참여의사표시 처리 및 참여 처리
   2. 수동참여 이벤트 인지 확인.
   3. 미참여 이벤트인지 확인( enroll.isDelete = false 인 참여 정보가 없는 이벤트)
   4. 진행 중인 이벤트인지 확인
   5. 중지상태인 이벤트 제외
   6. 가입자가 노출 대상인지 확인
   7. 참여 대상 정보 및 참여 조건 확인
      1. ~~시청 관련 정보(에셋 ID + 시청 ID) 혹은 구매 관련 정보(상품 ID + 구매 ID) 중 한 가지만 있는지 확인. 둘 다 있으면 실패~~
      2. 시청 관련 정보가 오는 경우
         1. assetID, serviceId, serviceDomain 필수.
         2. 시청 이벤트 아니면 실패
      3. 구매 관련 정보가 오는 경우
         1. productId, goodId, purchaseId, purchaseDomain 필수.
         2. 구매 이벤트 아니면 실패
      4. 이벤트 enrollActionType이 구매 타입이면 구매 관련 정보 필수, 시청 타입이면 시청 관련 정보 필수
      5. 예약된 이벤트인지 확인 (Enroll\_Reservation)
   8. 중복요청 시 처리
      1. enrollActionType에 따라 요청 받은 “productId” 혹은 “assetId” 로 참여가 이미 된 상태이면 “이미 처리되었음”.
      2. ~~이미 참여 했으나 요청 받은 에셋/상품으로 참여된 상태가 아니면 참여 처리.~~
   9. 참여 시 개인정보 수집(isCollectPhoneNumberForEnroll = true) 하는 이벤트인 경우 전화번호 입력 안되면 실패.
   10. 무료쿠폰 허용하지 않은 이벤트에 무료쿠폰 이용하여 참여하려는 경우 실패.
   11. EnrollServer로 예약 confirm API호출
       1. 응답 실패에 대해서는 설정 값으로 재시도
       2. 응답 값이 실패인 경우에는 재시도 하지않음

#### 요청 주소

http://serverIP:port/EventPortalServer/doManualEnrollStart.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| eventId | Int | 이벤트 아이디 | ● |  |
| subscriberId | String | 가입자 아이디  (가입자 정보조회, 참여/당첨, 쿠폰/xPush 연동시 사용할 이벤트포털 관련하여 공통으로 사용가능한 단일 가입자 아이디)  VOD시스템에서는 customer\_id  EPG, 쿠폰, xPUSH 시스템에서는 CAS\_ID | ● |  |
| soId | String | 가입자의 지역 정보. externalSoId 값 | ● |  |
| stbModelName | String | 셋탑 모델 명 | ● |  |
| phone | String | 연락처.(참여시 개인정보 수집하는 경우) |  |  |
| serviceId | String | 시청 아이디 | ●  시청 시 참여 이벤트인 경우 필수. |  |
| serviceDomain | String | 시청 도메인. 시청한 콘텐츠 정보를 관리하는 도메인 정보. C3, C7 등등 |  |
| asseId | String | 에셋 아이디.  providerId|assetId 형식 |  |
| purchaseId | String | 구매 아이디 | ●  구매 시 참여 이벤트인 경우 필수 |  |
| purchaseDomain | String | 구매 도메인. 구매한 상품 정보를 관리하는 도메인 정보. C3, C7 등등 |  |
| productId | String | 상품 productId |  |
| goodId | String | 상품 goodId. goodId 값이 없는 경우 빈 값(“”)으로 처리. |  |
| productType | String | 상품 타입.   * 0 : FOD * 1 : RVOD * 2 : PACKAGE * 3 : BUNDLE * 4 : SVOD * 5 : SVOD\_PACKAGE   참여 로그를 위해 수집하는 정보. 참여처리 내용에는 무관함. |  |  |
| isUsedFreeCoupon | Bool | 무료 쿠폰 사용 여부 true/false |  | false |
| price | Int | 가격 | ● |  |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| ~~subscriberWinInfo~~ | ~~SubscriberWinInfo~~ | ~~참여 후 실시간 추첨에 당첨된 경우 당첨 결과 정보.~~ |

#### 요청과 응답 예

### doEnrollProcessByPurchase

#### 상세 설명

1. 설명
   1. 구매 시 구매한 상품과 연관된 구매 이벤트 참여 처리.
2. 요청 상황
   1. 상품 구매 시
3. 처리 내용
   1. 참여처리 할 상품 연관 이벤트 목록
      1. 가입자가 노출 대상에 포함된 이벤트
      2. 참여 대상 컨텐츠가 이벤트 대상에 포함된 이벤트
      3. 진행 중 상태인 이벤트
      4. 중지 상태가 아닌 이벤트(진행중지, 상품소진완료)
      5. 무료쿠폰을 이용하여 구매 한 경우 무료쿠폰을 이용한 참여가 허용 된 이벤트
      6. 자동 참여 이벤트 (isAutoEnroll = false)
         1. 참여 로그 판단
            1. EnrollLog.subscriberId = 입력값
            2. EnrollLog.productId= 입력값
            3. EnrollLog.goodId = 입력값
         2. 참여 로그가 존재 할 경우 참여가 삭제 되었을 경우 참여
            1. Enroll.subscriberId = 입력값
            2. Enroll.isDeleted = true
         3. 참여 로그가 존재하지 않을 경우
            1. 해당 이벤트에 대한 모든 참여의 상태가 참여 의사표시(enrollStatus = 1), 참여 중(enrollStatus = 2) 이면 참여

Enroll.subscriberId = 입력값

Enroll.isDeleted = false OR Enroll.isDeleted is null

Enroll.enrollStatus = 1 OR Enroll.enrollStatus = 2

* + - * 1. 해당 이벤트에 대한 다른 참여의 상태가 삭제 되었을 경우 참여

Enroll.subscriberId = 입력값

Enroll.isDeleted = true

* + - * 1. 해당 이벤트에 대한 다른 참여가 없을 때 참여
    1. 수동 참여 이벤트 (isAutoEnroll = false)
       1. 참여 로그 판단
          1. EnrollLog.subscriberId = 입력값
          2. EnrollLog.productId= 입력값
          3. EnrollLog.goodId = 입력값
       2. 참여 로그가 존재하지 않을 경우
          1. 해당 이벤트에 대한 다른 참여의 상태가 참여 의사표시(enrollStatus = 1), 참여 중(enrollStatus = 2) 이면 참여

Enroll.subscriberId = 입력값

Enroll.isDeleted = false OR Enroll.isDeleted is null

Enroll.enrollStatus = 1 OR Enroll.enrollStatus = 2

* + 1. **위에서 조회한 각 이벤트에 대하여 EventEnrollServer에 참여 API로 요청.**
       1. 응답 실패에 대해서는 설정 값으로 재시도
       2. 응답 값이 실패인 경우에는 재시도 하지않음
  1. 예약처리 할 상품 연관 이벤트 목록 조회
     1. 가입자가 노출 대상에 포함된 이벤트
     2. 참여 대상 컨텐츠가 이벤트 대상에 포함된 이벤트
     3. 진행 중 상태인 이벤트
     4. 중지 상태가 아닌 이벤트(진행중지, 상품소진완료)
     5. 무료쿠폰 이용여부는 체크하지 않음 -> 무료쿠폰 사용여부 정보를 받아서 그대로 예약정보에 저장함.
     6. 수동 참여 이벤트 (isAutoEnroll = false)
        1. 참여 로그 판단
           1. EnrollLog.subscriberId = 입력값
           2. EnrollLog.productId= 입력값
           3. EnrollLog.goodId = 입력값
        2. 참여 로그가 존재 할 경우 해당 참여가 삭제 되었을 경우 예약
           1. Enroll.subscriberId = 입력값
           2. Enroll.isDeleted = true
        3. 참여 로그가 존재하지 않을 경우
           1. 다른 참여의 상태가 삭제 되었을 경우 예약

Enroll.subscriberId = 입력값

Enroll.isDeleted = true

* + - * 1. 해당 이벤트에 대한 다른 참여가 없을 때 예약
    1. **위에서 조회한 각 이벤트에 대하여 EventEnrollServer에 예약 API로 요청.**
       1. 응답 실패에 대해서는 설정 값으로 재시도
       2. 응답 값이 실패인 경우에는 재시도 하지않음

1. 주의사항
   1. [이벤트, 가입자, 참여의사표시, 참여처리] 단위로 동시 처리 안되도록 singleton 처리 필요함.

#### 요청 주소

http://serverIP:port/EventPortalServer/doEnrollProcessByPurchase.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| subscriberId | String | 가입자 아이디  (가입자 정보조회, 참여/당첨, 쿠폰/xPush 연동시 사용할 이벤트포털 관련하여 공통으로 사용가능한 단일 가입자 아이디)  VOD시스템에서는 customer\_id  EPG, 쿠폰, xPUSH 시스템에서는 CAS\_ID | ● |  |
| soId | String | 가입자의 지역 정보. externalSoId 값 | ● |  |
| stbModelName | String | 셋탑 모델 명 | ● |  |
| productId | String | 상품 productId | ● |  |
| goodId | String | 상품 goodId. goodId 값이 없는 경우 빈 값(“”)으로 처리. | ● |  |
| productType | String | 상품 타입.   * 0 : FOD * 1 : RVOD * 2 : PACKAGE * 3 : BUNDLE * 4 : SVOD * 5 : SVOD\_PACKAGE |  |  |
| purchaseDomain | String | 구매 도메인. 구매한 상품 정보를 관리하는 도메인 정보. CASTIS\_VOD 등등 | ● |  |
| purchaseId | String | 구매 아이디 | ● |  |
| isUsedFreeCoupon | Bool | 무료 쿠폰 사용 여부 true/false | ● | false |
| price | Int | 가격 | ● |  |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| ~~resultCount~~ | ~~Int~~ | ~~총 참여된 이벤트 수~~ |

#### 요청과 응답 예

### doEnrollProcessByService

#### 상세 설명

1. 설명
   1. 시청 시 에셋과 연관된 시청 이벤트 참여 처리
2. 요청 상황
   1. 콘텐츠 시청 시
3. 처리 내용
   1. 참여처리 할 에셋 연관 이벤트 목록 조회
      1. 가입자가 노출 대상에 포함된 이벤트
      2. 참여 대상 컨텐츠가 이벤트 대상에 포함된 이벤트
      3. 진행 중 상태인 이벤트
      4. 중지 상태가 아닌 이벤트(진행중지, 상품소진완료)
      5. 자동 참여 이벤트 (isAutoEnroll = false)
         1. 참여 로그 판단
            1. EnrollLog.subscriberId = 입력값
            2. EnrollLog.assetId= 입력값
         2. 참여 로그가 존재 할 경우 참여가 삭제 되었을 경우 참여
            1. Enroll.subscriberId = 입력값
            2. Enroll.isDeleted = true
         3. 참여 로그가 존재하지 않을 경우
            1. 해당 이벤트에 대한 모든 참여의 상태가 참여 의사표시(enrollStatus = 1), 참여 중(enrollStatus = 2) 이면 참여

Enroll.subscriberId = 입력값

Enroll.isDeleted = false OR Enroll.isDeleted is null

Enroll.enrollStatus = 1 OR Enroll.enrollStatus = 2

* + - * 1. 해당 이벤트에 대한 다른 참여의 상태가 삭제 되었을 경우 참여

Enroll.subscriberId = 입력값

Enroll.isDeleted = true

* + - * 1. 해당 이벤트에 대한 다른 참여가 없을 때 참여
    1. 수동 참여 이벤트 (isAutoEnroll = false)
       1. 참여 로그 판단
          1. EnrollLog.subscriberId = 입력값
          2. EnrollLog.assetId= 입력값
       2. 참여 로그가 존재하지 않을 경우
          1. 해당 이벤트에 대한 다른 참여의 상태가 참여 의사표시(enrollStatus = 1), 참여 중(enrollStatus = 2) 이면 참여

Enroll.subscriberId = 입력값

Enroll.isDeleted = false OR Enroll.isDeleted is null

Enroll.enrollStatus = 1 OR Enroll.enrollStatus = 2

* + 1. **위에서 조회한 각 이벤트에 대하여 EventEnrollServer에 참여 API로 요청.**
       1. 응답 실패에 대해서는 설정 값으로 재시도
       2. 응답 값이 실패인 경우에는 재시도 하지않음
  1. 예약처리 할 상품 연관 이벤트 목록 조회
     1. 가입자가 노출 대상에 포함된 이벤트
     2. 참여 대상 컨텐츠가 이벤트 대상에 포함된 이벤트
     3. 진행 중 상태인 이벤트
     4. 중지 상태가 아닌 이벤트(진행중지, 상품소진완료)
     5. 무료쿠폰 이용여부는 체크하지 않음 -> 무료쿠폰 사용여부 정보를 받아서 그대로 예약정보에 저장함.
     6. 수동 참여 이벤트 (isAutoEnroll = false)
        1. 참여 로그 판단
           1. EnrollLog.subscriberId = 입력값
           2. EnrollLog.assetId= 입력값
        2. 참여 로그가 존재 할 경우 해당 참여가 삭제 되었을 경우 예약
           1. Enroll.subscriberId = 입력값
           2. Enroll.isDeleted = true
        3. 참여 로그가 존재하지 않을 경우
           1. 다른 참여의 상태가 삭제 되었을 경우 예약

Enroll.subscriberId = 입력값

Enroll.isDeleted = true

* + - * 1. 해당 이벤트에 대한 다른 참여가 없을 때 예약
    1. **위에서 조회한 각 이벤트에 대하여 EventEnrollServer에 예약 API로 요청.**
       1. 응답 실패에 대해서는 설정 값으로 재시도
       2. 응답 값이 실패인 경우에는 재시도 하지않음

1. 주의사항
   1. [이벤트, 가입자, 참여의사표시, 참여처리] 단위로 동시 처리 안되도록 singleton 처리 필요함.

#### 요청 주소

http://serverIP:port/EventPortalServer/doEnrollProcessByService.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| subscriberId | String | 가입자 아이디  (가입자 정보조회, 참여/당첨, 쿠폰/xPush 연동시 사용할 이벤트포털 관련하여 공통으로 사용가능한 단일 가입자 아이디)  VOD시스템에서는 customer\_id  EPG, 쿠폰, xPUSH 시스템에서는 CAS\_ID | ● |  |
| soId | String | 가입자의 지역 정보. externalSoId 값 | ● |  |
| stbModelName | String | 셋탑 모델 명 | ● |  |
| asseId | String | 에셋 아이디. “providerId|assetId” 형식 | ● |  |
| serviceDomain | String | 시청 도메인. 시청한 콘텐츠 정보를 관리하는 도메인 정보. CASTIS\_VOD 등등 | ● |  |
| serviceId | String | 시청 아이디 | ● |  |
| price | Int | 가격 |  |  |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | int | 결과 코드 |
| errorString | String | 에러 내용 |
| ~~totalCount~~ | ~~Int~~ | ~~총 참여된 이벤트 수~~ |

#### 요청과 응답 예

## 이벤트 포털 관련 API

### getPortalMenuList

#### 상세 설명

1. 설명
   1. 이벤트포털로 진입 시 포털 메뉴 목록 조회
2. 요청 상황
   1. 이벤트포털의 진입 시
3. 처리 내용
   1. 이벤트포털 메뉴 목록 조회
   2. 정렬 기준
      1. 이벤트포털 메뉴의 sortSequence 오름 차순

#### 요청 주소

http://serverIP:port/EventPortalServer/getPortalMenuList.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
|  |  |  |  |  |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| totalCount | Int | 총 이벤트 수 |
| portalMenuList | *PortalMenu[]* | 이벤트포털 메뉴 목록 |
|  |  |  |
|  |  |  |

#### 요청과 응답 예

### getPortalEventList

#### 상세 설명

1. 설명
   1. 이벤트포털로 진입 시 특정 포털 메뉴에 속한 이벤트 목록 조회
2. 요청 상황
   1. 이벤트포털의 특정 메뉴 진입 시
3. 처리 내용
   1. 특정 이벤트포털 메뉴에 등록된 이벤트 목록
   2. 가입자가 노출 대상인 이벤트
   3. 현재 노출 기간인 모든 이벤트
   4. 이벤트포털 리스트용 이벤트 이미지 정보 포함
      1. 이벤트포털 메뉴에 등록된 이벤트의 리스트용 이미지 우선
      2. 없는 경우 이벤트 생성 시 등록된 리스트용 이미지.
   5. Profile 2,3 인 경우 참여정보, 당첨정보가 있으면 해당 정보도 함께 제공.
   6. 정렬 기준
      1. 이벤트포털 메뉴별 이벤트 목록 노출우선순위 값 오름차순
         1. Portal\_Menu\_Event\_Relation 테이블의 sortSequence 값 오름차순
      2. 노출우선순위 없는 경우
         1. 최근 등록순

#### 요청 주소

http://serverIP:port/EventPortalServer/getPortalEventList.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| subscriberId | String | 가입자 아이디  (가입자 정보조회, 참여/당첨, 쿠폰/xPush 연동시 사용할 이벤트포털 관련하여 공통으로 사용가능한 단일 가입자 아이디)  VOD시스템에서는 customer\_id  EPG, 쿠폰, xPUSH 시스템에서는 CAS\_ID | ● |  |
| soId | String | 가입자의 지역 정보. externalSoId 값 | ● |  |
| stbModelName | String | 셋탑 모델 명 | ● |  |
| portalMenuId | Int | 조회할 이벤트포털 메뉴 아이디.   * ~~입력 값이 없으면 이벤트포털 메뉴 중 sortSequence가 가장 낮은 portalMenuId를 이용~~ | ● |  |
| profile | Int | 결과 정보의 수준   * **(지원 프로파일 : 0, 1, 2, 3)** |  | 0 |
| sortType | String | * default : 노출우선순위 > 최근 등록순 |  | default |
| pageSize | Int | page당 이벤트의 개수  0이면 paging 기능을 사용하지 않는다 |  | 0 |
| pageIndex | Int | paging 기능 사용시 요청 page의 index  pageSize 값이 0이면 무시된다.  startItemIndex 값이 있으면 무시된다. |  | 0 |
| startItemIndex | Int | paging 기능 사용시  해당 index의 아이템부터 얻어온다  이 값이 있으면 pageIndex 값은 무시된다  이 값이 없으면 pageIndex 값을 이용하여 paging을 한다 |  |  |
| indexRotation | Bool | true인 경우, rotation 기능 사용. true/false  (pageSize의 수보다 결과 아이템 수가 모자라면 List의 가장 처음 아이템부터 추가하여 pageSize 수만큼 반환한다.) |  | false |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| totalCount | Int | 총 이벤트 수 |
| totalPage | Int | request의 pageSize에 따라 계산된 총 page 수  startItemIndex를 이용한 paging 처리시에는 무의미(-1) |
| eventList | *Event[]* | 이벤트 목록 |
|  |  |  |
|  |  |  |

#### 요청과 응답 예

### getPortalNoticeList

#### 상세 설명

1. 설명
   1. 이벤트포털의 공지사항 메뉴 진입 시 메뉴에 등록된 공지사항 목록 조회
2. 요청 상황
   1. 이벤트포털의 공지사항 메뉴 진입 시
3. 처리 내용
   1. 이벤트포털 공지사항 메뉴에 등록된 이벤트 목록
   2. 1달 이내에 등록된 공지사항만 조회
   3. 정렬 기준
      1. 노출우선순위 값 오름차순
         1. Portal\_Menu\_Notice\_Relation 테이블의 sortSequence 값 오름차순
      2. 노출우선순위 없는 경우
         1. 등록일 내림차순

#### 요청 주소

http://serverIP:port/EventPortalServer/getPortalNoticeList.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| portalMenuId | Int | 조회할 이벤트포털 메뉴 아이디.  ~~메뉴 아이디의 type이 Notice type이 아니면 예외처리.~~ | ● |  |
| sortType | String | * default : 노출우선순위 > 최근 등록순 |  | default |
| pageSize | Int | page당 이벤트의 개수  0이면 paging 기능을 사용하지 않는다 |  | 0 |
| pageIndex | Int | paging 기능 사용시 요청 page의 index  pageSize 값이 0이면 무시된다.  startItemIndex 값이 있으면 무시된다. |  | 0 |
| startItemIndex | Int | paging 기능 사용시  해당 index의 아이템부터 얻어온다  이 값이 있으면 pageIndex 값은 무시된다  이 값이 없으면 pageIndex 값을 이용하여 paging을 한다 |  |  |
| indexRotation | Bool | true인 경우, rotation 기능 사용. true/false  (pageSize의 수보다 결과 아이템 수가 모자라면 List의 가장 처음 아이템부터 추가하여 pageSize 수만큼 반환한다.) |  | false |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| totalCount | Int | 총 이벤트 수 |
| totalPage | Int | request의 pageSize에 따라 계산된 총 page 수  startItemIndex를 이용한 paging 처리시에는 무의미(-1) |
| noticeList | *Notice[]* | 공지사항 목록 |

#### 요청과 응답 예

### getMyEventList

#### 상세 설명

1. 설명
   1. 이벤트 포털 참여 이벤트 목록 정보
2. 요청 상황
   1. 이벤트 포털 참여 이벤트 메뉴 진입
3. 처리 내용
   1. 가입자가 참여 한 이벤트
   2. 노출 기간에 포함된 이벤트
   3. 사용 중 상태인 이벤트.(진행중지 포함)
   4. 타임세일 제외
   5. 우선순위
      1. 당첨 > 미당첨 > 응모완료 > N/N(목표달성형 참여 중 상태인 이벤트. 참여횟수 등의 값은 정렬기준에 포함하지 않고 참여중 상태인지만 반영.)
      2. 위 정렬기준에서 동일 상태 인 경우 : 이벤트 결과 최신 발표 순> 이벤트 참여 최신 순 > 이벤트 최신 등록 순 > 타이틀 가나다 순
   6. 각 이벤트 별 포함되어야 할 정보
      1. 아이디, 타이틀
      2. 리스트용 이미지
         1. 이벤트 생성 시 등록한 리스트용 이미지
      3. 최종 응모 날짜
         1. ~~자동 : 최종 구매/시청 날짜~~
         2. ~~수동 : 목표미달성 시 ()괄호 안에 최종 참여 날짜, 목표달성 시 ()괄호 없이 목표달성 날짜~~
      4. 당첨자 발표일
         1. ~~진행 중일 경우 빈칸~~
         2. ~~즉시 추첨의 경우 이벤트 결과 발표 된 날짜(?)~~
         3. ~~당첨자 발표 된 날짜~~
      5. 진행 현황
         1. ~~완료~~
         2. ~~진행 중(목표달성 수치/최종목표 수치)~~
      6. 상태
         1. ~~참여 중~~
         2. ~~응모완료(목표달성)~~
         3. ~~결과 확인(가입자가 직접 결과 확인을 해야 하는 경우???)~~
         4. ~~당첨/미당첨~~

#### 요청 주소

http://serverIP:port/EventPortalServer/getMyEventList.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| subscriberId | String | 가입자 아이디  (가입자 정보조회, 참여/당첨, 쿠폰/xPush 연동시 사용할 이벤트포털 관련하여 공통으로 사용가능한 단일 가입자 아이디)  VOD시스템에서는 customer\_id  EPG, 쿠폰, xPUSH 시스템에서는 CAS\_ID | ● |  |
| ~~soId~~ | ~~String~~ | ~~가입자의 지역 정보. externalSoId 값~~ | ~~●~~ |  |
| ~~stbModelName~~ | ~~String~~ | ~~셋탑 모델 명~~ | ~~●~~ |  |
| profile | Int | 결과 정보의 수준.  **프로파일 (2, 3)만 지원.** | ● | 3 |
| sortType | String | ~~default : 목표달성형 > 수동 경품 > 자동 경품~~ |  | default |
| pageSize | Int | page당 이벤트의 개수  0이면 paging 기능을 사용하지 않는다 |  | 0 |
| pageIndex | Int | paging 기능 사용시 요청 page의 index  pageSize 값이 0이면 무시된다  startItemIndex 값이 있으면 무시된다 |  | 0 |
| startItemIndex | Int | paging 기능 사용시  해당 index의 아이템부터 얻어온다  이 값이 있으면 pageIndex 값은 무시된다  이 값이 없으면 pageIndex 값을 이용하여 paging을 한다 |  |  |
| indexRotation | bool | true인 경우, rotation 기능 사용. true/false  (pageSize의 수보다 결과 아이템 수가 모자라면 List의 가장 처음 아이템부터 추가하여 pageSize 수만큼 반환한다.) |  | false |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| totalCount | Int | 총 이벤트 수 |
| totalPage | Int | request의 pageSize에 따라 계산된 총 page 수  startItemIndex를 이용한 paging 처리시에는 무의미(-1) |
| eventList | *event[]* | 이벤트 목록 |

#### 요청과 응답 예

### getEnrollReservationInfo

#### 상세 설명

1. 설명
   1. 특정 이벤트에 대한 참여 예약 정보 조회
2. 요청 상황
   1. 이벤트 수동참여를 위한 참여 예약정보 조회 필요 시.
      1. 예) 시청종료 후 이벤트 상세화면 진입 시
3. 처리 내용
   * + 1. EnrollReservationInfo 반환

#### 요청 주소

http://serverIP:port/EventPortalServer/getEnrollReservationInfo.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| subscriberId | String | 가입자 아이디  (가입자 정보조회, 참여/당첨, 쿠폰/xPush 연동시 사용할 이벤트포털 관련하여 공통으로 사용가능한 단일 가입자 아이디)  VOD시스템에서는 customer\_id  EPG, 쿠폰, xPUSH 시스템에서는 CAS\_ID | ● |  |
| eventId | Int | 이벤트 아이디 | ● |  |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| enrollReservationInfo | EnrollReservationInfo | 참여 예약 정보 |

#### 요청과 응답 예

### getEventInfo

#### 상세 설명

1. 설명
   1. 이벤트 포털의 이벤트 상세화면
2. 요청 상황
   1. 이벤트 포털에서 특정 이벤트 선택 시
   2. 일반 배너 영역에서 이벤트 선택 시
   3. 헬로보드에서 이벤트 프로모션 선택 시
   4. 에셋 상세화면에서 수동 이벤트 참여하기 선택 시
   5. ~~시청 종료 화면의 이벤트 배너에서 이벤트 확인하기 선택 시(구매/시청 정보 포함된 경우)~~
   6. ~~시청 종료 후 이벤트 상세 화면에서 다른 이벤트 상세 화면으로 진입 시(구매/시청 정보 포함)~~
3. 처리 내용
   * 1. 이벤트 상세 정보
     2. 이벤트포털 상세화면용 이미지
     3. ~~이벤트 대상 목록 및 대상 링크 정보~~
        1. ~~ActionTarget 중 구매/시청한 대상을 제외한 나머지 중에서 랜덤으로 N개 추출.~~
        2. ~~ActionTarget 조회 시 라이선스 유효한 대상만 포함.~~
        3. ~~구매 이벤트의 경우 구매 실행대상에서 조회~~
        4. ~~시청 이벤트의 경우 시청 실행대상에서 조회~~
     4. ~~개인정보 필요한 경품 당첨 시 개인 정보 입력 여부~~
     5. 참여 정보
        1. 최신 구매/시청 날짜 내림차순
     6. ~~당첨자 목록~~
        1. ~~혜택 우선순위 오름차순, 이름 오름차순~~
        2. ~~이름(김\*첨), 전화번호(010-\*\*\*\*-1234), 상품 내용~~
        3. ~~상품 수가 ALL인 혜택의 당첨자 제외.~~
     7. 당첨자 보기 버튼 활성화 여부
        1. 이벤트가 발표 완료
        2. 상품수가 ALL인 상품이 단독으로 존재 할 때 제외
           1. 상품 수 : 1, 2, 3, ALL -> 제외하지 않음
           2. 상품 수 : ALL -> 제외(ALL, ALL 도 제외)

#### 요청 주소

http://serverIP:port/EventPortalServer/getEventInfo.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| subscriberId | String | 가입자 아이디  (가입자 정보조회, 참여/당첨, 쿠폰/xPush 연동시 사용할 이벤트포털 관련하여 공통으로 사용가능한 단일 가입자 아이디)  VOD시스템에서는 customer\_id  EPG, 쿠폰, xPUSH 시스템에서는 CAS\_ID | ● |  |
| soId | String | 가입자의 지역 정보. externalSoId 값 | ● |  |
| stbModelName | String | 셋탑 모델 명 | ● |  |
| eventId | Int | 이벤트 아이디 |  |  |
| ~~asseId~~ | ~~String~~ | ~~에셋 아이디.~~  ~~“providerId|assetId” 형식~~ | ~~●~~  ~~assetId, productId 둘 중 하나는 필수. 둘 다 들어오는 경우 에셋, 상품에 대하여 한꺼번에 처리~~ |  |
| ~~ProductId~~ | ~~String~~ | ~~상품 productId~~ |  |
| ~~goodId~~ | ~~String~~ | ~~상품 goodId. goodId 값이 없는 경우 빈 값(“”)으로 처리.~~ | ~~●~~  ~~productId가 있을 때는 필수~~ |  |
| ~~productType~~ | ~~String~~ | ~~상품 타입.~~   * ~~SVOD~~ * ~~SVOD\_PACKAGE~~ * ~~BUNDLE~~ * ~~PACKAGE~~ * ~~FOD~~ * ~~RVOD~~ | ~~●~~  ~~productId가 있을 때는 필수~~ |  |
| ~~purchaseDomain~~ | ~~String~~ | ~~구매 도메인(수동 참여시 필요함)~~ |  |  |
| ~~purchaseId~~ | ~~String~~ | ~~구메 아이디~~ |  |  |
| ~~serviceDomain~~ | ~~String~~ | ~~시청 도메인~~ |  |  |
| ~~serviceId~~ | ~~String~~ | ~~시청 아이디~~ |  |  |
| profile | Int | 결과 정보의 수준(0~3) |  | 3 |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| event | *Event* | 이벤트 상세 정보 |
| subscriberEnroll DetailList | *SubscriberEnrollDetail[]* | 참여 상세 내역 |
| ~~eventActionTargetList~~ | *~~EventActionTarget[]~~* | ~~참여 가능 한 이벤트 실행 대상(에셋/상품) 목록 (이미 참여한 실행 대상 제외, 라이선스 유효성 체크, 랜덤 N개)~~ |
| ~~relatedEventList~~ | *~~RelatedEvent[]~~* | ~~요청 시 입력 받은 대상과 연관된 이벤트 목록 (참여한 이벤트 포함) 🡪 별도의 API로 분리~~ |
| ~~winnerList~~ | *~~Winner[]~~* | ~~당첨자 목록.~~   * ~~혜택 우선순위 오름차순, 이름 오름차순~~ * ~~이름(김\*첨), 전화번호(010-\*\*\*\*-1234), 상품 내용~~ * ~~상품 수가 ALL인 혜택의 당첨자 제외.~~ |
| isShowWinner | Boolean | 당첨자 보기 버튼 활성화 여부 |

#### 요청과 응답 예

### getRelatedEventListForActionTarget

#### 상세 설명

1. 설명
   1. 특정 참여대상 컨텐츠와 연관된 이벤트 조회
2. 요청 상황
   1. 시청 종료 후 이벤트 상세 화면에서 구매/시청한 콘텐츠와 연관 이벤트 조회 시
3. 처리 내용
   * 1. 에셋, 상품 아이디와 연관된 이벤트
     2. 노출대상에 포함되는 이벤트(SO, STBModel)
     3. 진행중인 이벤트
     4. 입력 받은 eventId에 해당하는 이벤트는 제외
     5. 에셋, 상품 정보가 함께 들어오는 경우 에셋 관련 이벤트 + 상품 관련 이벤트 합집합 목록에서 중복 제거한 결과
     6. 정렬
        1. 구매/시청 정보 있을 경우 해당 에셋이 포함된 이벤트 중 가입자가 참여 가능한 이벤트 목록
        2. 각 이벤트 별 참여 상태(참여 가능/참여 중/응모 완료)
        3. 정렬순서 : 우선순위 : 목표달성형 > 수동응모 : 경품응모 > 자동응모 : 자동응모 > 응모완료이벤트-최신참여 날짜 내림차순)
        4. 에셋, 상품 정보가 함께 들어오는 경우 에셋 관련 이벤트 + 상품 관련 이벤트 합집합 목록에서 중복 제거한 결과

#### 요청 주소

http://serverIP:port/EventPortalServer/getRelatedEventListForActionTarget.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| subscriberId | String | 가입자 아이디  (가입자 정보조회, 참여/당첨, 쿠폰/xPush 연동시 사용할 이벤트포털 관련하여 공통으로 사용가능한 단일 가입자 아이디)  VOD시스템에서는 customer\_id  EPG, 쿠폰, xPUSH 시스템에서는 CAS\_ID | ● |  |
| soId | String | 가입자의 지역 정보. externalSoId 값 | ● |  |
| stbModelName | String | 셋탑 모델 명 | ● |  |
| eventId | Int | 이벤트 아이디 | ● |  |
| asseId | String | 에셋 아이디.  “providerId|assetId” 형식 | ●  assetId, productId 둘 중 하나는 필수. 둘 다 들어오는 경우 에셋, 상품에 대하여 한꺼번에 처리 |  |
| ProductId | String | 상품 productId |  |
| goodId | String | 상품 goodId. goodId 값이 없는 경우 빈 값(“”)으로 처리. | ●  productId가 있을 때는 필수 |  |
| sortType | String | 정렬 타입   * “default”   + 참여상태 : 미참여 > 참여중 > 참여완료   + 미참여, 참여중 상태의 경우 : 목표달성형 > 수동경품응모 > 자동경품응모   + 참여완료 이벤트는 최신참여 날짜 내림차순 |  | “default” |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| relatedEventList | *Related Event[]* | 요청 시 입력 받은 대상과 연관된 이벤트 목록 (참여한 이벤트 포함) |
|  |  |  |

#### 요청과 응답 예

### getEventActionTargetList

#### 상세 설명

1. 설명
   1. 특정 이벤트에 연관된 대상 목록 및 링크 정보를 반환한다.
2. 요청 상황
   1. 이벤트 상세 화면 조회화면에서 이벤트 대상 선택 시
3. 처리 내용
   * 1. 이벤트 대상 목록 및 대상 링크 정보
        1. ActionTarget 중 구매/시청한 대상을 제외한 나머지 중에서 랜덤으로 N개 추출.
        2. 랜덤 추출 N개 값은 설정 값
        3. ActionTarget 조회 시 라이선스 유효한 대상만 포함.
        4. 구매 이벤트의 경우 구매 실행대상에서 조회
        5. 시청 이벤트의 경우 시청 실행대상에서 조회

#### 요청 주소

http://serverIP:port/EventPortalServer/getEventActionTargetList.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| eventId | Int | 이벤트 아이디 | ● |  |
| subscriberId | String | 가입자 아이디  (가입자 정보조회, 참여/당첨, 쿠폰/xPush 연동시 사용할 이벤트포털 관련하여 공통으로 사용가능한 단일 가입자 아이디)  VOD시스템에서는 customer\_id  EPG, 쿠폰, xPUSH 시스템에서는 CAS\_ID | ● |  |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| eventActionTargetList | *EventActionTarget[]* | 참여 가능 한 이벤트 실행 대상(에셋/상품) 목록 (이미 참여한 실행 대상 제외, 라이선스 유효성 체크, 랜덤 N개) |
|  |  |  |

#### 요청과 응답 예

### getWinnerList

#### 상세 설명

1. 설명
   1. 특정 이벤트의 당첨 정보 리스트를 반환한다.
2. 요청 상황
   1. 이벤트 상세 화면 조회화면에서 당첨자 목록 선택 시
3. 처리 내용
   * 1. 이벤트가 발표완료 된 상태
     2. 당첨자 목록
        1. 혜택 우선순위 오름차순, 이름 오름차순
        2. 이름(김\*첨), 전화번호(010-\*\*\*\*-1234), 상품 내용
        3. 상품 수가 ALL인 혜택의 당첨자 제외.

#### 요청 주소

http://serverIP:port/EventPortalServer/getWinnerList.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| eventId | Int | 이벤트 아이디 | ● |  |
| pageSize | Int | page당 당첨자의 개수  0이면 paging 기능을 사용하지 않는다 |  | 0 |
| pageIndex | Int | paging 기능 사용시 요청 page의 index  pageSize 값이 0이면 무시된다.  startItemIndex 값이 있으면 무시된다. |  | 0 |
| startItemIndex | Int | paging 기능 사용시  해당 index의 아이템부터 얻어온다  이 값이 있으면 pageIndex 값은 무시된다  이 값이 없으면 pageIndex 값을 이용하여 paging을 한다 |  |  |
| indexRotation | Bool | true인 경우, rotation 기능 사용. true/false  (pageSize의 수보다 결과 아이템 수가 모자라면 List의 가장 처음 아이템부터 추가하여 pageSize 수만큼 반환한다.) |  | false |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| winnerList | *Winner[]* | 당첨자 목록.   * 혜택 우선순위 오름차순, 이름 오름차순 * 이름(김\*첨), 전화번호(010-\*\*\*\*-1234), 상품 내용 * 상품 수가 ALL인 혜택의 당첨자 제외. |
| totalCount | Int | 총 당첨자 수 |
| totalPage | Int | request의 pageSize에 따라 계산된 총 page 수  startItemIndex를 이용한 paging 처리시에는 무의미(-1) |

#### 요청과 응답 예

### ~~getUncheckedWinEventList~~

당첨미확인 이벤트 목록 조회 API 삭제

### getUncheckedWinEventCount

#### 상세 설명

1. 설명
   1. 확인하지 않은 당첨 이벤트 목록 조회
2. 요청 상황
   1. 이벤트 포털 참여 이벤트 메뉴 진입 시 확인하지 않은 당첨 이벤트 있는 경우 알림 팝업 노출
3. 처리 내용
   1. 노출 기간 인 이벤트
   2. 당첨된 이벤트 중 winConfirmStatus 가 미확인 상태인 이벤트 수

#### 요청 주소

http://serverIP:port/EventPortalServer/getUncheckedWinEventCount.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| subscriberId | String | 가입자 아이디  (가입자 정보조회, 참여/당첨, 쿠폰/xPush 연동시 사용할 이벤트포털 관련하여 공통으로 사용가능한 단일 가입자 아이디)  VOD시스템에서는 customer\_id  EPG, 쿠폰, xPUSH 시스템에서는 CAS\_ID | ● |  |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| totalCount | Int | 총 이벤트 수 |

#### 요청과 응답 예

## 당첨자 관련 API

### doPrizeReceipt

#### 상세 설명

1. 설명
   1. 이벤트의 경품 수령 확인
2. 요청 상황
   1. 당첨된 이벤트 혜택 중 개인정보 수집이 필요한 혜택에 대하여 개인정보를 입력하는 경우
   2. ~~즉시 지급 경품 당첨 시(쿠폰/TV코인 등) : 이벤트 포털 상세화면 진입 시~~
   3. ~~실물 경품 당첨 시~~
      1. ~~번호 미 수집하는 경우 : 경품 받기 버튼 선택 시~~
      2. ~~번호 수집하는 경우 : 경품 받기 버튼 선택 후 전화번호 수집 입력 동의 시~~
3. 처리 내용
   1. 수령확인 상태 및 개인정보를 저장.
   2. EventEnrollServer의 “수령확인 API” 연동 필요.
      1. 응답 실패에 대해서는 설정 값으로 재시도
      2. 응답 값이 실패인 경우에는 재시도 하지않음
   3. 이미 수령확인 된 경우 “이미 처리되었음” 반환하고 다른 처리 하지 않음.
   4. eventId, subscriberId에 대하여 복수의 winner 정보가 있는 경우 그 중 개인정보 수집이 필요한 winner 정보에 대하여 모두 처리.(추후 중복당첨 고려)

#### 요청 주소

http://serverIP:port/EventPortalServer/doPrizeReceipt.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| subscriberId | String | 가입자 아이디  (가입자 정보조회, 참여/당첨, 쿠폰/xPush 연동시 사용할 이벤트포털 관련하여 공통으로 사용가능한 단일 가입자 아이디) | ● |  |
| eventId | Int | 이벤트 아이디 | ● |  |
| phone | String | 전화번호 | ● |  |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |

#### 요청과 응답 예

### doWinConfirm

#### 상세 설명

1. 설명
   1. 당첨된 이벤트의 상세내역 확인 시 당첨확인 상태(winConfirmStatus) 를 당첨확인 상태로 변경.
2. 요청 상황
   1. 당첨된 이벤트 중 상세화면 조회하지 않은 이벤트의 상세화면 진입 시
3. 처리 내용
   1. 당첨 확인 상태를 “당첨확인” 상태로 변경.
   2. EventEnrollServer의 당첨확인 API 연동 필요.
      1. 연동 실패 시 당첨확인 실패 처리.
      2. 응답 실패에 대해서는 설정 값으로 재시도
      3. 응답 값이 실패인 경우에는 재시도 하지 않음
   3. 이미 처리된 경우 EventEnrollServer 연동 필요 없이 “성공”.
   4. eventId, subscriberId에 대하여 복수의 winner 정보가 있는 경우 모두 처리(추후 중복당첨 고려)

#### 요청 주소

http://serverIP:port/EventPortalServer/doWinConfirm.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| subscriberId | String | 가입자 아이디  (가입자 정보조회, 참여/당첨, 쿠폰/xPush 연동시 사용할 이벤트포털 관련하여 공통으로 사용가능한 단일 가입자 아이디) | ● |  |
| eventId | Int | 이벤트 아이디 | ● |  |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |

#### 요청과 응답 예

### ~~getEnrollEventCount~~

참여 이벤트 수 조회 API 삭제

## 외부 연동을 위한 일반 이벤트 API

### getEventList

#### 상세 설명

1. 설명
   1. 일반적인 이벤트 조회 API
2. 요청 상황
   1. CUP Editor 에서 이벤트 조회 시
   2. 일반적이 이벤트 정보 조회 시
3. 처리 내용
   1. eventId 조회 시 아이디 일치하는 이벤트
   2. title 조회 시 검색어 포함하는 이벤트
   3. licenseStartTime 조회 시 검색 시작일 이후에 진행기간이 포함되는 이벤트
      1. actionEndTime >= licenseStartTime
   4. licenseStartTime, licenseEndTime 조회 시 검색 기간에 진행기간이 포함되는 이벤트.
      1. licenseEndTime 만 있으면 예외처리.
   5. 나머지 검색 조건에 해당 하는 이벤트
   6. 정렬 기준
      1. 아이디 순
      2. 타이틀 순
      3. 생성일 순
      4. 이벤트 노출 시작일, 종료일 순
      5. 이벤트 진행(라이선스) 시작일, 종료일 순

#### 요청 주소

http://serverIP:port/EventPortalServer/getEventList.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| eventId | Int | 이벤트 아이디. “=” 검색 |  |  |
| Title | String | 이벤트 타이틀 검색어.   * Like 검색 |  |  |
| eventType | String | UI 이벤트 타입   * 0 : 경품응모 * 1 : 스탬프 * 2 : N+1 * 3 : 합산페이백 |  |  |
| ~~autoEnroll~~ | ~~String~~ | ~~자동응모 여부~~   * ~~0 : 자동응모~~ * ~~1 : 수동응모~~ |  |  |
| searchActionStartTime | String | 이벤트 진행기간의 검색 시작일.   * yyyyMMddHHmm 형식 * 검색 시작일 이후에 이벤트 진행기간이 포함되는 이벤트 조회. * 진행종료일 >= searchActionStartTime | ●  licenseEndTime이 있을 경우 필수 |  |
| searchActionEndTime | String | 이벤트 진행기간에 대한 검색 종료일.  반드시 searchActionStartTime 값과 함께 있어야 함. (searchActionEndTime 만 있으면 예외 처리)   * yyyyMMddHHmm 형식 * 검색기간에 이벤트 진행기간이 포함되는 이벤트 조회 * 진행종료일 >= searchActionStartTime and 진행시작일 <= searchActionEndTime |  |  |
| eventStatus | String | 이벤트 상태.   * 10 : 이벤트 전 * 20 : 이벤트 중 * 30 : 마감임박 D-3 * 40 : 추첨 대기(진행기간 종료 후 추첨 전까지) * 50 : 추첨 완료(ALL 이외 나머지 상품의 당첨자가 모두 결정 된 경우) * 51 : 상품소진완료(“참여 혜택 당첨 완료 시, 이벤트 종료“ 설정된 이벤트의 상품이 소진 된 경우) * 70 : 발표 완료(발표 버튼을 누른 경우) * 80 : 진행 중지 |  |  |
| Profile | Int | 결과 정보의 수준   * **(지원 프로파일 : 0~~, 1~~)** |  | 0 |
| sortType | String | * titleAsc * titleDesc * ~~idAsc~~ * ~~idDesc~~ * ~~creationTimeAsc~~ * ~~creationTimeDesc~~ * actionStartTimeAsc * actionStartTimeDesc * ~~diaplayEndTimeAsc~~ * ~~displayEndTimeDesc~~ |  | titleAsc |
| pageSize | Int | page당 이벤트의 개수  0이면 paging 기능을 사용하지 않는다 |  | 0 |
| pageIndex | Int | paging 기능 사용시 요청 page의 index  pageSize 값이 0이면 무시된다  startItemIndex 값이 있으면 무시된다 |  | 0 |
| startItemIndex | Int | paging 기능 사용시  해당 index의 아이템부터 얻어온다  이 값이 있으면 pageIndex 값은 무시된다  이 값이 없으면 pageIndex 값을 이용하여 paging을 한다 |  |  |
| indexRotation | Bool | true인 경우, rotation 기능 사용. true/false  (pageSize의 수보다 결과 아이템 수가 모자라면 List의 가장 처음 아이템부터 추가하여 pageSize 수만큼 반환한다.) |  | false |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| totalCount | Int | 총 이벤트 수 |
| totalPage | Int | request의 pageSize에 따라 계산된 총 page 수  startItemIndex를 이용한 paging 처리시에는 무의미(-1) |
| eventList | *event[]* | 이벤트 목록 |

#### 요청과 응답 예

|  |
| --- |
| <http://serverIP:port/EventPortalServer/getEventList.json?title>=할인&searchActionStartTime=201502010000&searchActionEndTime=201502102359&profile=0&sortType=titleAsc&pageSize=10&pageIndex=0 |
| response  {  “resultCode” : 100,  “errorString” : “”,  “totalCount” : “25”,  “totalPage” : “3”,  “eventList” : [  {  “eventId” : 324,  “title” : “RVOD 할인 이벤트”,  “description” : “”,  “eventType” : “0”,  “isAutoEnroll” : “true”,  “status” : “10”,  “creationTime” : “2015-01-15 11:53:49”,  “lastModifyTime” : “2015-01-15 11:53:49”,  “displayStartTime” : “2015-01-21 00:00”,  “displayEndTime” : “2015-03-15 23:59”,  “actionStartTime” : “2015-02-01 00:00”,  “actionEndTime” : “2015-02-28 23:59”  },  {  “eventId” : 231,  “title” : “최신작 할인 이벤트”,  “description” : “”,  “eventType” : “1”,  “isAutoEnroll” : “true”,  “status” : “10”,  “creationTime” : “2015-01-15 11:53:49”,  “lastModifyTime” : “2015-01-15 11:53:49”,  “displayStartTime” : “2015-01-01 00:00”,  “displayEndTime” : “2015-02-28 23:59”,  “actionStartTime” : “2015-01-11 00:00”,  “actionEndTime” : “2015-02-20 23:59”  }  ]  } |

예. getEventList – JSON 응답

### getAvailableEventList

#### 상세 설명

1. 설명
   1. 유효한 이벤트 리스트 조회 API
2. 요청 상황
   1. CUP Editor 이벤트를 검색해서 특정 영역에 노출할 이벤트를 설정하여 저장할 시 - 조회한 이벤트들이 현재시간, 현재 가입자 기준으로 유효한지를 확인할 경우
3. 처리 내용
   1. 입력 받은 이벤트 아이디 목록 중 현재 시점에서 참여할 수 있는 이벤트 목록만 반환
   2. 반환하는 이벤트 목록은 프로파일 0
   3. 진행기간 이내
   4. 삭제 상태 아니고, 중지상태 아닌 이벤트
   5. so, stbModel이 노출룰에 포함

#### 요청 주소

http://serverIP:port/EventPortalServer/getAvailabeEventList.[xml | json]

#### 요청 변수 및 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름 | 타입 | 설명 | 필수여부 | 기본값 |
| eventIds | String | 이벤트 아이디리스트  Integer|Integer|Integer 형식  |로 이벤트 아이디를 구분(이벤트 아이디는 Integer 형) | ● |  |
| soId | String | 가입자의 지역 정보. externalSoId 값 | ● |  |
| stbModelName | String | 셋탑 모델 명 | ● |  |

#### 응답 항목 및 설명

|  |  |  |
| --- | --- | --- |
| 이름 | 타입 | 설명 |
| resultCode | Int | 결과 코드 |
| errorString | String | 에러 내용 |
| eventList | *event[]* | 이벤트 목록 |

#### 요청과 응답 예

|  |
| --- |
| <http://serverIP:port/EventPortalServer/getEventList.json?title>=할인&searchActionStartTime=201502010000&searchActionEndTime=201502102359&profile=0&sortType=titleAsc&pageSize=10&pageIndex=0 |
| response  {  “resultCode” : 100,  “errorString” : “”,  “eventList” : [  {  “eventId” : 324,  “title” : “RVOD 할인 이벤트”,  “description” : “”,  “eventType” : “0”,  “isAutoEnroll” : “true”,  “status” : “10”,  “creationTime” : “2015-01-15 11:53:49”,  “lastModifyTime” : “2015-01-15 11:53:49”,  “displayStartTime” : “2015-01-21 00:00”,  “displayEndTime” : “2015-03-15 23:59”,  “actionStartTime” : “2015-02-01 00:00”,  “actionEndTime” : “2015-02-28 23:59”  },  {  “eventId” : 231,  “title” : “최신작 할인 이벤트”,  “description” : “”,  “eventType” : “1”,  “isAutoEnroll” : “true”,  “status” : “10”,  “creationTime” : “2015-01-15 11:53:49”,  “lastModifyTime” : “2015-01-15 11:53:49”,  “displayStartTime” : “2015-01-01 00:00”,  “displayEndTime” : “2015-02-28 23:59”,  “actionStartTime” : “2015-01-11 00:00”,  “actionEndTime” : “2015-02-20 23:59”  }  ]  } |

예. getEventList – JSON 응답

# 결과 코드 (Result Code)

|  |  |
| --- | --- |
| 100 | 성공 (Success) |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |