**Technical Design Specifications**

**for**

1. **Form Submissions - Connector - NextGen**
2. **Link Action Notifications - NextGen**

**Form Submissions - Connector - NextGen**

**Purpose:** To develop a service to create / update new contact in the NextGen Account – Based on form submissions from classic – execlusively the forms hosted at various location for each account using the classic form builder module.

**Pre-Requirements:** [To be provided by NextGen Dev team]

1. NextGen Contacts Web API Service URL. -
2. Mapping Table for - Forms for forms identification in NextGen.
3. Mapping Table for – All account level Custom Fields

**Note:** All the form fields data ( Other than Standard and Custom Fields) in classic will be store as form sumission HTML in NextGen under form submissions table and will not be stores as field data in contacts. ( The same approach that we have been maintaining in classic).

**Requirements:**

There should be a hosted realtime service that should be called on each sucessful form submission to create a contact on the Nextgen – via the referanced form source for the account.

Classic needs to be extended with a new column to have status on contact submission to NextGen and Response associated to the submission.

There will be additional schedular service that will re-try the failed real time contact submissions to Nextgen for 10 Iterations on Hourly basis or Untill Gets Created in NextGen Successful which ever is earlier.

There will be a report in the classic at each account level that will show all the pending contacts to be submitted to NextGen at account level to validate.

**Note:** The above requirements does not handle any failures where Classic form submissions services are down through which contacts dircetly submit their submissions .

**Classic – Dev - 40 Hrs, QA – 10 Hrs**

**Nextgen – Dev - , QA –**

**Link Action Notifications - NextGen**

**Purpose:** To develop a service to Invoke nextGen Automation workflow to perform an action on link click of a link sent via classic thorugh campaigns. This is requires to handle the MAP defined in classic to continue in nextGen.

This will also handle the use cases of

1.Double Opt-in

2. Un-Subscribe

**Pre-Requirements:** [To be provided by NextGen Dev team]

1. NextGen Link Notification Web API Service URL.
2. Define Double Opt-in and Un-Subscribe link ID’s and create required workflows with appropriate actions on link clicks actions in NextGen.

**Requirements :**

When ever a click happens in classic – A realtime service request will be sent with the classic Contact ID, Link ID , Campaign ID to nextGen

Based on the Link ID & Campaign ID, Contact ID – Sent by Classic – nextgen will Identify the workflow’s and actions to be performed that are associated to link to perform the action.

And send back the status on the notification acknowledgement.

Any time the link notification acknowledgement is failed – the Classic will retry untill 10 time or for the successful acknowledgement which ever is earlier.

**Classic – Dev - 12 Hrs, QA – 5 Hrs**

**Nextgen – Dev - , QA –**