**OpenComm : v 1.0**

**Created By : Amol Shelke**

**Created On : 29/06/2015**

**INTRODUCTION :**

In all type of application we need to communicate to client through sms , email , two way messaging , IVR etc.. Purpose of this api application is to developed central communication platform for all type of applications. This will behave like third party application. All other applications (It might be php , .Net or Java) can use this platform for communicate.

**VERSION 1.0 :**

1.0 version will be only for sms related api. Later on we will integrate email, 2 way communication and ivr api's.

Usecases

Flow diagram

**DATABASE STRUCTURE**

1. sms\_config

* + ID(Int)
  + ORG\_CODE (varchar)
  + URL (varchar)
  + UID(varchar)
  + PIN(varchar)

2. sms\_inbox

* ID(Int)
* ORG\_CODE(varchar)
* ROUTE (Int)
* MOBILE(varchar)
* RAW\_MSG\_ID(int)FK
* MSG\_ID (varchar) coming from api
* MSG\_PUSH\_ID (varchar) coming from api.
* MSG\_STATUS enum (SENT/DELIVERED/DND/FAILED/PROCESSNG)
* SENT\_ON (datetime)
* DELEVIRED\_ON(datetime)
* DEL\_FLG (char) default N

3. raw\_messages (Its bz when we pass sms to multiple users , the same data will be duplicated)

* ID
* MSG (varchar)

API's

* sendSms
  + i/p :

1. mobile no (multiple comma seperated)

2. message

3. route

4. org\_code

* + Description :

We have to develope this api in such a way that

* + - * Syncronious : When user send sms , system will send sms , store it into the database and the send status to customer. We have to store the message\_id / push\_id into database. Also backend process will update message status continiously.
      * Asyncronious : When user send sms , system will imidiate send status to customer, and backend process will update the databse and backend process continiously update sms\_status
* getDeliveryReport
  + - * i/p :
        + ORG\_CODE
        + MSG\_ID/PUSH\_ID (Comma seperated in multiple case)
* Response : Design soon (It will be json object )
* getDeliveryReportBetDate
  + - * i/p : (We can go for json object )
        + ORG\_CODE
        + MSG\_ID/PUSH\_ID (Comma seperated in multiple case)
        + START\_DATE, END\_DATE
  + Response : Design soon (It will be json object )
* getBalance :
  + - * i/p :
        + ORG\_CODE
        + ROUTE
* sendScheduledSms : this will be same as send sms only difference in we have one extra parameter datetime, the date time should be between 9:00 AP to 9:00 PM (We need to vaalidate this things )

Note : As per try rule we will send bulk sms only between 9:00 AM to 9:00 PM in our system we need to validate above information