# Send out SMS Notifications

Version: 1.1

**Change History**

| Version | Changes Summary | Author |
| --- | --- | --- |
| 0.1 | Initial set of Detailed requirements | Paul Cleary |
| 0.2 | Insert of test SMS messages, added Audit Trail requirement, process reporting now to be realtime in the logfile | Paul Cleary |
| 1.0 | Moved backup of CSV file to be in the CSV File creation process. Modified trigger so pickup of the CSV file is now automatic, with the option to do manually for test purposes. Created final of this chapter | Paul Cleary |
| 1.1 | Added requirement that each SMS Notification message has a unique message id and this is recorded in the Campaign Entry table | Paul Cleary |

# Overview

This chapter contains the detailed requirements for the process of taking the Notifications CSV file and creating & sending out SMS notifications to “opted-in” myLotto users.

This process is very technically complex (refer to the P2P design document “Push2Play Detailed Design”) so the format for this requirements chapter is to treat the internals of the process as a “black box” and just describe the success criteria (=desired outcomes).

The Push 2 Play (P2P) database tables and their columns are described in the P2P design document “Push2Play Detailed Design” – the latest version can be obtained from Ferdinand Contreras at Lotto NZ.

# Context Diagram

The following diagram is a high level view of all the Push 2 Play processes that take place on a Lotto Game Draw day. The processes outlined in red are covered in this chapter.



# Process Success Criteria

**Preconditions:**

The “BI Create Notifications CSV file” process has completed successfully and generated a CSV file which is in a known location

**Trigger:** At a specified start time on an Operations work schedule, an automatic FTP process uploads the Notification CSV file to a location where it will be picked up by the MULE system and input to this process.

Note: the Lotto NZ IT Operations team can take a copy of this CSV file and adapt it for testing purposes. They can then manually move their adapted CSV file to the same location so it is input to this process for test purposes

**Process Area: File is Uploaded and filtered**

The data rows in the CSV file pass a preliminary simple CSV data integrity check:

* Data columns are appropriately blank or are populated
* Data values are correctly formatted (Chars, numbers or dates) – MULE would automatically do this formatting check

More details of these filter rules are contained in the Push2Play Jira project user story & subtasks for this process

**Process Area: Build an SMS message from each row in the CSV file**

The process correctly builds an SMS message:

* The standard SMS message template text is not changed or damaged by the build
* Data values from a CSV file row are inserted in the correct locations in the message template
* Data values are not changed (except where required to downsize a message instance) or damaged by the build
* Each message will be assigned a unique message id

**Process Area: Downsize initial SMS messages greater than 160 characters**

If total length of initial SMS message =< 160 characters then do nothing

If total length of initial SMS message > 160 characters then the following decision table applies:

| When… | Then action is … |
| --- | --- |
| Notification is for a dip | Reduce Player name to 1st 12 characters + \*\*\* |
| Notification is for a favourite AND reducing the Favourite name to 15 characters results in the draft being =< 160 characters | Reduce Favourite name to 1st 12 characters + \*\*\* |
| Notification is for a favourite AND reducing the Favourite name to 15 characters still results in the draft being > 160 characters | Reduce Favourite name to 1st 12 characters + \*\*\*  Reduce Player name to 1st 12 characters + \*\*\* |

**Process Area: Pre-check on transmitted SMS message**

The Lotto NZ IT Operations team member is able to insert SMS test messages, based on a sample taken from the Notifications CSV file, so they can verify the SMS message content is being correctly formatted.

They require that these messages are sent out as part of the first batch of SMS messages.

**Process Area: Control of the process run**

The Lotto NZ IT Operations team member is able to:

* Specify changes to the purchase Cutoff time as contained in the SMS message for a draw, e.g. move it forward from 7:30PM to 7:15PM
* Stop the process run
* Mark the process run as Completed
* They cannot restart a process run once it has been stopped  
  (Resume facility requirement is logged in the P2P project decision register)

**Process Area: Audit trail of SMS messages actually sent out**

An entry in the table campaign\_entry must contain the following information:

* A unique id for the Notification message
* Date & time when the original Notification message is successfully dispatched to the SMS transmission system (aka Burst)
* Date & time when a message response is received from the SMS transmission system

So if IT Operations needed to stop the process prematurely, there is a record of what users have already had a Notifications SMS message sent out to them.

**Process Area: Realtime Operational reporting during each run of the process**

The process generates an operational report in realtime for each run. The report data is to be written to a log file.

The following data is to be recorded in the logfile:

* Start time of process run with total number of intended SMS messages, and any control information (e.g. draw number) and settings (if any)
* Cumulative number of input notification messages processed (refreshed every 3 minutes)
* Cumulative number of output messages sent to the SMS transmission system (refreshed every 3 minutes)
* Cumulative number of output messages accepted by the SMS transmission system (refreshed every 3 minutes)
* Documentation of each occurrance of a failure to successfully send an SMS message, with mobile phone number and error code
* End time of process run with total number of SMS messages actually processed