



Business Situation:

Our client wanted to stay competitive in the Text Messaging Services Industry that has a significant share in the Marketing and Advertising campaigns in USA and elsewhere in the world. However, when they approached Mindfire Solutions their Code was not in the best condition and they had to restart the servers on a daily basis and there were times when the messaging broadcast would fail randomly.

In this scenario, Mindfire had its task cut and proceeded with a modular approach giving more weightage to the more pressing issues first.

The Mindfire Solution:

The primary customers of our client were Advertisers who can get distributed Shortcode or they can own a Shortcode. The Advertiser can set up campaigns for Restaurants, Comedy Clubs, College, Real Estate etc. They can also set Zip Code location processing for a scheduled Broadcast. The main purpose of the Campaigns was to give real time notifications to a user about an offer going on, Coupons & information about the Advertiser's shop location primarily through Broadcasts or through Response Messages. The Advertisers also get access to the subscribers' Name, DOB (date of birth), email, Zip etc. which helps them to provide the best services.

At Mindfire Solutions, we analyzed the code base which had flaws and did code optimization in every corner of the application. Our developers wrote approximately 75 stored procedures which had a capacity to give 5 to 7 resultsets. We used Bootstrap, ChartJS for Admin dashboard without dependencies on the Base CSS. Most of the queries were replaced by the stored procedures so that Database hit was managed. The performance of the application improved considerably without having the client to restart or suffer from a system hanging up while broadcasting.

We faced a challenge for new API integration without support from Service providers. Our developers worked on a new API for the client which in turn enabled the Advertiser to send test to N numbers, N different messages in one to one relation. This API was set up using CF Scheduled services and Caching query result. The Stats, Growth, Summary Reports, Messages Logs and Advertiser Dashboard were set up so Advertisers and Our Client have a clear picture on Application and Campaigns. Surprise Redemption Code campaign was set up with having 2 way communications with Advertiser's application, which went on for several months with success.

Fetching DOB data collection from Subscriber for giving special offers on birthdays was set up, by using a list of Regular Expressions to accept any format. The data was achieved with 80% success. A robust structure was established for setting up the campaigns so that the Advertiser can limit the number of outgoing messages, collect the data and send emails in real time. New Mail Chimp, Custom JSON API was set for sending real time data to Advertisers. Our developers also converted HTTP to SMPP protocol because the providers had a lot to change on their side. The application stopped after successful transfer of the Shortcodes. We also applied every New CTIA changes for sending of Messages to Subscribers.

The application had a unique pass-thru facility to the Advertisers which notifies them as soon as they get Subscriptions or any information for Campaigns. The various tracking reports were installed to get the Subscribers List, Growth of Subscribers, Stats given by subscriber and Summary Report for getting



response messages that went out from Application for Subscriber messages. The application also had few API Listeners open so that the Advertiser can send messages without logging in or can manage broadcast at their end. Base camp was used for Bug tracking and junit tool was used by client end for testing.

Architecture Diagram:

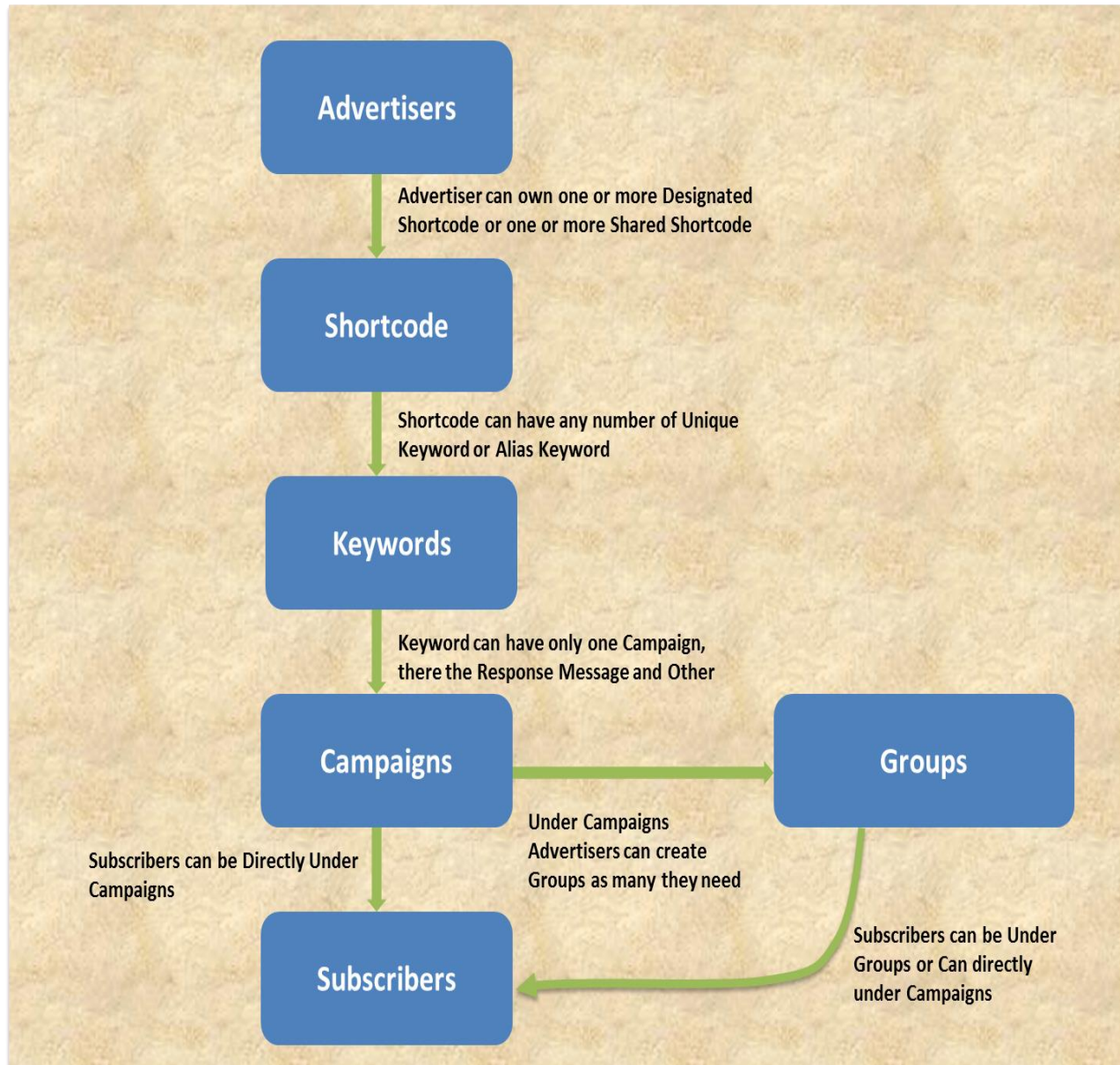


Fig 1: The above figure shows the architecture diagram of the Text Messaging Solution



How the System works?

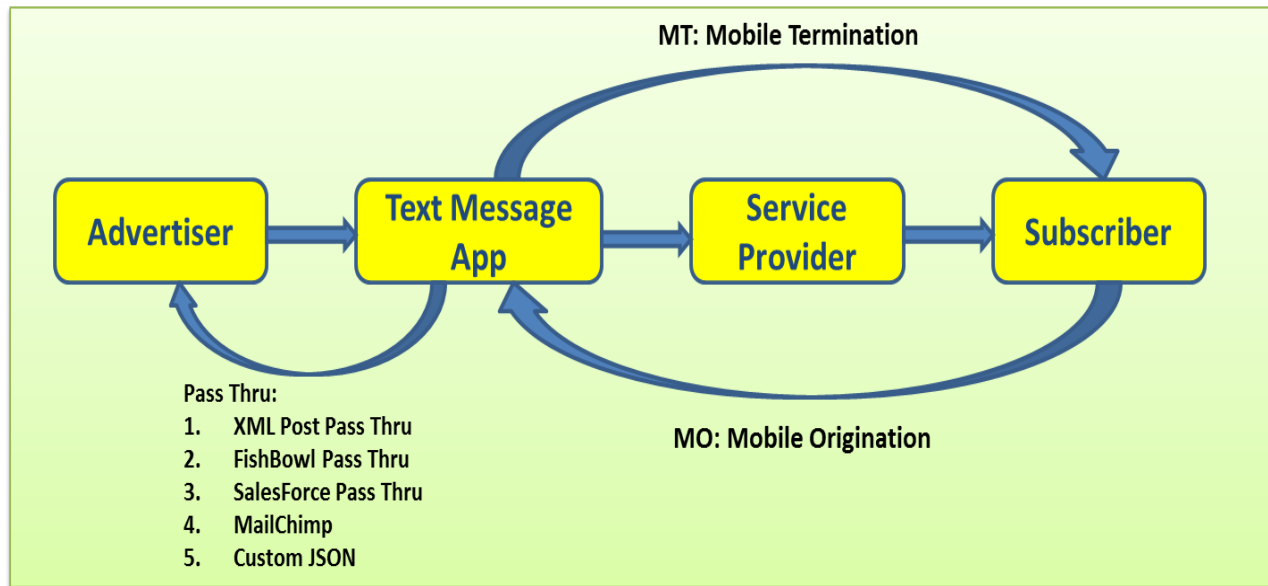


Fig 2: The above figure shows the WorkFlow image of the Text Messaging Solution

There were 3 processes which worked behind the application:

1. **MO(Mobile Origination)**: Here the Messages is received by Subscribers. The message is processed and marked valid or invalid
2. **MT(Mobile Termination)**: Here the Message Response from Application to Subscribers and Broadcast set by Advertisers were handled.
3. **Pass Thru**: There were many kinds of Pass Thru in the Application. These are used to indicate the Advertisers in Real time about data received.

Screenshots:

Following are some of the screenshots of various sections of the application:

Screenshot 1: User Dashboard
signifying the various metrics of the subscriber base

