

모바일메시지서비스

【 가온사용자매뉴얼 】

2022.07



개정이력

버전	개정일자	개정사유	개정내역	작성자	승인자
1.0	2021.08.20	제정	최초 제정	이준희	박재아
1.1	2021.09.08	추가	옵션 추가	이준희	
1.2	2021.09.10	수정	1. 옵션 내용 수정 2. TEMPLATE_CODE 테이블 설명 추가 3. APP_ERROR_CODE 테이블 삭제	이준희	
1.3	2021.10.19	수정	1. 동보 발송 개수 옵션 추가 2. POSTGRESQL 스키마 추가 3. 사용법 변경에 따른 수정	이준희	
1.4	2021.11.22	추가	TOSS 발송 시 주의점 추가	이준희	
1.5	2021.12.21	추가	버튼 사용법 추가	이준희	
1.6	2022.01.06	수정	STATE 설명 수정 안 쓰는 컬럼 표시	이준희	
1.7	2022.02.21	삭제	TOSS 관련 내용 삭제	이준희	
1.8	2022.03.23	추가	로그테이블 이관 조건 설명	이준희	
1.9	2022.04.12	수정	1. 오타수정 2. CUR_STATE 조건 변경	이준희	
1.10	2022.04.27	추가	TEMPLATE_CODE테이블 설명	이준희	
1.11	2022.05.27	추가	발송 불가 메시지 에러코드 추가	이준희	
1.12	2022.06.21	추가	1. 발송 제한시간 옵션 추가 2. 로그테이블 이관 여부 옵션 추가	이준희	
1.13	2022.06.30	추가	1. ALTIBASE 스키마 추가 2. 상태코드 9인경우 에러코드 설명 추가	이준희	
1.14	2022.07.07	수정	1. 메시지 상태 코드 수정 2. 가온 에러 수정	이준희	

개정이력

버전	개정일자	개정사유	개정내역	작성자	승인자
1.15	2022.12.13	수정	<ul style="list-style-type: none"> - 가온 모듈 실행 옵션 설명 추가 - 테이블 생성 쿼리에서 각 테이블별 컬럼 설명이 몇 페이지에 있는지 추가 	이준희	

For NT, Unix, LINUX with JDBC

사용자 매뉴얼

version 1.15

Copyright © 2021, NanalSMI Inc. All rights reserved.

이 프로그램은 핵, 항공, 대량 수송 수단, 의료 또는 기타 본질적으로 위험한 상황에서 사용하도록 개발된 것이 아닙니다. 따라서 그런 목적으로 사용된 경우, 사용자는 응용 프로그램의 안전한 사용을 보장하기 위한 모든 적절한 안전 조치, 백업, 대비 및 기타 조치를 반드시 취해야 합니다. 프로그램이 이러한 목적으로 사용되었을 경우 모바일메시지서비스는 이러한 프로그램 사용으로 인한 피해를 책임지지 않습니다.

이 프로그램(소프트웨어와 설명서 포함)은 저작권법, 특허 및 기타 지적재산권 관련 법규에 의해 보호됩니다. 이 프로그램을 리버스 엔지니어링하거나 분해하거나 또는 역 컴파일 하는 것은 금지되어 있습니다. 또한 이 프로그램에서 사용되는 프로토콜을 리버스 엔지니어링으로 분석하는 것도 금지되어 있습니다.

이 문서의 내용은 사전 공지 없이 변경될 수 있습니다. 이 문서의 어떤 부분도 어떠한 목적이든 모바일메시지서비스의 허가 없이 기계적, 전자적 또는 기타 임의의 수단을 통해 배포되거나 복사할 수 없습니다.

1. 개요

1.1 가온 이란?

- Gaon은 모바일메시지서비스의 앱 기반 메시지 게이트웨이 서비스를 사용하기 위한 이용기관이 기관의 시스템과 쉽게 연동할 수 있도록 도와주는 앱 기반 메시지 게이트웨이 도우미로서, 기관의 시스템에 설치된다.

1.2 이용기관 시스템과 연동

- Gaon의 실행을 위해서는 Database가 반드시 필요하며, Gaon과의 연동은 한 가지 방법으로 가능하다.
- 가. Database에 Record Insert

1.3 용어

- BizM 게이트웨이
앱 기반 메시지 게이트웨이를 말하며, 메시지를 보내면 메시지가 전달될 앱 플랫폼 사업자에게 전달하고, 앱 플랫폼 사업자로부터 메시지 전송 결과를 받아 되돌려 주는 일을 하는 서버
- JDBC
자바 프로그램 내에서 데이터베이스 질의문 즉, SQL을 실행하기 위한 자바 API(application programming interface)이다.

2. Gaon 요구사항

2.1 System

- 필요한 HDD는 100만 건당 평균적으로 500 MB 를 차지하므로 월 50만 건씩 1년을 보관한다고 하면 총 600만 건이므로 3 GB정도를 생각하면 된다. CPU는 Pentium 200 이상, 전용선은 56 KB 이상이면 된다.
- Gaon을 구동하기 위해 모바일메시지서비스 BizM 게이트웨이에 System OS Version, DB Version, JDK version을 알려주어, 해당 Binary File 및 Gaon Binary File을 받는다.

2.2 Database

- 이용기관 System과 연동을 위해 ID, Password 등을 미리 알아야 한다. 또한 DB사용을 위해서는 JDBC의 URL과 JDBC Driver Type 을 알아야 한다.

● 연동할 수 있는 Database ID, Password

- 기존에 사용하던 ID를 사용하거나 새로운 ID를 만들어서 사용한다. ID 생성 방법은 DBMS의 매뉴얼을 참고하기 바란다.

● Database

- Gaon이 사용할 Database를 만들어 준다. 기존에 사용하던 것을 사용해도 무방하나 Gaon용으로 따로 만들어 사용하는 것을 권장한다.

2.3 모바일메시지서비스 BizM 게이트웨이 접속 환경

- 실행하기에 앞서 모바일메시지서비스 앱 기반 메시지 게이트웨이에 접속 할 수 있는 환경을 미리 알아야 하는데, 다음과 같다. 이 사항은 Gaon의 binary를 받을 때 같이 받는 것이므로, 혹시 빠져있다면 다시 연락하여 받도록 한다.

● BizM 게이트웨이 IP, Port

- 네트워크 상의 연결을 확인하기 위한 정보이다. 이 값이 있어야 Gaon이 실행 시에 접속할 서버의 물리적 위치를 찾을 수 있다. 모바일메시지서비스 측에서 받은 IP와 Port로 Telnet연결을 시도해서 연결이 되면 정상적이라고 보면 된다.

● BizM 게이트웨이 ID, Password

- 설치된 Gaon은 Server에서 인증이 되어 있어야 한다. 각 Gaon은 고유한 ID와 Password를 부여 받게 된다. 이 값이 세팅 되지 않으면 접속 후 인증 에러를 내며 종료된다.

3. 설치

3.1 배포판

Binary 배포 판에는 다음과 같은 파일들이 들어 있다.

● gaon.jar

주 실행 파일로서 아래에 나타난 내부 실행 파일들을 구동하고 감시하는 역할을 하며, 다음의 기능을 포함하고 있다.

- BizM 게이트웨이에 접속하여 앱 기반 메시지를 망으로 송신하는 기능
- BizM 게이트웨이에 접속하여 앱 기반 플랫폼에서 돌아온 전송 결과들을 수집하는 기능
- BizM 게이트웨이로부터의 전송결과가 실패인 경우, 설정(실패인 경우 문자 자동 전송)에 따라 메시지 내용 길이에 의해 SMS/LMS로 자동 전송 처리 하는 기능

● setting.yaml.sample

setting.yaml파일을 만들기 위한 참고 파일로서, 이 문서에서 언급하는 구성 방식에 대한 샘플이 들어 있는 파일이다.

3.2. 설치

적당한 디렉터리 하나를 만든 후 배포파일의 압축을 푼다. 배포파일은 자동실행 zip 또는 그냥 zip파일로 제공되므로, winzip등의 프로그램을 사용하여 풀거나 그냥 실행하면 된다.

이후, setting.yaml.sample을 setting.yaml로 복사한 후, 일부 설정들(BizM 게이트웨이 IP, Port, ID, Password, DB 설정)을 수정한 후, gaon.jar 파일을 실행하면 된다.

setting.yaml의 설정 방법은 5장을 참고하기 바란다. 관리 편의를 위해, 백그라운드 실행용으로 Gaon을 실행 한다.

리눅스, UNIX(HP-UX, Solaris) 일 경우 실행 방법

```
$ nohup java -jar gaon.jar /home/setting.yaml(생략가능, default: ./setting.yaml) > /dev/null &
```

윈도우일 경우 실행하는 방법

```
C:\W>javaw -jar gaon.jar /home/setting.yaml(생략가능, default: ./setting.yaml)
```

4. Process 구조

4.1 프로세스 블록 다이어그램

Gaon은 실행 시 6개의 쓰레드로 이루어진다.

● Main

메인스레드로 GaonSender, GaonReceiver, GaonLMSSender, GaonDBLogger 쓰레드를 실행하고, 주기적으로 스레드가 작동 중인지 확인합니다.

● GaonSender

주기적으로 DB를 감시하며, DB에서 보낼 메시지를 Select해서 BizM 게이트웨이로 보낸다.

● GaonReceiver

1. 주기적으로 DB를 감시하여 BizM게이트웨이로 부터 전송 결과를 받아야 하는 데이터가 존재하는 경우, 서버에 연결해서 데이터를 요청 후 수신하여 GaonResultProcessor로 전달합니다.
2. GaonResultProcessor를 실행 시키며, 주기적으로 GaonResultProcessor가 실행 중인지 확인합니다.

● GaonResultProcessor

Receiver에서 받는 결과를 APP_SEND_DATA의 해당 메시지 행에 결과 값을 갱신합니다.

● GaonLMSSender

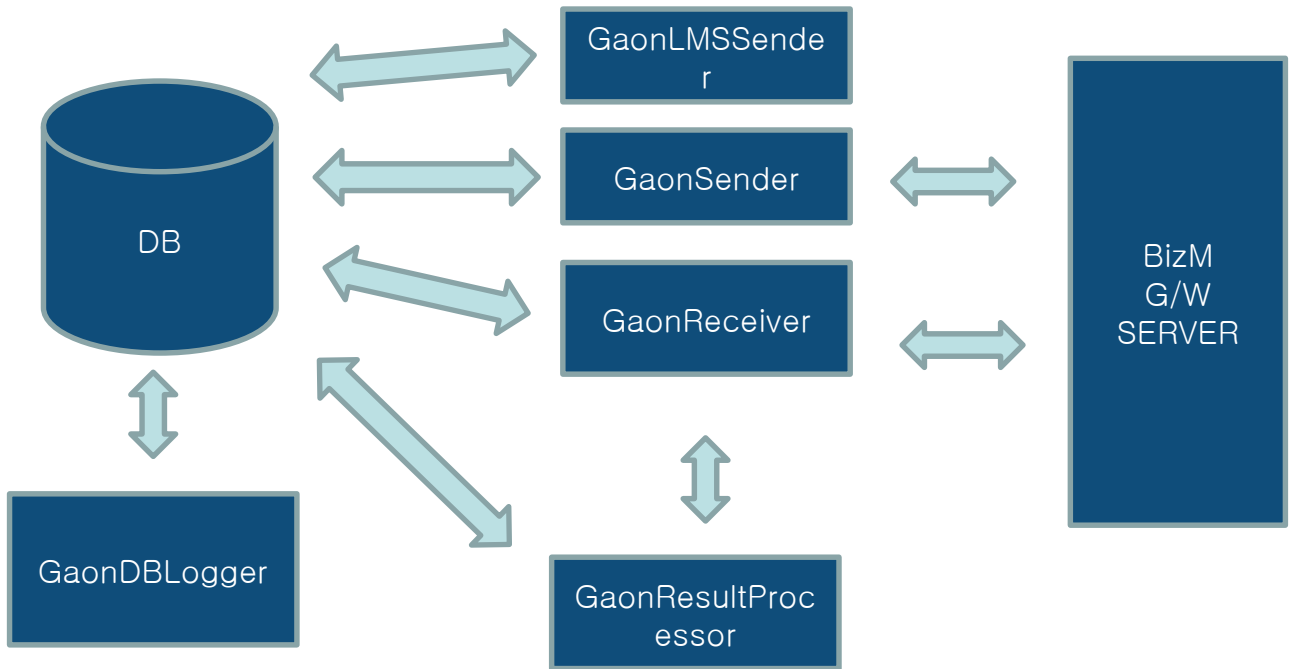
주기적으로 DB를 감시하여 문자로 발송해야 하는 대상이 있는 경우, NURI의 테이블에 입력해줍니다.

● GaonDbLogger

APP_SEND_DATA의 테이블에 있는 데이터 중 로그 테이블로 이관할 수 있는 데이터를 로그 테이블로 이관시키는 역할을 합니다.

4. Process 구조

4.1 프로세스 블록 다이어그램



5. 환경설정파일 작성

Gaon은 setting.yaml이라는 환경 설정 파일이 있어야 하며, 실행 시 다음과 같이 지정 해주어야 한다.

※ 각 설정은 배포된 Gaon 버전에 따라 setting.yaml 설정 파일에서 생략(기본 설정) 또는 누락되어 있을 수 있다.

5.1 Gaon 설정

DB 설정 정보

dbName

DBMS의 종류를 설정한다. ORACLE, MSSQL, MYSQL, CUBRID, TIBERO, POSTGRESQL 中 1

dbDriverName, dbUrl

JDBC driver type 과 url을 설정한다. JDBC드라이버 설정, 구문이 복잡하므로 위에 있는 setting.yaml.sample 파일 내부의 예제를 복사해서 설정할 것을 권장한다.

dbUser, dbPassword

dbUser와 dbPassword 는 각각 DB User name과 Password를 설정해 둔다.

Host Info (접속 정보)

gwIp, gwPort

모바일메시지서비스 BizM 게이트웨이에서 앱 기반 메시지 전송을 원하는 업체에 제공하며, Gaon이 접속할 게이트웨이의 ip와 port번호 이다.

gwClientId, gwClientPw

모바일메시지서비스 BizM 게이트웨이에서 앱 기반 메시지 전송을 원하는 업체에 제공하며, 인증 시에 꼭 필요하다.

5. 환경설정파일 작성

■ 누리 사용을 위한 설정 정보(알림톡 실패 시 SMS/LMS 처리하기 위한 필수 설정)

● msgDataSeq, msgContentsInfoSeq

앱메시지 발송 실패 시 문자로 처리할 때, NURI의 NURI_MMS_CONTENTS_INFO 테이블과 NURI_MSG_DATA테이블에 seq 컬럼에 값을 입력하는 정책으로 AUTO, DEFAULT, 시퀀스명 3종류가 입력 가능하다.

기존 MSG_DATA테이블에서 사용하는 시퀀스 이름으로, nuri.conf의 db_sequence_name.XXXXXXX에서 정의한 이름과 동일하게 입력하는 것을 권장함

- AUTO인 경우, CONT_SEQ, MSG_SEQ에 설정된 AUTOINCREMENT 속성을 이용한다.
- DEFAULT인 경우 APP_SEND_CONTENTS의 PACK_UNIQUEKEY 컬럼의 값을 NURI_MMS_CONTENTS_INFO의 CONT_SEQ에 복사하고, APP_SEND_DATA의 MSG_SEQ컬럼의 값을 NURI_MSG_DATA의 MSG_SEQ 컬럼으로 복사한다.
- 그 외 시퀀스 명을 입력한 경우, 해당 시퀀스를 이용하여 CONT_SEQ, MSG_SEQ에 값을 입력한다

■ 사용 테이블 명을 변경(가온 테이블명)

● appSendDataTableName

앱 기반 메시지 발송 및 결과 정보 데이터를 저장하기 위한 테이블 명

● appSendContentsTableName

앱 기반 메시지 템플릿 및 세부 내용 정보 데이터를 저장하기 위한 테이블 명

● templateCodeTableName

템플릿 코드 관리를 위한 테이블 명

● appSendDataLogTableName

앱 기반 메시지 발송을 위한 테이블의 로그 테이블 명

5. 환경설정파일 작성

■ 사용 테이블 명을 변경(누리 테이블명)

● msgDataTableName

기존 누리에서 사용하고 있는 nuri_msg_data 테이블 명

● msgContentsInfoTableName

기존 누리에서 사용하고 있는 nuri_mms_contents_info 테이블 명

■ 알림톡 실패 시 SMS/LMS 자동 발송 여부

● useLms

알림톡 발송 실패 시 SMS/LMS 발송 여부 (yes|no)

● copyCustomColumnWhenSendLms

문자발송시 사용자정의컬럼의 데이터를 같이 복사해서 입력할 것인지 여부

■ 모듈 실행 LOG 기록 범위

● loggerLevel

모듈을 실행 해서 나오는 로그의 기록 범위

■ 테이블 로그 기록

● logRealTime

yes - 5초 주기로 APP_SEND_DATA테이블을 검사해, LOG테이블로 이관

no - 5분 주기로 APP_SEND_DATA테이블을 검사해, LOG테이블로 이관

● makeLogMode

데이터를 로그 테이블로 이관할 시 하나의 로그 테이블로 만들 것인지,

로그테이블_YYYYMM 형식으로 분리해서 이관할지에 대한 설정 (DEFAULT, ONE)

● isTransferLog

모듈을 실행 해서 나오는 로그의 기록 범위

5. 환경설정파일 작성

■ 서버에 재접속 시도

● **socketConnectionTimeout**

서버 접속 타임아웃 시간 (단위: 초)

● **retryTime**

서버에 접근 실패 시, 다음 재시도까지 걸리는 시간 (단위 : 분)

● **retryCount**

서버에 재접속을 시도할 횟수. 이 횟수를 초과하는 경우 모듈이 종료됨
0을 입력시 무한 반복

● **sendLmsWhenRetryEnd**

retryCount만큼 재접속 시도 후에도 재접속 실패시 문자로 보낼지 여부.
Yes는 문자보내고 다시 재접속 시도, no로 설정시 모듈 실행 종료

● **sendLmsWhenConnectionFail**

서버에 접속 실패 시, 재시도를 하지 않고 바로 SMS/LMS로 발송할 것인지 여부.
YES로 설정하면 retryTime, retryCount 설정은 무시됨.

5. 환경설정파일 작성

■ 사용자 정의 컬럼, 인덱스

● **maxCustomColumnIndexCount**

사용자가 추가할 수 있는 최대 컬럼과 인덱스 개수 (최대 10개)

● **customColumnForAppSendData**

사용자 정의 컬럼 목록.

설정 방법)

customColumnForAppSendData:

- [컬럼 명] [컬럼 타입]
- [컬럼 명] [컬럼 타입]

● **customIndex**

사용자 정의 인덱스 목록.

설정방법)

customIndex:

- indexName: [인덱스 명]
- columns: [컬럼 명] [ASC|DESC], [컬럼 명]

5. 환경설정파일 작성

■ 로그 파일 저장 기간

○ logSaveTerm

일정기간 지난 로그파일을 삭제함 (단위 : 일)

○ logFilePath

로그파일을 저장할 경로

■ 앱 메시지 발송

○ tranTerm

앱 메시지 발송을 위해 DB에 접근하는 딜레이 (단위 : 초)

○ beforeTime

앱 메시지 발송을 위해 DB에 접근할 때 몇 시간이전 데이터까지 읽어올 것인가

○ duplicatePhoneCount

하루에 한 번호 당 보낼 수 있는 최대 건수.

0으로 설정 시 무제한

■ 타임아웃

○ timeoutLimit

메시지 전송 후 해당 설정 시간 동안 응답을 받지 못하면 타임아웃을 발생시킨다.(단위 : 분)

최소값은 30분

■ 발송량 관리

○ groupMessageCount

동보 발송 시, 한 묶음으로 보낼 수 있는 최대 연락처 개수

최소 10 ~ 1000값 만큼 설정 할 수 있다.

5. 환경설정파일 작성

■ 발송 제한 시간

○ noSendTimeStart

발송 제한 시작시간. noSendTimeEnd와 같은 시간일 경우 발송 제한 비활성화

○ noSendTimeEnd

발송 제한 종료시간. noSendTimeStart와 같은 시간일 경우 발송 제한 비활성화

■ 문자 전환 시 머리말 포함 여부

○ lmsIncludeHeader

발송 실패로 문자 전환 시 템플릿에 있는 머리말을 문자 메시지 내용에 포함 시킬 것인지 여부

■ 문자 전환 대상 에러 필터

○ excludeLmsError

발송 실패로 문자 전환시 해당 에러코드는 문자로 전환하지 않도록 설정
입력값 예) ["3011", "3012"]

6. 이용기관시스템과 연동

연동 방법은 메시지 전송 연동과 결과 확인 연동이 있다. 대개 이용기관의 특성에 따라, 이용기관에서 메시지를 보낼 용도로 사용할 때는 메시지 전송 연동만 하며, 이용기관의 회원들이 메시지를 보낼 때는 전송 연동과 결과 확인 연동을 모두 하게 된다.

6.1 Gaon이 사용하는 테이블

테이블명	역할
APP_SEND_DATA	모든 전송요구가 수집되는 테이블로서, 전송 요구와 결과 대기 등이 일어나는 테이블이다
APP_SEND_CONTENTS	모든 전송에 대한 템플릿 코드 및 메시지 내용에 대해 저장되어 있는 테이블이다
TEMPLATE_CODE	템플릿 코드 관리를 위한 테이블

6.2 전송 연동

전송하는 방법은 다음과 같다.

DB – Insert

주로, 웹을 통해 메시지요청을 할 경우 웹 페이지에서 DB에 접속하여 메시지를 전송할 때 사용한다.

6.2.1.DB-Insert

Gaon은 실행되면 바로 트랜잭션 테이블(APP_SEND_DATA, APP_SEND_CONTENTS)이 존재하는지 확인하여, 없을 경우 다음과 같은 스키마로 테이블을 생성하게 된다. 또한, 앱 기반 메시지(예. 알림톡, 네이버톡톡, 토스 등) 등 전송이 실패한 경우, SMS/LMS 자동전송 수행을 위하여 Nuri가 보낼 수 있도록 Nuri가 사용하는 문자테이블에 데이터를 넣어줍니다.

6. 이용기관시스템과 연동 DB Schema

■ MYSQL(각 테이블별 컬럼의 설명은 31p 참조)

```
CREATE TABLE IF NOT EXISTS APP_SEND_DATA
(
    MSG_SEQ                INT AUTO_INCREMENT        NOT NULL,
    REQ_SEND_DATE          CHAR(14)                  NOT NULL,
    PACK_UNIQUEKEY         INT,
    PHONE_NUM              VARCHAR(12)                NOT NULL,
    NURI_MSG_SEQ           INT,
    USER_SEQ               INT,
    CUR_STATE              INT                        DEFAULT 0,
    APP_GUBUN              VARCHAR(8)                 NOT NULL,
    CALL_BACK              VARCHAR(12)                NOT NULL,
    SUB_ID                 VARCHAR(30),
    GAON_MSG_TYPE          CHAR(1)                   NOT NULL,
    SMS_MSG_DATA           VARCHAR(135),
    TEMPLATE_CODE          VARCHAR(30),
    RSLT_CODE_APP          CHAR(4),
    MODULE_SEND_TIME       CHAR(14),
    SVR_RECV_TIME          CHAR(14),
    APP_SEND_TIME          CHAR(14),
    APP_RECV_TIME          CHAR(14),
    SERIAL_NUMBER          VARCHAR(39),
    SEND_PACK_UKEY         INT,
    PRIMARY KEY (MSG_SEQ)
)
```


6. 이용기관시스템과 연동 DB Schema

CREATE TABLE IF NOT EXISTS APP_SEND_CONTENTS

```
(
    REQ_SEND_DATE          CHAR(14)
    PACK_UNIQUEKEY         INT(11)          NOT NULL,
    MSG_SUBJECT            VARCHAR(50),
    MSG_DATA               VARCHAR(3000)      NOT NULL,
    MSG_TYPE               CHAR(2)           NOT NULL,
    HEADER                 VARCHAR(32),
    BTN_CNT                INT(1)            DEFAULT 0,
    ATTACHMENT             VARCHAR(5000),
    LINK_CNT               INT(1)            DEFAULT 0,
    SUPPLEMENT             VARCHAR(5000),
    PHONE_CNT              INT(11)           DEFAULT 0 NOT NULL,
    CUR_STATE              INT(1)            DEFAULT 0,
    PRIMARY KEY (PACK_UNIQUEKEY)
)
```

CREATE TABLE IF NOT EXISTS TEMPLATE_CODE

```
(
    APP_GUBUN              VARCHAR(8)          NOT NULL,
    TEMPLATE_CODE          VARCHAR(30)         NOT NULL,
    HEAD                   VARCHAR(500)        NOT NULL,
    FOOT                   VARCHAR(500),
    TITLE                  VARCHAR(100),
    USE_BUTTON             CHAR(1)             DEFAULT 'N',
    PRIMARY KEY (APP_GUBUN, TEMPLATE_CODE)
)
```

6. 이용기관시스템과 연동 DB Schema

■ ORACLE(각 테이블별 컬럼의 설명은 31p 참조)

CREATE TABLE APP_SEND_DATA

```
(
    MSG_SEQ                NUMBER(11)                NOT NULL,
    REQ_SEND_DATE          CHAR(14)                    NOT NULL,
    PACK_UNIQUEKEY         NUMBER(11),
    PHONE_NUM              VARCHAR(12)                NOT NULL,
    NURI_MSG_SEQ           NUMBER(11),
    USER_SEQ               NUMBER(11),
    CUR_STATE              NUMBER(1)    DEFAULT 0,
    APP_GUBUN              VARCHAR(8)                  NOT NULL,
    CALL_BACK              VARCHAR(12)                NOT NULL,
    SUB_ID                 VARCHAR(30),
    GAON_MSG_TYPE          CHAR(1)                    NOT NULL,
    SMS_MSG_DATA           VARCHAR(135),
    TEMPLATE_CODE          VARCHAR(30),
    RSLT_CODE_APP          CHAR(4),
    MODULE_SEND_TIME       CHAR(14),
    SVR_RECV_TIME          CHAR(14),
    APP_SEND_TIME          CHAR(14),
    APP_RECV_TIME          CHAR(14),
    SERIAL_NUMBER          VARCHAR(39),
    SEND_PACK_UKEY         NUMBER(11),
    CONSTRAINT APP_SEND_DATA _PK PRIMARY KEY (MSG_SEQ) ENABLE
)
```

6. 이용기관시스템과 연동 DB Schema

```
CREATE TABLE APP_SEND_CONTENTS
```

```
(  
    REQ_SEND_DATE          CHAR(14)  
    PACK_UNIQUEKEY         NUMBER(11)          NOT NULL,  
    MSG_SUBJECT            VARCHAR(50),  
    MSG_DATA               VARCHAR(3000)        NOT NULL,  
    MSG_TYPE               CHAR(2)             NOT NULL,  
    HEADER                 VARCHAR(32),  
    BTN_CNT                NUMBER (1)  DEFAULT 0,  
    ATTACHMENT             VARCHAR(5000),  
    LINK_CNT               NUMBER (1)  DEFAULT 0,  
    SUPPLEMENT             VARCHAR(5000),  
    PHONE_CNT              NUMBER (11)  DEFAULT 0    NOT NULL,  
    CUR_STATE              NUMBER (1)  DEFAULT 0,  
    CONSTRAINT APP_SEND_CONTENTS_PK PRIMARY KEY (PACK_UNIQUEKEY) ENABLE  
)
```

```
CREATE TABLE TEMPLATE_CODE
```

```
(  
    APP_GUBUN              VARCHAR(8)          NOT NULL,  
    TEMPLATE_CODE          VARCHAR(30)         NOT NULL,  
    HEAD                   VARCHAR(500)        NOT NULL,  
    FOOT                   VARCHAR(500),  
    TITLE                  VARCHAR(100),  
    USE_BUTTON             CHAR(1)  DEFAULT 'N',  
    CONSTRAINT TEMPLATE_CODE_PK PRIMARY KEY (APP_GUBUN, TEMPLATE_CODE) ENABLE  
)
```

6. 이용기관시스템과 연동 DB Schema

■ MSSQL(각 테이블별 컬럼의 설명은 31p 참조)

```
CREATE TABLE APP_SEND_DATA
```

```
(
    MSG_SEQ                IDENTITY(1,1)                NOT NULL,
    REQ_SEND_DATE           CHAR(14)                     NOT NULL,
    PACK_UNIQUEKEY          INT,
    PHONE_NUM               VARCHAR(12)                  NOT NULL,
    NURI_MSG_SEQ            INT,
    USER_SEQ                INT,
    CUR_STATE               INT                          DEFAULT 0,
    APP_GUBUN               VARCHAR(8)                   NOT NULL,
    CALL_BACK               VARCHAR(12)                   NOT NULL,
    SUB_ID                  VARCHAR(30),
    GAON_MSG_TYPE           CHAR(1)                      NOT NULL,
    SMS_MSG_DATA            VARCHAR(135),
    TEMPLATE_CODE           VARCHAR(30),
    RSLT_CODE_APP           CHAR(4),
    MODULE_SEND_TIME        CHAR(14),
    SVR_RECV_TIME           CHAR(14),
    APP_SEND_TIME           CHAR(14),
    APP_RECV_TIME           CHAR(14),
    SERIAL_NUMBER           VARCHAR(39),
    SEND_PACK_UKEY          INT,
    PRIMARY KEY (MSG_SEQ)
)
```

6. 이용기관시스템과 연동 DB Schema

```
CREATE TABLE APP_SEND_CONTENTS
```

```
(  
    REQ_SEND_DATE          CHAR(14)  
    PACK_UNIQUEKEY          IDENTITY(1,1) NOT NULL,  
    MSG_SUBJECT             VARCHAR(50),  
    MSG_DATA                VARCHAR(3000) NOT NULL,  
    MSG_TYPE                CHAR(2)      NOT NULL,  
    HEADER                  VARCHAR(32),  
    BTN_CNT                 INT           DEFAULT 0,  
    ATTACHMENT              VARCHAR(5000),  
    LINK_CNT                INT           DEFAULT 0,  
    SUPPLEMENT              VARCHAR(5000),  
    PHONE_CNT               INT           DEFAULT 0 NOT NULL,  
    CUR_STATE               INT           DEFAULT 0,  
    PRIMARY KEY CLUSTERED (PACK_UNIQUEKEY)  
)
```

```
CREATE TABLE TEMPLATE_CODE
```

```
(  
    APP_GUBUN              VARCHAR(8)      NOT NULL,  
    TEMPLATE_CODE          VARCHAR(30)     NOT NULL,  
    HEAD                   VARCHAR(500)    NOT NULL,  
    FOOT                   VARCHAR(500),  
    TITLE                  VARCHAR(100),  
    USE_BUTTON             CHAR(1)         DEFAULT 'N',  
    PRIMARY KEY CLUSTERED (APP_GUBUN, TEMPLATE_CODE)  
)
```

6. 이용기관시스템과 연동 DB Schema

 Tibero(각 테이블별 컬럼의 설명은 31p 참조)

```
CREATE TABLE APP_SEND_DATA
(
    MSG_SEQ                NUMBER(11)                NOT NULL,
    REQ_SEND_DATE          CHAR(14)                   NOT NULL,
    PACK_UNIQUEKEY         NUMBER(11),
    PHONE_NUM              VARCHAR(12)                NOT NULL,
    NURI_MSG_SEQ           NUMBER(11),
    USER_SEQ              NUMBER(11),
    CUR_STATE              NUMBER(1)   DEFAULT 0,
    APP_GUBUN              VARCHAR(8)                 NOT NULL,
    CALL_BACK              VARCHAR(12)                NOT NULL,
    SUB_ID                 VARCHAR(30),
    GAON_MSG_TYPE          CHAR(1)                   NOT NULL,
    SMS_MSG_DATA            VARCHAR(135),
    TEMPLATE_CODE          VARCHAR(30),
    RSLT_CODE_APP          CHAR(4),
    MODULE_SEND_TIME       CHAR(14),
    SVR_RECV_TIME          CHAR(14),
    APP_SEND_TIME          CHAR(14),
    APP_RECV_TIME          CHAR(14),
    SERIAL_NUMBER          VARCHAR(39),
    SEND_PACK_UKEY         NUMBER(11),
    CONSTRAINT APP_SEND_DATA_PK PRIMARY KEY (MSG_SEQ)
)
```

6. 이용기관시스템과 연동 DB Schema

```
CREATE TABLE APP_SEND_CONTENTS
```

```
(  
    REQ_SEND_DATE          CHAR(14)  
    PACK_UNIQUEKEY         NUMBER(11)          NOT NULL,  
    MSG_SUBJECT            VARCHAR(50),  
    MSG_DATA               VARCHAR(3000)        NOT NULL,  
    MSG_TYPE               CHAR(2)             NOT NULL,  
    HEADER                 VARCHAR(32),  
    BTN_CNT                NUMBER (1)  DEFAULT 0,  
    ATTACHMENT             VARCHAR(5000),  
    LINK_CNT               NUMBER (1)  DEFAULT 0,  
    SUPPLEMENT            VARCHAR(5000),  
    PHONE_CNT              NUMBER (11)  DEFAULT 0  NOT NULL,  
    CUR_STATE              NUMBER (1)  DEFAULT 0,  
    CONSTRAINT APP_SEND_CONTENTS _PK PRIMARY KEY (PACK_UNIQUEKEY)  
)
```

```
CREATE TABLE TEMPLATE_CODE
```

```
(  
    APP_GUBUN              VARCHAR(8)          NOT NULL,  
    TEMPLATE_CODE          VARCHAR(30)         NOT NULL,  
    HEAD                   VARCHAR(500)        NOT NULL,  
    FOOT                   VARCHAR(500),  
    TITLE                  VARCHAR(100),  
    USE_BUTTON             CHAR(1)  DEFAULT 'N',  
    CONSTRAINT TEMPLATE_CODE _PK PRIMARY KEY (APP_GUBUN, TEMPLATE_CODE)  
)
```


6. 이용기관시스템과 연동 DB Schema

 Cubrid(각 테이블별 컬럼의 설명은 31p 참조)

```
CREATE TABLE APP_SEND_DATA
```

```
(
    MSG_SEQ                NUMERIC(11) AUTO_INCREMENT,
    REQ_SEND_DATE          CHAR(14)                NOT NULL,
    PACK_UNIQUEKEY         NUMERIC(11),
    PHONE_NUM              VARCHAR(12)              NOT NULL,
    NURI_MSG_SEQ           NUMERIC(11),
    USER_SEQ               NUMERIC(11),
    CUR_STATE              NUMERIC(1)  DEFAULT 0,
    APP_GUBUN              VARCHAR(8)                NOT NULL,
    CALL_BACK              VARCHAR(12)              NOT NULL,
    SUB_ID                 VARCHAR(30),
    GAON_MSG_TYPE          CHAR(1)                  NOT NULL,
    SMS_MSG_DATA           VARCHAR(135),
    TEMPLATE_CODE          VARCHAR(30),
    RSLT_CODE_APP          CHAR(4),
    MODULE_SEND_TIME       CHAR(14),
    SVR_RECV_TIME          CHAR(14),
    APP_SEND_TIME          CHAR(14),
    APP_RECV_TIME          CHAR(14),
    SERIAL_NUMBER          VARCHAR(39),
    SEND_PACK_UKEY         NUMERIC(11),
    PRIMARY KEY (MSG_SEQ)
)
```

6. 이용기관시스템과 연동 DB Schema

```
CREATE TABLE APP_SEND_CONTENTS
```

```
(  
    REQ_SEND_DATE          CHAR(14)  
    PACK_UNIQUEKEY         NUMERIC(11)  AUTO_INCREMENT      NOT NULL,  
    MSG_SUBJECT            VARCHAR(50),  
    MSG_DATA               VARCHAR(3000)      NOT NULL,  
    MSG_TYPE               CHAR(2)           NOT NULL,  
    HEADER                 VARCHAR(32),  
    BTN_CNT                NUMERIC(1)  DEFAULT 0,  
    ATTACHMENT             VARCHAR(5000),  
    LINK_CNT               NUMERIC(1)  DEFAULT 0,  
    SUPPLEMENT            VARCHAR(5000),  
    PHONE_CNT              NUMERIC(11)  DEFAULT 0      NOT NULL,  
    CUR_STATE              NUMERIC(1)  DEFAULT 0,  
    CONSTRAINT APP_SEND_CONTENTS _PK PRIMARY KEY (PACK_UNIQUEKEY)  
)
```

```
CREATE TABLE TEMPLATE_CODE
```

```
(  
    APP_GUBUN              VARCHAR(8)           NOT NULL,  
    TEMPLATE_CODE          VARCHAR(30)          NOT NULL,  
    HEAD                   VARCHAR(500)         NOT NULL,  
    FOOT                   VARCHAR(500),  
    TITLE                   VARCHAR(100),  
    USE_BUTTON             CHAR(1)      DEFAULT 'N',  
    CONSTRAINT TEMPLATE_CODE _PK PRIMARY KEY (APP_GUBUN, TEMPLATE_CODE)  
)
```

6. 이용기관시스템과 연동 DB Schema

■ POSTGRESQL(각 테이블별 컬럼의 설명은 31p 참조)

CREATE TABLE APP_SEND_DATA

```
(
    MSG_SEQ                SERIAL,
    REQ_SEND_DATE          CHAR(14)                NOT NULL,
    PACK_UNIQUEKEY        INT,
    PHONE_NUM              VARCHAR(12)              NOT NULL,
    NURI_MSG_SEQ           INT,
    USER_SEQ               INT,
    CUR_STATE              INT                DEFAULT 0,
    APP_GUBUN              VARCHAR(8)                NOT NULL,
    CALL_BACK              VARCHAR(12)              NOT NULL,
    SUB_ID                 VARCHAR(30),
    GAON_MSG_TYPE          CHAR(1)                NOT NULL,
    SMS_MSG_DATA           VARCHAR(135),
    TEMPLATE_CODE          VARCHAR(30),
    RSLT_CODE_APP          CHAR(4),
    MODULE_SEND_TIME       CHAR(14),
    SVR_RECV_TIME          CHAR(14),
    APP_SEND_TIME          CHAR(14),
    APP_RECV_TIME          CHAR(14),
    SERIAL_NUMBER          VARCHAR(39),
    SEND_PACK_UKEY         INT,
    PRIMARY KEY (MSG_SEQ)
)
```

6. 이용기관시스템과 연동 DB Schema

```
CREATE TABLE APP_SEND_CONTENTS
```

```
(
    REQ_SEND_DATE          CHAR(14)          ,
    PACK_UNIQUEKEY         SERIAL,
    MSG_SUBJECT            VARCHAR(50),
    MSG_DATA               VARCHAR(3000)      NOT NULL,
    MSG_TYPE              CHAR(2)            NOT NULL,
    HEADER                 VARCHAR(32),
    BTN_CNT                INT                DEFAULT 0,
    ATTACHMENT             VARCHAR(5000),
    LINK_CNT               INT                DEFAULT 0,
    SUPPLEMENT             VARCHAR(5000),
    PHONE_CNT              INT                DEFAULT 0 NOT NULL,
    CUR_STATE              INT                DEFAULT 0,
    PRIMARY KEY (PACK_UNIQUEKEY)
)
```

```
CREATE TABLE TEMPLATE_CODE
```

```
(
    APP_GUBUN              VARCHAR(8)          NOT NULL,
    TEMPLATE_CODE          VARCHAR(30)         NOT NULL,
    HEAD                   VARCHAR(500)        NOT NULL,
    FOOT                   VARCHAR(500),
    TITLE                  VARCHAR(100),
    USE_BUTTON             CHAR(1)             DEFAULT 'N',
    PRIMARY KEY (APP_GUBUN, TEMPLATE_CODE)
)
```

6. 이용기관시스템과 연동 DB Schema

■ ALTIBASE(각 테이블별 컬럼의 설명은 31p 참조)

CREATE TABLE APP_SEND_DATA

```
(
    MSG_SEQ                INT,
    REQ_SEND_DATE          CHAR(14)                NOT NULL,
    PACK_UNIQUEKEY        INT,
    PHONE_NUM              VARCHAR(12)                NOT NULL,
    NURI_MSG_SEQ           INT,
    USER_SEQ               INT,
    CUR_STATE              INT                DEFAULT 0,
    APP_GUBUN              VARCHAR(8)                NOT NULL,
    CALL_BACK              VARCHAR(12)                NOT NULL,
    SUB_ID                 VARCHAR(30),
    GAON_MSG_TYPE          CHAR(1)                NOT NULL,
    SMS_MSG_DATA           VARCHAR(135),
    TEMPLATE_CODE          VARCHAR(30),
    RSLT_CODE_APP          CHAR(4),
    MODULE_SEND_TIME       CHAR(14),
    SVR_RECV_TIME          CHAR(14),
    APP_SEND_TIME          CHAR(14),
    APP_RECV_TIME          CHAR(14),
    SERIAL_NUMBER          VARCHAR(39),
    SEND_PACK_UKEY         INT,
    PRIMARY KEY (MSG_SEQ)
)
```

6. 이용기관시스템과 연동 DB Schema

```
CREATE TABLE APP_SEND_CONTENTS
```

```
(
    REQ_SEND_DATE          CHAR(14)          ,
    PACK_UNIQUEKEY         INT,
    MSG_SUBJECT            VARCHAR(50),
    MSG_DATA               VARCHAR(3000)      NOT NULL,
    MSG_TYPE               CHAR(2)           NOT NULL,
    HEADER                 VARCHAR(32),
    BTN_CNT                INT                DEFAULT 0,
    ATTACHMENT             VARCHAR(5000),
    LINK_CNT               INT                DEFAULT 0,
    SUPPLEMENT             VARCHAR(5000),
    PHONE_CNT              INT                DEFAULT 0 NOT NULL,
    CUR_STATE              INT                DEFAULT 0,
    PRIMARY KEY (PACK_UNIQUEKEY)
)
```

```
CREATE TABLE TEMPLATE_CODE
```

```
(
    APP_GUBUN              VARCHAR(8)          NOT NULL,
    TEMPLATE_CODE          VARCHAR(30)         NOT NULL,
    HEAD                   VARCHAR(500)        NOT NULL,
    FOOT                   VARCHAR(500),
    TITLE                  VARCHAR(100),
    USE_BUTTON             CHAR(1)             DEFAULT 'N',
    PRIMARY KEY (APP_GUBUN, TEMPLATE_CODE)
)
```

6. 이용기관시스템과 연동 DB Schema

6.2.2. 테이블 필드 설명

■ APP_SEND_DATA

필드명	설명
MSG_SEQ(필수)	메시지 발송 일련 번호
REQ_SEND_DATE(필수)	이용기관이 전송하기 위해 DB에 입력한 시간 미래 시간을 입력하면 예약발송 가능
PACK_UNIQUEKEY	현 전송 건을 구분할 수 있는 구분 Key 만약 그룹으로 전송한 경우 그룹수신자들은 동일 APP_SEND_CONTENTS테이블의 PACK_UNIQUEKEY와 같은 PACK_UNIQUEKEY를 입력합니다.
USER_SEQ	유저가 임의로 사용 가능한 일련번호
PHONE_NUM(필수)	수신자 번호
NURI_MSG_SEQ	앱메시지 발송 실패 시, NURI모듈에 삽입된 MSG_SEQ의 일련번호
CUR_STATE(필수)	메시지 상태(32p 참조)
APP_GUBUN(필수)	카카오(KAKAO),네이버(NAVER)
CALL_BACK(필수)	발신자정보
SUB_ID	빌링 ID
GAON_MSG_TYPE(필수)	앱메시지 문자 타입 'S' : 단문, 'L' : 단문 or 장문 + 버튼 추가
SMS_MSG_DATA	GAON_MSG_TYPE이 'S'인 경우 문자내용을 이 컬럼에 입력
TEMPLATE_CODE(필수)	사전에 앱 플랫폼사로부터 심사 통과되어 사용 승인 받아둔 템플릿 코드 입력 사용하고자 하는 템플릿 코드가 있으면 입력. 없으면 기존 사용하는 문자 문구를 파악 후 템플릿 등록 요청을 국자원에 요청함

6. 이용기관시스템과 연동 DB Schema

6.2.2. 테이블 필드 설명

■ APP_SEND_DATA(계속)

필드명	설명
RSLT_CODE_APP	발송 응답 코드
MODULE_SEND_TIME	모듈에서 서버로 발송한 시간
SVR_RECV_TIME	서버로부터 레포트를 받은 시간
APP_SEND_TIME	서버에서 각 플랫폼으로 발송한 시간 (2.0.0.000001 부터 갱신 안됨)
APP_RECV_TIME	각 플랫폼에서 응답 받은 시간
SERIAL_NUMBER	시리얼 넘버
SEND_PACK_UKEY	서버에서 각 패킷을 구분하기 위한 일련번호시간 (2.0.0.000003 갱신 안됨)

6. 이용기관시스템과 연동 DB Schema

■ APP_SEND_CONTENTS

필드명	설명
REQ_SEND_DATE	이용기관이 전송하기 위해 DB에 입력한 시간
PACK_UNIQUEKEY(필수)	일련번호
MSG_SUBJECT	메시지 제목
MSG_DATA(필수)	메시지 내용
MSG_TYPE(필수)	메시지 타입(현재 이미지 알림톡은 지원 하지 않음) (AT: 알림톡, AI: 이미지 알림톡)
HEADER	이미지 알림톡에서 필요
BTN_CNT	버튼 수 (현재 최대 1개)
ATTACHMENT	버튼 세부 내용 (카카오, 토스, 네이버 공통) { "button": [{ "name": "비즈메시지 소개", "type": "WL", "url_pc": "http://bizmessage.kakao.com/", "url_mobile": "http://bizmessage.kakao.com/" }] }
LINK_CNT(사용불가)	바로링크 수 (사용 불가)
SUPPLEMENT(사용불가)	바로링크 세부내용 (사용불가) [{ "name": "비즈메시지 소개", "type": "WL", "url_pc": "http://bizmessage.kakao.com/", "url_mobile": "http://bizmessage.kakao.com/" }]
PHONE_CNT	(1.0.0.000006 버전 이후로 사용 안함) 연락처 개수
CUR_STATE	(1.0.0.000006 버전 이후로 사용 안함) 0: 들어온 상태 1: 발송완료 2: 포맷오류

6. 이용기관시스템과 연동 DB Schema

■ TEMPLATE_CODE

필드명	설명
APP_GUBUN(필수)	카카오(KAKAO), 네이버(NAVER)
TEMPLATE_CODE(필수)	템플릿 코드
HEADER(필수)	템플릿의 머릿말
FOOT	템플릿의 꼬릿말(사용하지 않음)
TITLE	템플릿의 타이틀(사용하지 않음)
USE_BUTTON	버튼 사용 여부 (신청한 템플릿에 버튼이 존재하면 Y, 없으면 N)

6. 이용기관시스템과 연동

6.2.3. 상태 코드 변화

■ APP_SEND_DATA

상태값	설명
0	메시지 등록
1	발송 요청 완료
2	응답완료(성공/실패는 RSLT_CODE_APP 컬럼 확인) / 발송 불가 (RSLT_CODE_APP 컬럼 확인, 문자 전환 기능 X)
3	문자 발송 / 발송 불가 (RSLT_CODE_APP 컬럼 확인, 문자 전환 기능 O)

6. 이용기관시스템과 연동

6.2.4. 결과코드 값

■ KAKAO

코드	설명
0000	정상코드
1001	Request Body가 Json형식이 아님
1002	허브 파트너 키가 유효하지 않음
1003	발신 프로필 키가 유효하지 않음
1004	Request Body(Json)에서 name을 찾을 수 없음
1006	삭제된 발신프로필. (메시지 사업 담당자에게 문의)
1007	차단 상태의 발신프로필. (메시지 사업 담당자에게 문의)
1011	계약정보를 찾을 수 없음. (메시지 사업 담당자에게 문의)
1012	잘못된 형식의 유저키 요청
1013	유효하지 않은 app연결
1014	유효하지 않은 사업자번호
1015	유효하지 않은 app user id 요청
1016	사업자등록번호 불일치
1012	잘못된 형식의 유저키 요청

6. 이용기관시스템과 연동

코드	설명
1020	전화번호 or app user id가 유효하지 않거나 미입력 요청
1021	차단 상태의 카카오톡 채널 (카카오톡 채널 운영툴에서 확인)
1022	닫힘 상태의 카카오톡 채널 (카카오톡 채널 운영툴에서 확인)
1023	삭제된 카카오톡 채널 (카카오톡 채널 운영툴에서 확인)
1024	삭제대기 상태의 카카오톡 채널 (카카오톡 채널 운영툴에서 확인)
1025	메시지차단 상태의 카카오톡 채널 (카카오톡 채널 운영툴에서확인)
1026	해당 message_type에서 사용할 수 없는 response_method로요청 (이미지알림톡(AI)는 realtime으로 발송 불가)
1030	잘못된 파라미터 요청
2003	(테스트 발송) 카카오톡 채널을 추가하지 않았음
2004	템플릿 일치 확인시 오류 발생(내부 오류 발생)
2006	<4.2 메시지 전송 요청>에 명시된 시리얼번호 형식 불일치
3000	예기치 않은 오류 발생
3005	메시지를 발송했으나 수신확인 안됨 (성공불확실)- 서버에는 암호화되어 보관되며 3일 이내 수신 가능
3006	내부 시스템 오류로 메시지 전송 실패
3008	전화번호 오류

6. 이용기관시스템과 연동

코드	설명
3010	Json 파싱 오류
3011	메시지가 존재하지 않음
3012	메시지 일련번호가 중복됨 == 메시지 일련번호는 CS처리를 위해 고유한 값이 부여되어야함.
3013	메시지가 비어 있음
3014	메시지 길이 제한 오류 (템플릿별 제한 길이 또는 1000자 초과)
3015	템플릿을 찾을 수 없음
3016	메시지 내용이 템플릿과 일치하지 않음
3018	메시지를 전송할 수 없음
3025	변수 글자수 제한 초과
3026	상담/봇 전환 버튼 extra, event 글자수 제한 초과
3027	메시지 버튼/바로연결이 템플릿과 일치하지 않음
3028	메시지 강조 표기 타이틀이 템플릿과 일치하지 않음
3029	메시지 강조 표기 타이틀 길이 제한 초과 (50자)
3030	메시지 타입과 템플릿 강조유형이 일치하지 않음
3031	헤더가 템플릿과 일치하지 않음

6. 이용기관시스템과 연동

코드	설명
3032	헤더 길이 제한 초과(16자)
3033	아이템 하이라이트가 템플릿과 일치하지 않음
3034	아이템 하이라이트 타이틀 길이 제한 초과(이미지 없는 경우 30자, 이미지 있는 경우 21자)
3035	아이템 하이라이트 디스크립션 길이 제한 초과(이미지 없는 경우 19자, 이미지 있는 경우 14자)
3036	아이템 리스트가 템플릿과 일치하지 않음
3037	아이템 리스트의 아이템의 디스크립션 길이 제한 초과(23자)
3038	아이템 요약정보가 템플릿과 일치하지 않음
3039	아이템 요약정보의 디스크립션 길이 제한 초과(14자)
3040	아이템 요약정보의 디스크립션에 허용되지 않은 문자 포함(통화기호/코드, 숫자, 콤마, 소수점, 공백을 제외한 문자 포함)
4000	메시지 전송 결과를 찾을 수 없음
4001	알 수 없는 메시지 상태
5000	(테스트 발송) 관리자 혹은 일회성 인증을 받은 사용자가 아님
5001	(테스트 발송) 일일 발송량 초과
9991	발송실패(발송정보 알 수 없음)
9998	시스템에 문제가 발생하여 담당자가 확인하고 있는 경우
9999	담당자가 확인 중입니다. 시스템에 문제가 발생하여 담당자가 확인하고 있는 경우

6. 이용기관시스템과 연동

■ NAVER

코드	설명
0000	메세지가 유저에게 앱과 웹으로 성공적으로 전송되었습니다.
0001	메세지가 유저에게 웹으로 전송되었으나 앱 전송에는 실패하였습니다.
1001	AgentKey나 PartnerKey가 누락되었습니다.
1002	보내기 API의 요청 Body에 문제가 있습니다.
1003	전화번호나 액세스 토큰 중 적어도 하나의 파라미터가 필요합니다.
1004	전화번호 패턴이 잘못되었습니다.
1005	파트너키나 발송그룹키가 반드시 있어야 합니다.
2001	템플릿에 필요한 파라미터가 부족합니다.
2002	템플릿 아이디가 존재하지 않습니다.
2003	템플릿에 필요한 파라미터가 일치하지 않습니다.
2004	템플릿에 필요한 버튼 갯수와 요청한 버튼 갯수가 일치하지 않습니다.
2005	템플릿에 필요한 파라미터가 일치하지 않습니다. 버튼코드를 확인하세요.
2006	등록된 템플릿과 일치하지 않는 메세지(message) 파라미터입니다.
2007	승인되지 않은 템플릿은 발송할 수 없습니다. 템플릿 승인 상태를 확인하세요.
2008	등록된 템플릿과 메세지 파라미터를 비교하던 중 오류가 발생하였습니다. 2006 응답코드와 거의 동일한 상황이나, 특수문자에 의해 불일치가 있는 경우입니다.
2009	관리자에 의해 차단된 템플릿입니다. 자세한 내용은 문의해주세요.

6. 이용기관시스템과 연동

코드	설명
2010	템플릿에 사용하는 샘플 이미지가 등록되지 않았습니다. 샘플 이미지를 등록해주세요.
2011	이미지 해시 아이디에 해당하는 파트너키나 템플릿 그룹키를 사용해야 합니다.
2101	템플릿에 저장된 템플릿그룹 아이디와 일치하지 않는 템플릿 요청입니다. 템플릿 설정을 확인해주세요.
2102	유효하지 않은 템플릿 그룹키 입니다. 그룹키를 확인해주세요.
2103	템플릿 그룹의 파트너 목록에 존재하지 않는 요청입니다.
2104	이미지 해시 아이디에 해당하는 템플릿 그룹키를 확인해주세요.
3001	에이전트 키에 해당하는 파트너가 매핑되어 있지 않습니다.
3002	전화번호에 해당하는 유저가 존재하지 않습니다.
3003	매핑된 파트너가 존재하지 않습니다.
3004	이미 존재하는 메세지 키입니다.
3005	전화번호에 해당하는 서로 다른 유저가 존재합니다.
4001	유저가 정보성 알림을 받을 수 있는 상태가 아닙니다.
4002	유저가 스마트알림 수신을 차단하였습니다.
4003	유저가 파트너 계정을 차단하였습니다.
9051	메세지 시스템의 오류로 정보성 알림 송신이 불가능합니다. 주로 특특메세지를 보내다가 발생하는 오류이며, 자세한 오류코드는 점차 업데이트하겠습니다.
9999	알 수 없는 이유로 메세지가 전송되지 않았습니다.

6. 이용기관시스템과 연동

GAON

코드	설명
G100	APP_GUBUN에 KAKAO, NAVER 이외의 값이 입력됨(대문자만 입력가능)
G101	전화번호 에러
G200	TEMPLATE_CODE 테이블에 템플릿이 존재하지 않음
G201	버튼의 JSON형식이 맞지 않음
G202	버튼이 존재하나 url이 http:// 혹은 https://로 시작하지 않음
G300	메시지 내용이 빈값
G301	GAON_MSG_TYPE이 S인 경우 내용이 90Byte 초과
G400	각 번호별 일일 발송량 초과
G777	TEST BED
G900	서버연결 실패
G901	인증 실패
G902	타임 아웃
G903	메시지 내용이 제한 길이를 초과

6. 이용기관시스템과 연동

6.2.5. 앱 기반 메시지 보내는 방법

■ 공통

KAKAO와 NAVER로 앱메시지를 보내기 위해서는 TEMPLATE_CODE 테이블에 등록하신 템플릿 코드와 머리말을 입력해야 합니다.

■ 템플릿 등록

- APP_GUBUN에는 템플릿을 신청한 플랫폼 명을 입력(KAKAO 또는 NAVER)
- TEMPLATE_CODE에는 신청한 템플릿의 템플릿코드를 입력
- HEAD 신청한 템플릿의 머리말을 입력
- 결과를 엑셀파일로 받으신 경우 템플릿 내용부분에서 #{변수명}을 제외한 부분을 입력
- FOOT, TITLE 컬럼은 사용하지 않음. 입력X
- USE_BUTTON 컬럼은 신청한 템플릿에 버튼이 존재하는 경우 Y, 없으면 N

샘플SQL)

INSERT INTO

TEMPLATE_CODE (APP_GUBUN, TEMPLATE_CODE, HEAD, USE_BUTTON)

VALUES(' KAKAO ' , ' TEST_001 ' , ' TEST HEAD ' , ' N ');

6. 이용기관시스템과 연동

6.2.5. 앱 기반 메시지 보내는 방법

■ 단문

단문(버튼 이용 불가)만 전송하기 위한 용도로 APP_SEND_DATA 테이블에 필요한 데이터를 넣어주면 된다.

- MSG_SEQ는 자동 증가 시퀀스값을 넣어준다.
- REQ_SEND_DATA에는 발송 날짜를 입력한다. 미래 시간을 입력하는 경우 해당 시간에 발송이 된다. (yyyyMMddHHmmss형식)
- PHONE_NUM에는 수신자의 연락처를 입력해준다.
- CUR_STATE는 0으로 고정된 값을 넣으면 된다.
- APP_GUBUN에는 KAKAO, NAVER중 원하는 플랫폼을 넣어주면 된다.
- CALL_BACK에는 발신자 연락처를 입력해준다.
- GAON_MSG_TYPE에는 'S'를 넣어준다.
- SMS_MSG_DATA에는 보낼 내용을 입력해 준다.
- TEMPLATE_CODE에는 KAKAO, NAVER의 템플릿 코드를 입력해 준다.
- **머리말은 TEMPLATE_CODE테이블에 등록해둔 머리말을 자동으로 조합해서 보내기 때문에 SMS_MSG_DATA에 머리말을 입력할 필요가 없습니다.**

6. 이용기관시스템과 연동

샘플 SQL

```
INSERT INTO APP_SEND_DATA(  
MSG_SEQ, REQ_SEND_DATE, PHONE_NUM, CUR_STATE, APP_GUBUN, CALL_BACK,  
GAON_MSG_TYPE, SMS_MSG_DATA, TEMPLATE_CODE)  
VALUES (1, '20211012044300', '01012341234', 0, 'KAKAO', '12341234', 'S', '테스트  
메시지 입니다', 'templatecode_0001')
```

6. 이용기관시스템과 연동

■ 장문 or 단문 + 버튼

단문 or 장문에 필요 시 버튼을 추가하여 보낼 수 있다.

APP_SEND_DATA의 경우

- MSG_SEQ에는 자동증가 시퀀스를 입력해준다.
- REQ_SEND_DATA에는 발송 날짜를 입력한다. 미래 시간을 입력하는 경우 해당 시간에 발송이 된다. (yyyyMMddHHmmss형식)
- PACK_UNIQUEKEY에는 APP_SEND_CONTENTS에 입력한 PACK_UNIQUEKEY와 같은 값을 입력해준다.
- PHONE_NUM에는 수신자 연락처를 입력해준다.
- CUR_STATE에는 0을 고정적으로 넣어주면 된다.
- APP_GUBUN에는 KAKAO, NAVER중 발송을 원하는 플랫폼을 선택하여 입력해주면 된다.
- CALL_BACK에는 발신자 연락처를 입력해 준다.
- GAON_MSG_TYPE에는 'L'을 입력해준다.
- TEMPLATE_CODE에는 KAKAO, NAVER의 템플릿 코드를 입력해 준다.

6. 이용기관시스템과 연동

장문 or 단문 + 버튼(계속)

APP_SEND_CONTENTS의 경우.

- PACK_UNIQUEKEY는 자동증가 시퀀스이다.
- CUR_STATE에는 0을 고정적으로 입력해주면 된다.
- MSG_DATA에는 템플릿의 머리말과 꼬릿말을 제외한 내용을 입력해주면 된다
- MSG_TYPE에는 AT를 고정적으로 넣어주면 된다.
- 머리말은 TEMPLATE_CODE테이블에 등록해둔 머리말을 자동으로 조합해서 보내기 때문에 MSG_DATA에 머리말을 입력할 필요가 없습니다.
- 버튼을 이용 시 JSON 형식으로 ATTACHMENT 컬럼에 입력해 주시면 됩니다.

ex)

```
{
  "button": [
    {
      "name": "비즈메시지 소개",
      "type": "WL",
      "url_pc": "http://bizmessage.kakao.com/",
      "url_mobile": "http://bizmessage.kakao.com/"
    }
  ]
}
```

- Name에는 버튼명을 입력하면 된다.
- Type에는 WL을 고정적으로 입력하면 된다.
- url_pc, url_mobile 둘 다 입력해야 하며, KAKAO는 300자, NAVER는 256자로 제한됩니다.
- 현재 버튼은 한 개만 지원하고 있습니다.
- 2.0.9 버전 이전 - 버튼의 URL은 'http://' 로 시작해야 합니다
- 2.0.10버전 이후 - 'https://'로 입력된 경우 'http://'로 변환되고, 버튼의 URL에는 프로토콜(http://)을 입력하지 않아도 자동으로 http://를 추가해줍니다.

6. 이용기관시스템과 연동

장문 or 단문 + 버튼

샘플 SQL)

```
INSERT INTO APP_SEND_CONTENTS(REQ_SEND_DATE, PACK_UNIQUEKEY,  
MSG_DATA, MSG_TYPE, PHONE_CNT, CUR_STATE) VALUES ('20210908120000', 1,,  
'테스트 메시지 입니다', 'AT', 1, 0);
```

```
INSERT INTO APP_SEND_DATA(REQ_SEND_DATE, PACK_UNIQUEKEY, PHONE_NUM,  
CUR_STATE, APP_GUBUN, CALL_BACK, TEMPLATE_CODE, GAON_MSG_TYPE)  
VALUES ('20210908120000', 1, '01012345678', 0, 'KAKAO', '12345678',  
'testtemplate001', 'L');
```

버튼 이용 샘플 SQL)

```
INSERT INTO APP_SEND_CONTENTS(REQ_SEND_DATE, PACK_UNIQUEKEY,  
MSG_DATA, MSG_TYPE, PHONE_CNT, CUR_STATE, BTN_CNT, ATTACHMENT)  
VALUES ('20210908120000', 1, '테스트 메시지 입니다', 'AT', 1, 0, 1,  
'{"button":[{"name":"비즈메시지  
소개","type":"WL","url_pc":"http://bizmessage.kakao.com/",  
"url_mobile":"http://bizmessage.kakao.com/"}]}');
```

```
INSERT INTO APP_SEND_DATA(REQ_SEND_DATE, PACK_UNIQUEKEY, PHONE_NUM,  
CUR_STATE, APP_GUBUN, CALL_BACK, TEMPLATE_CODE, GAON_MSG_TYPE)  
VALUES ('20210908120000', 1, '01012345678', 0, 'KAKAO', '12345678',  
'testtemplate001', 'L');
```

6. 이용기관시스템과 연동

6.2.6. 결과 수신 방법

- APP_SEND_DATA의 (RSLT_CODE_APP, SVR_RECV_TIME, APP_RECV_TIME, SERIAL_NUMBER) 컬럼의 값이 서버로 부터 응답을 받으면 갱신됨.

- RSLT_CODE_APP : 앱 메시지 전송 결과 코드
- SVR_RECV_TIME : 서버에서 메시지를 받은 시간
- APP_RECV_TIME : 서버에서 각 플랫폼으로부터 응답을 받은 시간

6.2.7. 실패 시 SMS/MMS 자동 전환 전송 방법

- SMS/MMS 로 자동 전환 전송을 이용할 경우, NURI 모듈이 필요함.
- NURI 모듈을 사용하기 위해 가온 모듈 환경설정에서 dbName, dbDriveName, dbUrl, dbUser, dbPassword, msgDataSeq, msgContentsInfoSeq, msgDataTableName, msgContentsInfoTableName의 값을 NURI모듈에서 사용하는 설정값으로 설정해주어야 함
- APP_SEND_DATA의 NURI_MSG_SEQ 컬럼은 SMS/MMS 자동 전환 전송이 이루어지면 NURI의 MSG_DATA테이블에 데이터를 넣은 후 MSG_DATA테이블의 MSG_SEQ의 값을 업데이트 해줌

6. 이용기관시스템과 연동

6.2.8. 로그테이블

- APP_SEND_DATA에 있는 데이터를 로그테이블로 이관할 때 CUR_STATE, APP_RESULT_CODE의 값과 setting.yaml의 useLms값에 따라 테이블로 이관할지 말지를 결정한다.

CUR_STATE	APP_RESULT_CODE	설명
2	0000	이관
2	0000 이외	문자 발송 기능 사용시 : 문자 발송 후 이관 문자 발송 사용 하지 않는 경우 : 이관
3	G902	타임아웃으로 beforeTime 옵션의 값만큼 보관 후 이관
3	G902 이외	이관
ALL	ALL	메시지 등록일이 현재시간에서 beforeTime 옵션의 시간을 뺀 시간보다 이전일 경우 이관



모바일메시지서비스 이용문의

- 모바일메시지서비스 웹사이트 : <http://www.mgov.go.kr>
- 국가정보자원관리원 운영총괄과 모바일메시지서비스 담당 : 042-250-5333
- 국가정보자원관리원 모바일메시지서비스 운영·기술지원 : 042-250-5537