
txtAlert Documentation

Release 0.1

Praekelt Foundation

August 14, 2015

1	txtAlert Operations	3
1.1	Sending of Reminders	3
1.2	MRS Integration	3
1.3	Clinic Admin	3
1.4	CD4 Admin	3
1.5	Mobi Bookings	4
1.6	GDocs Integration	4
2	Documentation for txtAlert API	5
2.1	Please Call Me API documentation	5
2.2	Registering PCMs	5
2.3	Sending SMSs	6
2.4	Receiving Delivery Reports	6
3	Indices and tables	7

txtAlert is a mobile technology tool developed by Praekelt Foundation. It uses text message (SMS) reminders to improve the health of HIV positive people, by reminding them when they need to go for their next visit to the clinic.

Visit <http://praekeltfoundation.org/txtalert.html> for more information.

Note: This documentation is generated automatically from the GPL source code available at <https://github.com/praekelt/txtalert>.

Contents:

txtAlert Operations

1.1 Sending of Reminders

A cron job runs every day at noon, this does the following:

1. It imports all data from the MRS.
2. It updates all appointment records based on the MRS import.
3. It sends out reminders according to the latest information.

Note: This happens at noon as generally the clinics require up until the morning to process the appointment information from the day before. Sending it earlier generally results in wrong appointment information being sent because the MRS wasn't updated yet by the Clinic Admins.

1.2 MRS Integration

Currently txtAlert only integrates with one medical record system, [Therapy Edge](#).

Therapy Edge provides an XML-RPC interface where on a daily basis appointment information is available. txtAlert augments its own database of patient and appointment information with the information that is made available from this API.

1.3 Clinic Admin

txtAlert provides a Django based Clinic Admin on the `/clinic/admin/` path. This allows clinics without a Medical Record System to still update patients' appointment information and through that have SMS reminders sent out.

This interface is multitenant and allows for multiple logins for different clinics.

1.4 CD4 Admin

Certain Clinics have the ability to also send out CD4 counts to patients in bulk. This happens in the Django Admin on the `/admin/cd4/` path. The Clinic admin can upload an MS Excel file with the following columns in the following order:

1. Lab ID number

2. Cellphone number
3. CD4 count

An upload in this format will result in the CD4 counts being applied to the following message template:

```
Hello. Thanks for doing your CD4 test.
Your CD4 results are back. Your count is [COUNT].
Please report to %s as soon as possible for further treatment.
```

txtAlert will update the messages with the delivery status notifications when they arrive from the mobile networks.

1.5 Mobi Bookings

There is a simple Mobi based admin designed for small scale use. This is available on the `/bookings` path.

It provides both a patient and a clinic admin interface.

The patient interface allows the patient to view appointments and request a change.

The clinic admin interface allows a clinic admin to manage patients' appointments.

Note: The bookings interface is currently not being used nor is it being actively maintained.

1.6 GDocs Integration

At one point txtAlert allowed for importing of patient records from Google Docs based spreadsheets. That functionality has been removed as it was error prone and difficult to maintain. The Clinic Admin has replaced that functionality.

Documentation for txtAlert API

2.1 Please Call Me API documentation

A “Please Call Me” is a text message that is sent from one subscriber to another where asking the latter to call the sender of the message back.

PCMs (Please Call Me’s) are usually sent by dialing a specific USSD code and arrive at the receivers handset as a sponsored SMS message informing the receiver of the request.

In South Africa all major telecom operators provide a number of free PCMs, to all subscribers. This allows for those who cannot afford credit on their phones to still communicate when needed.

In txtAlert PCMs are used as a low-cost means for the patient and the clinic to get in touch. When a patient is unable to attend an appointment, he or she can send a PCM to txtAlert. This PCM is registered by txtAlert and the patient is identified by the MSISDN. The clinic is notified of the PCM and will call the patient back to schedule a new appointment.

2.2 Registering PCMs

URI /api/v1/pcm.json

HTTP Method POST

2.2.1 Parameters

sender_msisdn The MSISDN sending the PCM

sms_id Unique ID from the SMSC

recipient_msisdn The MSISDN receiving the PCM

message_content The original PCM message (optional)

or

Using HTTP to post the JSON of [Vumi](#) message to the URL.

2.2.2 Example

An example of HTTP POSTing with parameters with cURL:

```
$ curl --user 'user:password' \  
> --data 'sender_msisdn=271234567890&sms_id=abfegvcd&recipient_msisdn=271234567890' \  
> http://localhost:8000/api/v1/pcm.json  
Please Call Me registered
```

FrontlineSMS supports this out of the box. Check the [FrontlineSMS](#) documentation for how to do this.

An example of HTTP POSTing JSON from Vumi's APIs with cURL:

```
$ curl --user 'user:password' \  
> --data '{"from_addr": "271234567890", "to_addr": "271234567890", "content": "message", "message_id": "1234567890"}' \  
> http://localhost:8000/api/v1/pcm.json
```

Both API calls:

1. return an HTTP 201 Created header if successful.
2. return an HTTP 409 Conflict if an exact same SMS was received in the last 2 hours.
3. return an HTTP 400 Bad Request if not all parameters are present.

2.3 Sending SMSs

SMSs are sent via Vumi Go's HTTP API. txtAlert should have an account configured and appropriate token set to allow for outbound messaging.

2.4 Receiving Delivery Reports

The path for receiving network acknowledgements & delivery reports is:

```
/api/v1/events.json
```

This URL endpoint expects a Vumi event message. Event messages of type `ack` and `delivery_report` are supported. These update the original outbound message with the appropriate txtAlert status matching pending, failed or delivered.

Indices and tables

- `genindex`
- `modindex`
- `search`