System Design Document

24 November 2014

For:

Rape Crisis Cape Town Trust

360 Software Development

360 SOFTWARE DEVELOPMENT

Rape Crisis Mobile Application

Team :

- Lawrence Stent
- Daniel Louw
- Thierry - Luc Denichaud
- Adrian David Smith

Contents

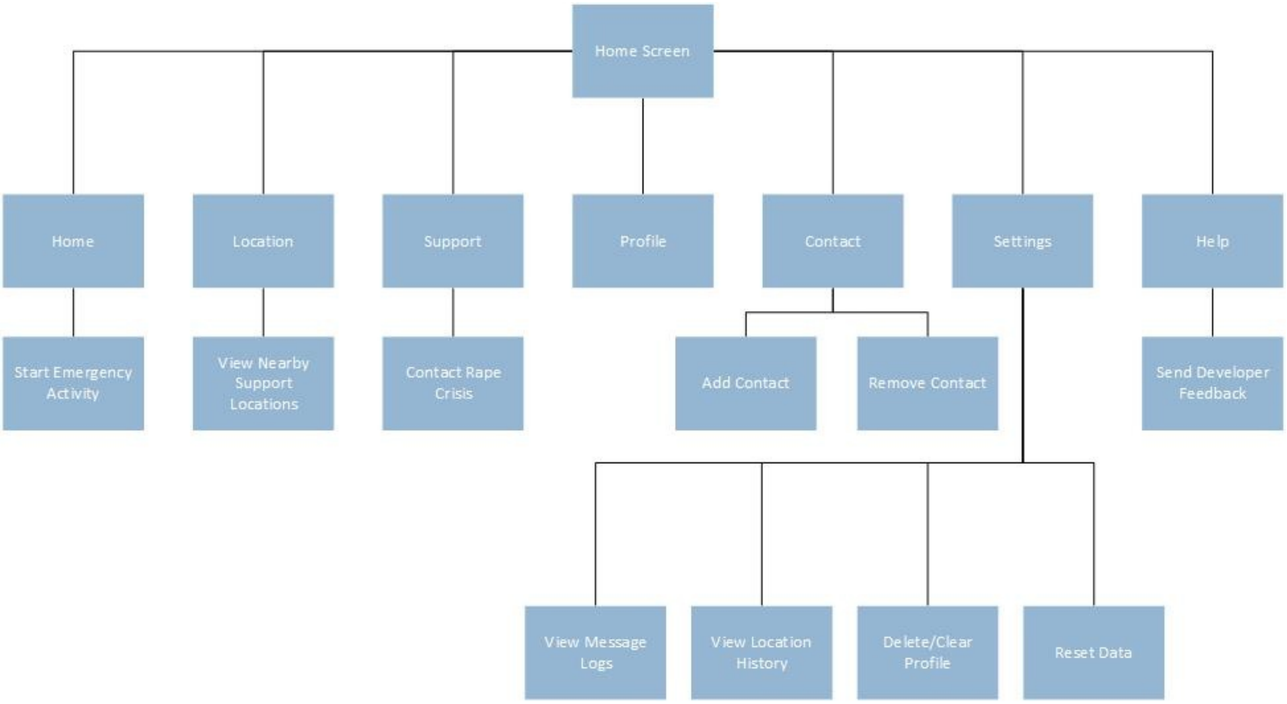
Index of Figures

Introduction

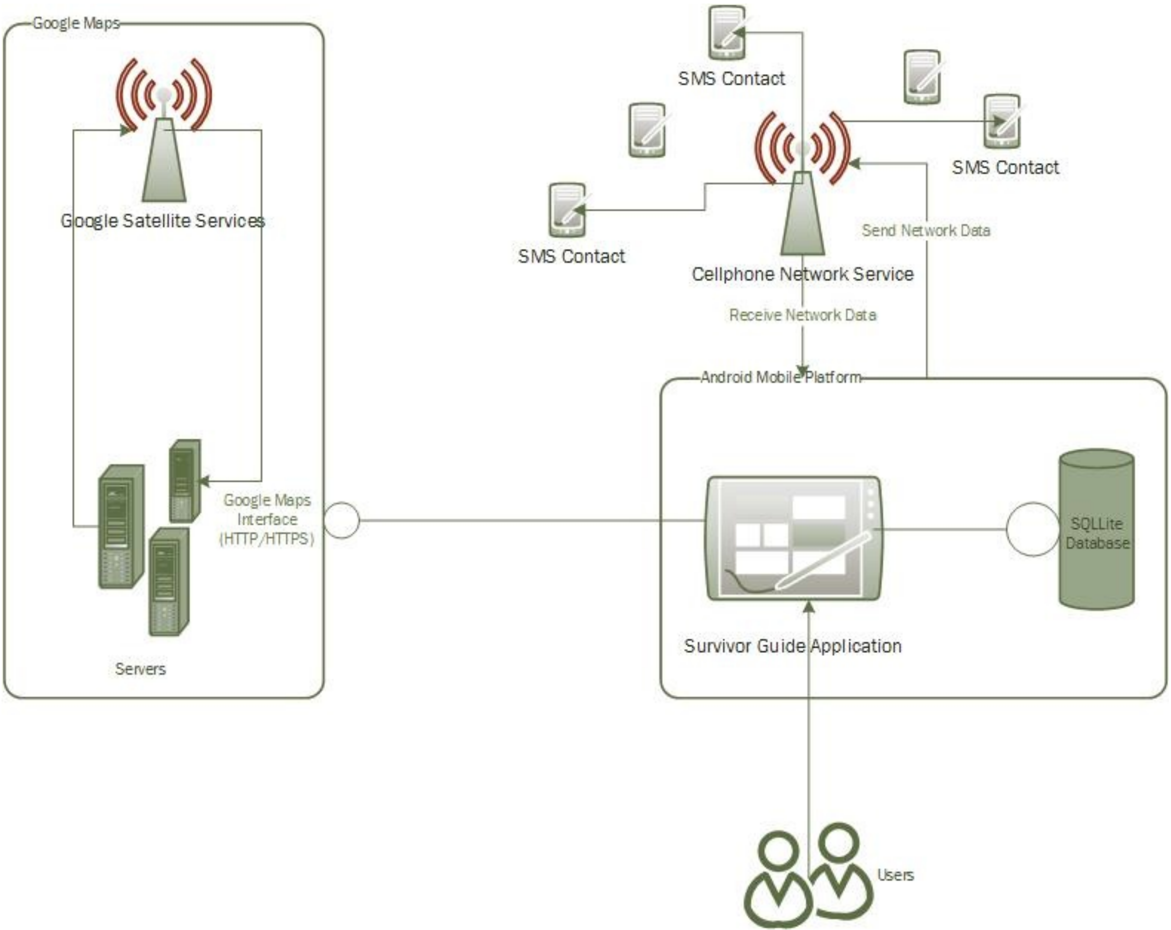
This design brief will represent the high and low levels of interaction between the user and the application. Along with simple and easy to understand illustrations. The system accepts the user's personal details allowing him/her the accessibility to use the application and get into contact with the necessary support structures offered by the application.

Within this brief the users of the application will be identified, the functions of the application and the database foundation and structures represented.

High Level Architectural Design
Fig 1.a

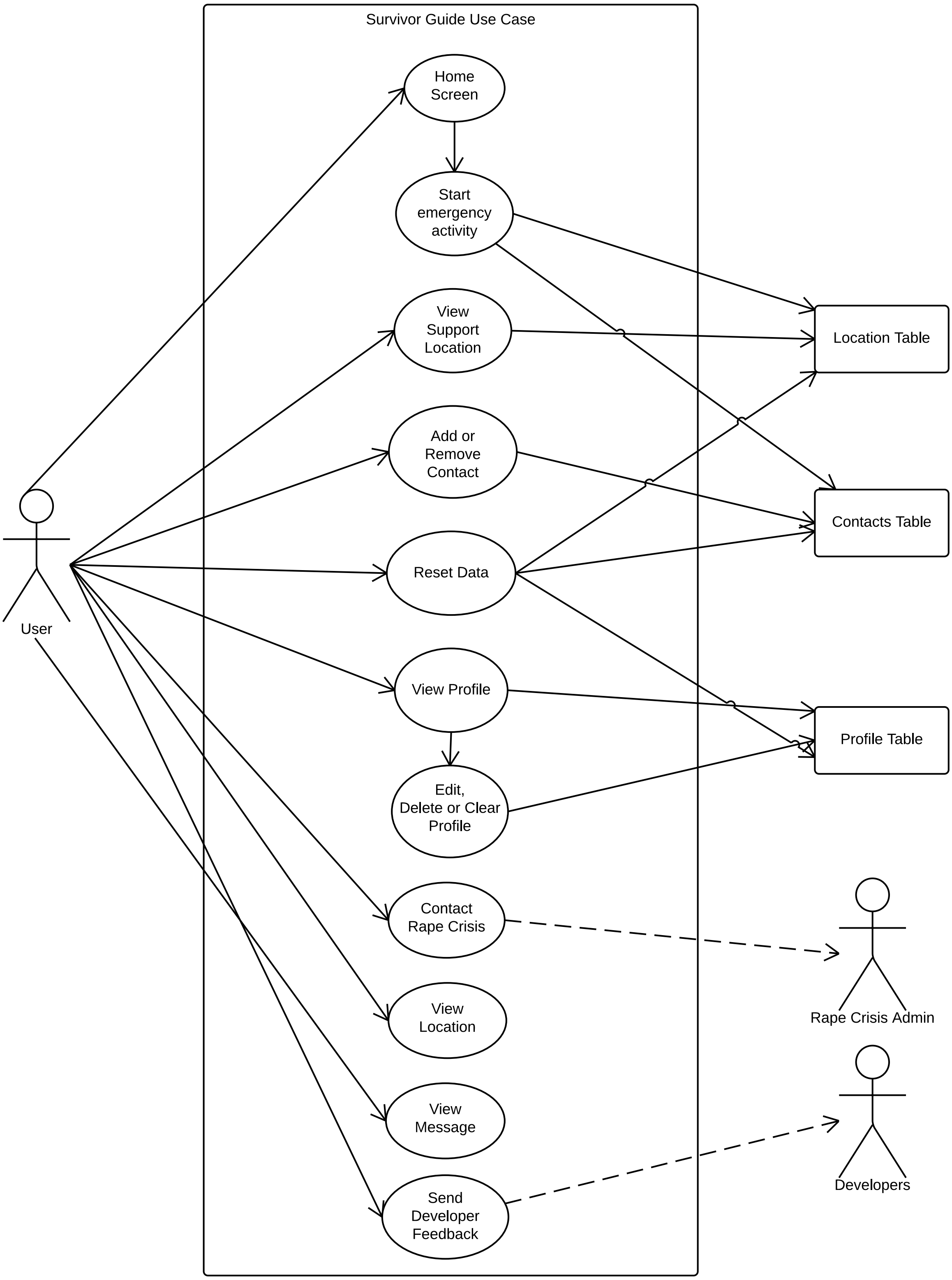


Network Architecture
Fig 1.b



Low Level Architectural Design

Fig 2

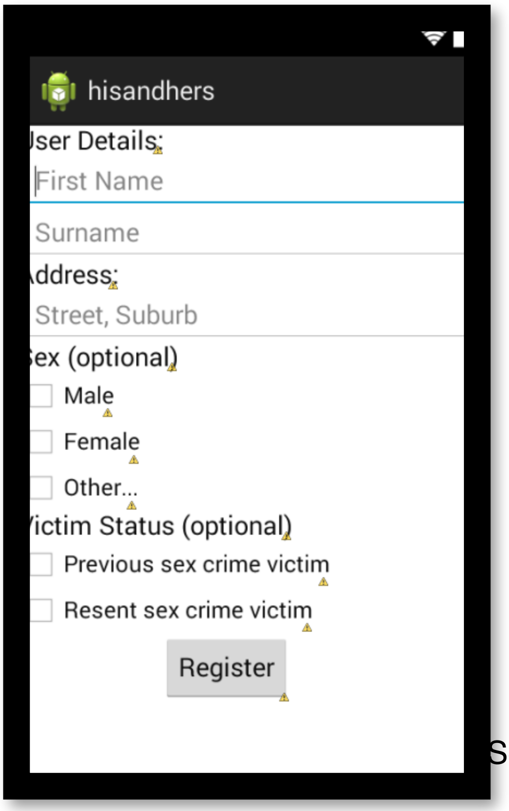


Input Interactions

On registering with the application the user will be requested to fill in the following (**Fig3.a**):

Main Screen on register:	User personal Information:	Additional Information:
1st level	User Details	
2nd level	Name	
2nd level	Surname	
1st level	Address	
2nd level	Street	
2nd level	House number	
2nd level	Suburb	
1st level	Phone/contact	
2nd level	Cellphone	
2nd level	Land line	
2nd level	Email	
1st level		Sex (optional)
2nd level		Gender orientation
1st level		Victim of a sex crime (optional)
2nd level		Previous Victim
2nd level		Recent victim
2nd level		Support

Fig 3.b



hisandhers

ser Details:

First Name

Surname

ddress:

Street, Suburb

ex (optional)

□ Male

□ Female

□ Other...

ictim Status (optional)

□ Previous sex crime victim

□ Resent sex crime victim

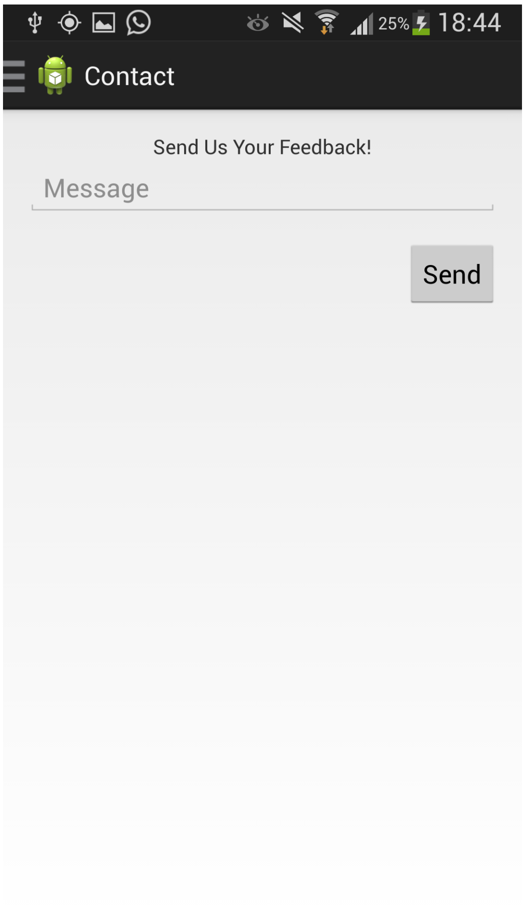
Register

Above (**Fig 3.b**) is a Pilot version of the register page and the required inputs needed by the user.

The user will have to provide the system/application with his/her details such as name, surname and address so that the company helping the user will have a physical address on where to locate and help the victim. The check boxes are additional options that are provided for the user. The information will guide Rape crisis on how they wish to act.

The additional information such as "Victim of a sex crime", the reason for this is if the user fill in recent victim, as soon as the information is received Rape Crisis Cape Town Trust system will flag this person's name making them a top/high priority to get into contact with.

Fig 3.c



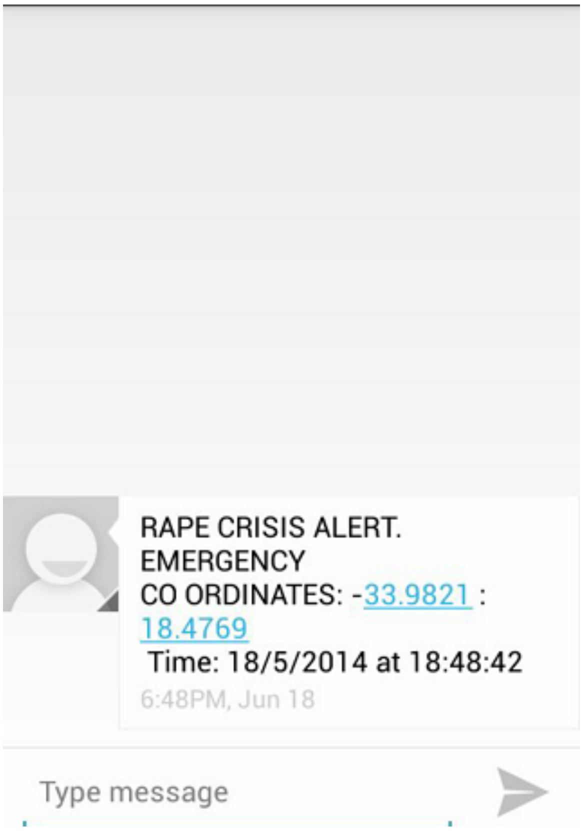
Above (**Fig 3.c**) shows the feedback page where the user can email the developers with their feedback and it will get sent to our developer inbox.

Request Interactions

Requests interactions represent all service requests put to the system and include requests for functional processes and outputs in the form of screen displays and reports. The following section shows the interaction results in the form of the GUI. In this document there are several screenshots that show the result of a request interaction made by the user within the program. In this instance, since it is a mobile application, the interactions are focused on achieving the goal of the app which is an emergency messenger when someone feels that they are in trouble. Keeping this in mind while prototyping, these are the planned outputs of the functions within the application.

This is the screenshot of the message for the emergency SMS sent to all the selected contacts of the user. Within this SMS are the coordinates of the user that feels that they are in danger. These coordinates are clickable and, when pressed, opens their cellphones Google Maps application and zooms on the coordinates shown in the SMS.

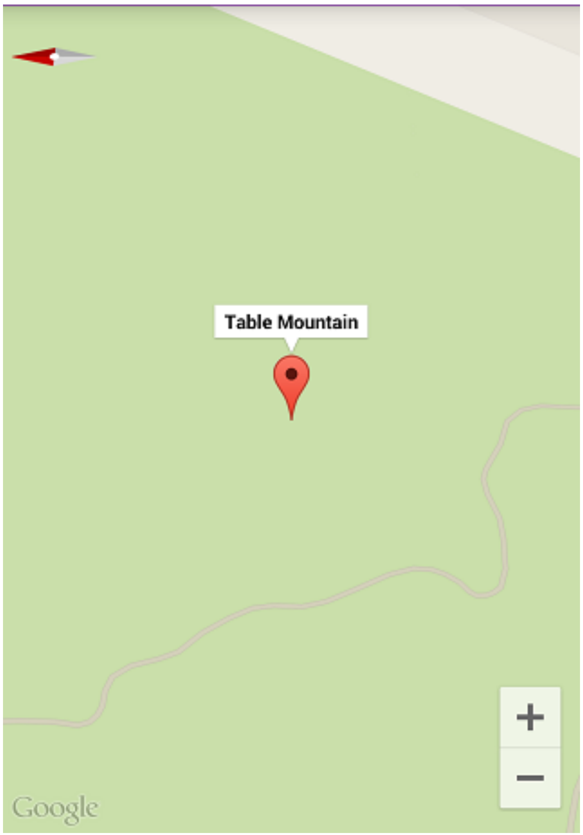
Fig 3.d



RAPE CRISIS ALERT.
EMERGENCY
CO ORDINATES: -33.9821 :
18.4769
Time: 18/5/2014 at 18:48:42
6:48PM, Jun 18

The following screenshot shows an example output of what the application will output when the coordinates from the SMS are pressed:

Fig 3.e

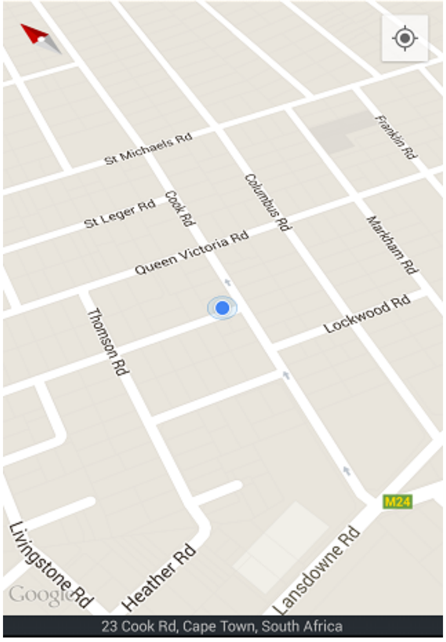


In this example, if the user had pressed the emergency button on Table Mountain, a marker would be placed on the location the SMS was sent from with the location information shown above it.

]

Another example of Google Maps and Location outputs would be when the location page is selected within the application. This page will get the users current location and output the locations address at the bottom of the screen. This is designed to show the user that their device has accurately identified their current location and that their device is ready in case of emergency. This may also double as just a map that will help the user find out where they are in case they feel lost.

Fig 3.f

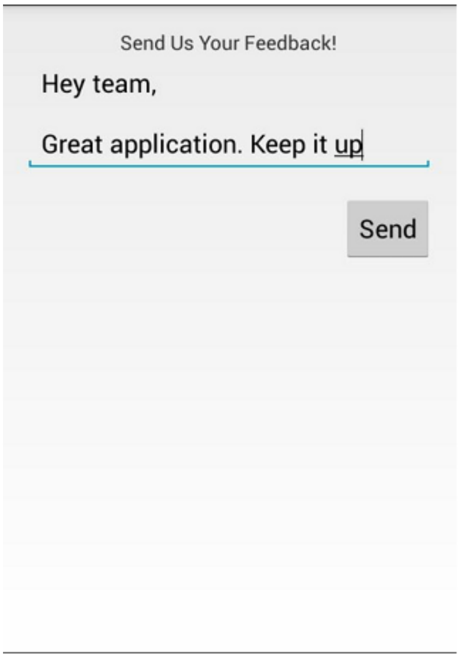


23 Cook Rd, Cape Town, South Africa

As shown in the above screenshot, the user is shown their current address and their current position via the blue icon in the center of the screen. This is the map view of Google Maps so that they can see the street names and locations in case they feel that they need directions.

The application in its prototype state has the option of sending Email feedback Via the Gmail application that android provides its users. The following screenshots show the process of sending the feedback in this manner and the outputs of this process.

Fig 3.g

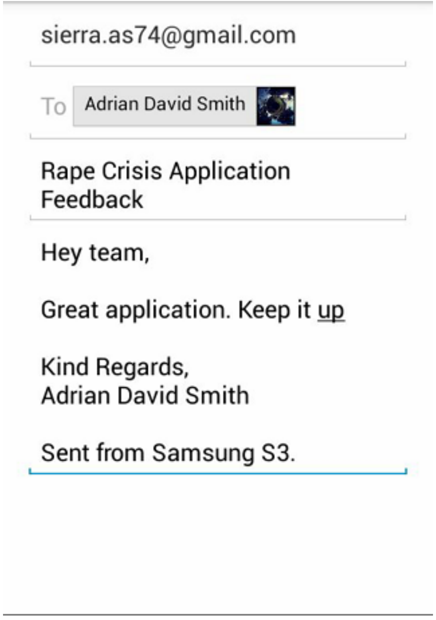


Hey team,

Great application. Keep it up

This is the screenshot of where the user may input their feedback and hit the send button. Once the send button is pressed they are presented with this screen as the output:

Fig 3.h



sierra.as74@gmail.com

To Adrian David Smith

Rape Crisis Application Feedback

Hey team,

Great application. Keep it up

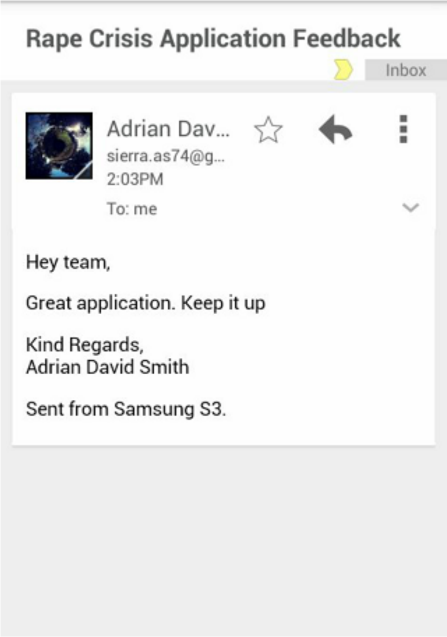
Kind Regards,
Adrian David Smith

Sent from Samsung S3.

This is automatically generated from the input the user provided in the previous screenshot. This feature will automatically fill out the development teams Email address and populate the Email body.

The following screenshot is the final output of the email sent to the development team as feedback about our application:

Fig 3.i



Rape Crisis Application Feedback

Adrian Dav...
sierra.as74@g...
2:03PM
To: me

Hey team,

Great application. Keep it up

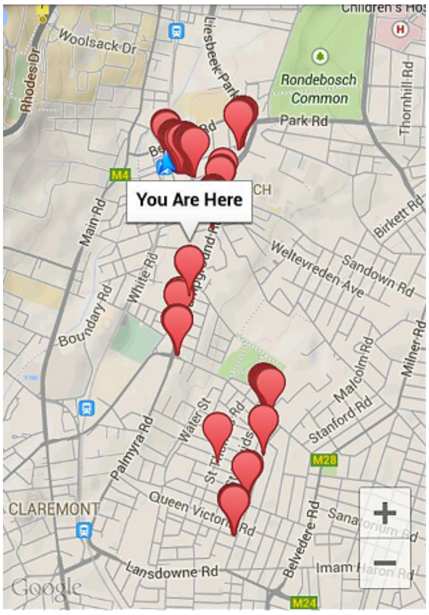
Kind Regards,
Adrian David Smith

Sent from Samsung S3.

As shown above this is the final output of the user feedback email generated from the users input within the application. All output is handled by the native Gmail application that all android devices come with.

Finally we have the report output of our application. Since this application is more about providing emergency messaging functionality, reports do not really play a major part in the application being developed. The final screenshot shows a report generated once the user presses the emergency button in the application:

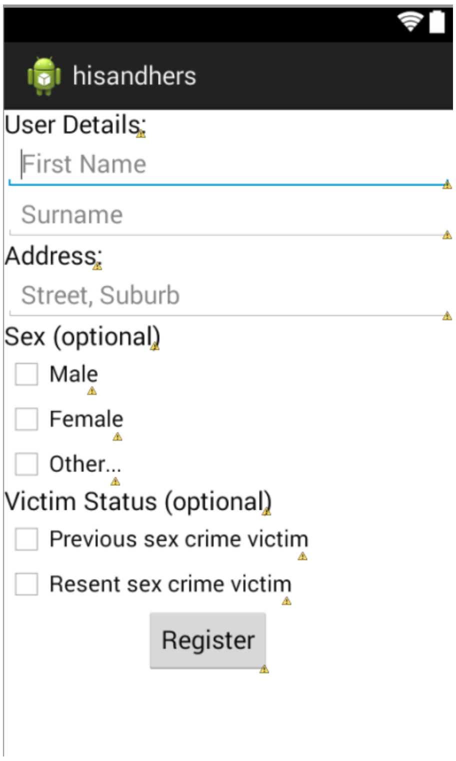
Fig 3.j



In this screenshot, each marker on the map represents the user's location being updated as they move around every few seconds. This report logs every point the user has been between pressing the emergency button and stopping the emergency broadcasts. This is the output generated when the user selects to view this report. This is useful for possibly highlighting areas that are dangerous as well as the possibility of providing evidence that the user was moved around during an attack. It may also be useful to track the place that any event may have occurred.

The following screenshot is of a possible profile menu that allows for the user to create a profile for themselves and store their personal information on their device. What this allows for is the application will have instant access to all necessary information when it is needed since all information on this form is captured directly into the database.

Fig 3.k



User Details:

First Name

Surname

Address:

Street, Suburb

Sex (optional)

□ Male

□ Female

□ Other...

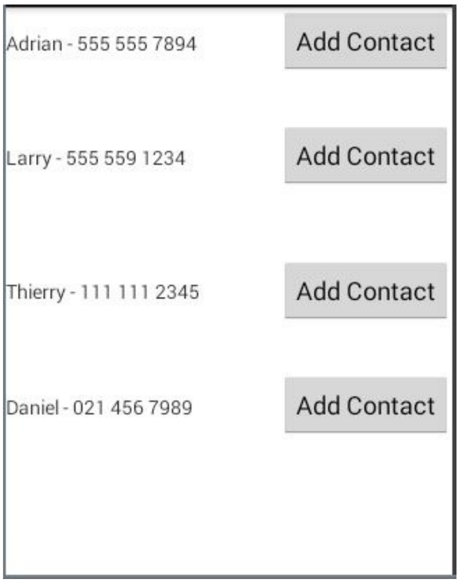
Victim Status (optional)

□ Previous sex crime victim

□ Resent sex crime victim

As shown above the user is asked to provide their personal information and then press the register button. Once this button is pressed the user's information is stored on the database in the system. Once the data is captured the form then searches the database for the user's information and populates the profile page automatically whenever the user views their information.

The contacts screen of the application allows for the user to select which contacts they want as an emergency contact. These are the people they want to alert if they feel that they are in trouble. In the following screenshot the user is shown the contacts name as saved on their device and a button for each contact that will allow for the user to select the contact as their emergency contact.


Adrian - 555 555 7894

Larry - 555 559 1234

Thierry - 111 111 2345

Daniel - 021 456 7989

Fig 3.l

As shown above, the contacts stored in the contact book of the user's device along with the numbers for each of the contacts are displayed to the user. Once the user presses the add contact button the contacts information is stored in the database. Once this happens and the user presses the emergency button the contacts selected here are notified with the message shown earlier in this document.

Database Design

Database Tables

Location Table

Primary Key	Data fields			
locationID	locationLatitude	locationLongitude	locationAddress	locationCreated
101	33.56	18.23	23 Field Street, Cape Town	22:34 - 4-12-2014
102	33.45	17.45	4 Atlas Road, Bellville	00:12 - 6-11-2014
103	33.02	18.95	99 Signal Hill Road, Cape Town	04:56 20-11-2014

Profile Table

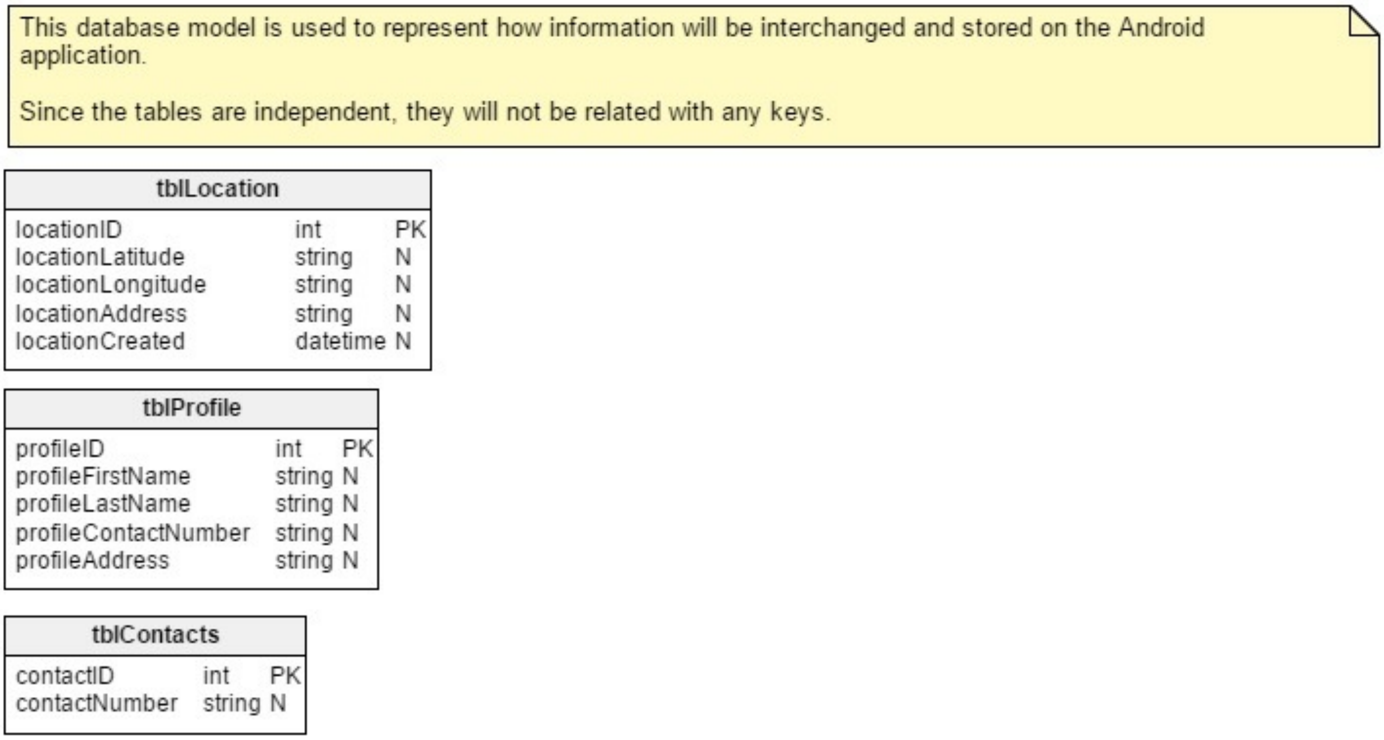
Primary Key	Data fields			
profileId	profileFirstName	profileLastName	profileContactNumber	profileAddress
10001	Bob	Smith	+06862363	34 Mount Road, Cape Town
10002	Tom	Chalk	+0823451234	26 Fir Lane, Cape Town
10003	Andy	Glass	+0742231111	101 Kirstenbosch Drive, Cape Town

Contact Table

Primary Key	Data fields
contactId	contactNumber
30002	+0214565748
30003	+0763456267
30004	+0765463722

Entity Relationship Diagram

Fig 4



This database model is used to represent how information will be interchanged and stored on the Android application.

Since the tables are independent, they will not be related with any keys.

tblLocation

locationID	int	PK
locationLatitude	string	N
locationLongitude	string	N
locationAddress	string	N
locationCreated	datetime	N

tblProfile

profileID	int	PK
profileFirstName	string	N
profileLastName	string	N
profileContactNumber	string	N
profileAddress	string	N

tblContacts

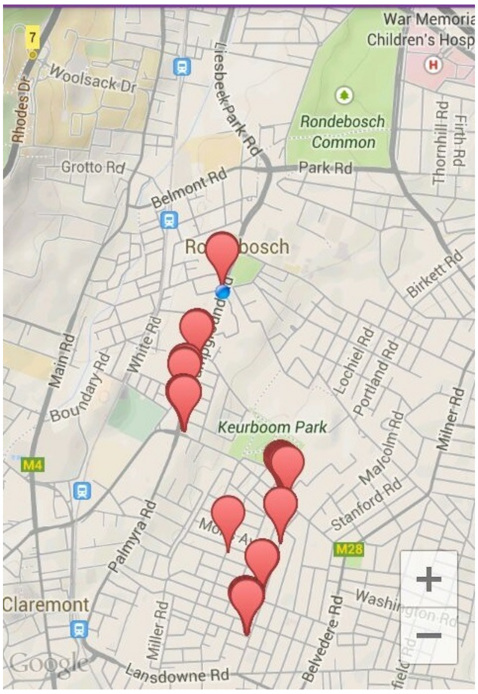
contactID	int	PK
contactNumber	string	N

Reports

The following reports are designed based on the data that will be accessible via the database and text files that are stored on the mobile device respectively. These reports will be viewed by the user and cannot be manipulated in any way. The user will be given the option to reset and clear the data of these reports.

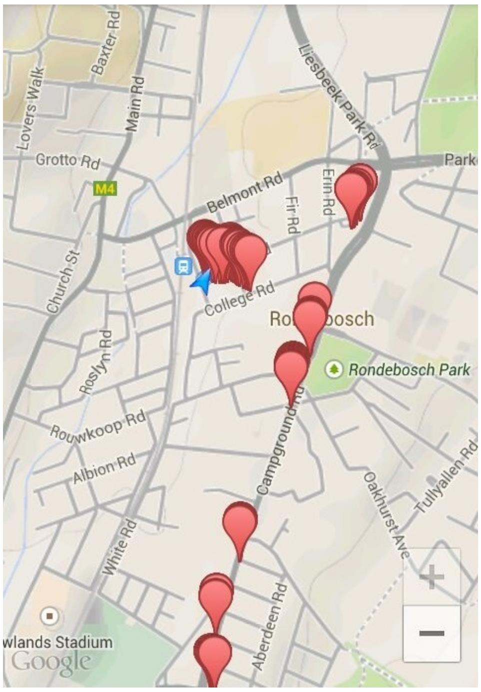
One key area of focus is how the data will be displayed to the user. We have chosen that when the user logs their location via the Emergency option that it stores it in the database and provides live feedback of location points on an interactive map. This will show the user where they are and what information was sent through to other contacts. This is shown below by Figure 5.a

Fig 5.a



The user will be given the option to view their location history via the application. This option will display an interactive map to the user and display all the geo-points that were logged during an emergency within the database. These results shown will be of all the data and not the most recent emergency session that the user has engaged in. This is shown below by Figure 2. The data within this report can also be reset and the database results cleared when the user selects such an option in the “Settings” menu.

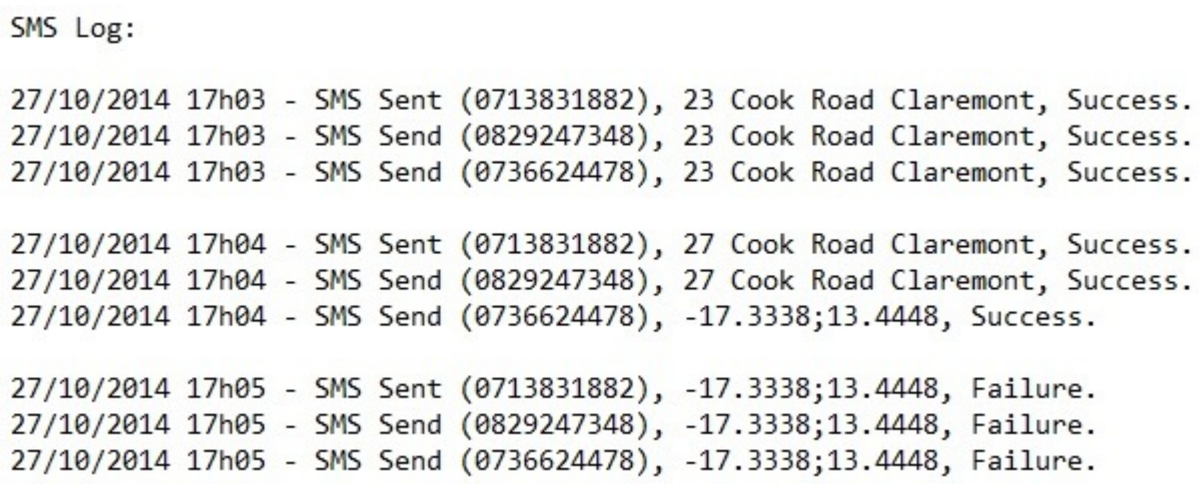
Fig 5.b



Another key report that will be focused on and displayed to the user will be the SMS log. This log will be a file stored locally on the mobile device and will contain information regarding which contact was messaged, the location of the device when the message was sent, the date of the message and if the message was sent successfully or failed. The report below (Figure 3) demonstrates this along with the location of the device. If an address was failed to be retrieved from the device, a geo-point will be sent. If the device does not have network connectivity or is unable to send an SMS to a contact, a "failed" will be logged and displayed to the user. This report must be unambiguous and easily understood by the user and a simple text output on the device was chosen.

The user will have an option to reset this data and clear this report in the "Settings" option.

Fig 5.c



```
SMS Log:

27/10/2014 17h03 - SMS Sent (0713831882), 23 Cook Road Claremont, Success.
27/10/2014 17h03 - SMS Send (0829247348), 23 Cook Road Claremont, Success.
27/10/2014 17h03 - SMS Send (0736624478), 23 Cook Road Claremont, Success.

27/10/2014 17h04 - SMS Sent (0713831882), 27 Cook Road Claremont, Success.
27/10/2014 17h04 - SMS Send (0829247348), 27 Cook Road Claremont, Success.
27/10/2014 17h04 - SMS Send (0736624478), -17.3338;13.4448, Success.

27/10/2014 17h05 - SMS Sent (0713831882), -17.3338;13.4448, Failure.
27/10/2014 17h05 - SMS Send (0829247348), -17.3338;13.4448, Failure.
27/10/2014 17h05 - SMS Send (0736624478), -17.3338;13.4448, Failure.
```