

# Installation and Usability Manual

Animal Tracker And Health Monitoring Application

**Name: Gearoid Lacey**

**Student Number: C00183380**

**Due date: 5<sup>th</sup> April 2017**

<b>1 Introduction.....</b>	<b>3</b>
1.1 Installation Instructions .....	3
<b>2 Mobile Application.....</b>	<b>4</b>
2.1 Home screen .....	4
2.2 Register screen .....	4
2.3 Change Password .....	5
2.4 Login screen.....	6
2.5 Menu screen.....	7
2.5.1 Add Animal Profile screen.....	8
2.5.2 Create Boundary screen.....	9
2.5.3 Latest Location screen.....	10
2.5.4 Analyse Paths screen .....	11
2.5.5 GPS Configuration screen.....	12
2.5.6 Cluster Locations screen.....	13
2.5.7 Update Animal Profile screen.....	14
2.5.8 Delete Animal Profile screen.....	15
2.5.9 Logout Menu Option.....	16
<b>3 NanoTracker .....</b>	<b>17</b>

# 1 Introduction

The purpose of this document is to provide an outline to the AniMap mobile application as well as to provide installation instructions for the applications

AniMap is a proof of concept project that aims to try and determine the source of Tuberculosis (hereby referred to as TB) in a herd of cattle by recording an animal's location in a field using a GPS device called NanoTracker. Usage instructions for the NanoTracker will also be provided

## 1.1 Installation Instructions

AniMap is a mobile application that is available for download for mobile phones running the Android or Windows Operating Systems. The installation instructions for AniMap have only been tested on Android devices although it should be a similar process for windows phones.

Step 1. Scan the QR code in the About section of the website.

Step 2. Open the downloaded APK file.

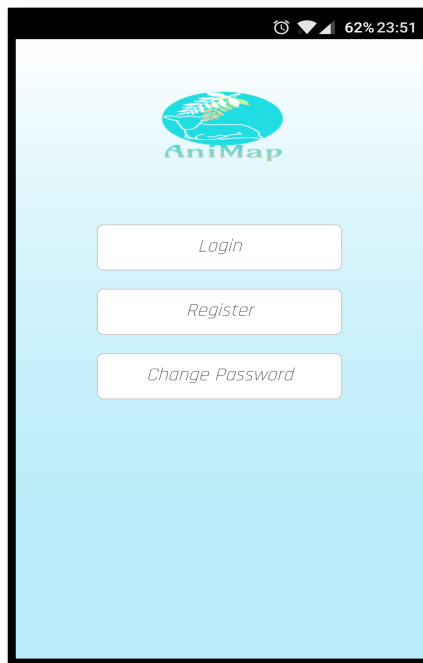
Step 3. Click Install. Notice the permissions alerts you to the fact that you may be charged for sending texts. This for the GPS Configuration and Create Boundary functions in section 2 of the document, where both functionalities text the NanoTracker to remotely configure it.

## 2 Mobile Application

The following sections outline how to use the AniMap application

### 2.1 Home screen

On opening the application the user is presented with a screen displaying three options. They are Login, Register or Change Password which can be seen in Figure 1. This page requires no backend functionality, as the user will only be moving to another page.

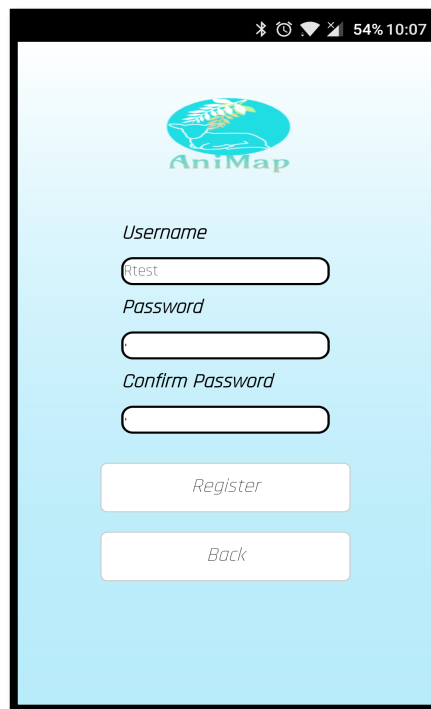


*Figure 1. Home screen*

### 2.2 Register screen

If the user clicks the Register button they are redirected to the Register screen. They are then required to enter a username and password. They are also required to confirm their password to eradicate any potential errors in the password creation process. When the user clicks the register button the form data is sent using jQuery to the Flask API. For this project all validation is performed in the API. On the Register screen in particular there are checks in place to determine if the user has filled in all the fields, if the username is not

already taken and that both password fields match. If any of this validation fails the user is presented with an appropriate error message describing what was wrong. On successful registration the user is automatically logged in and redirected to the Menu screen. Note all passwords are hashed before they are stored in the database.

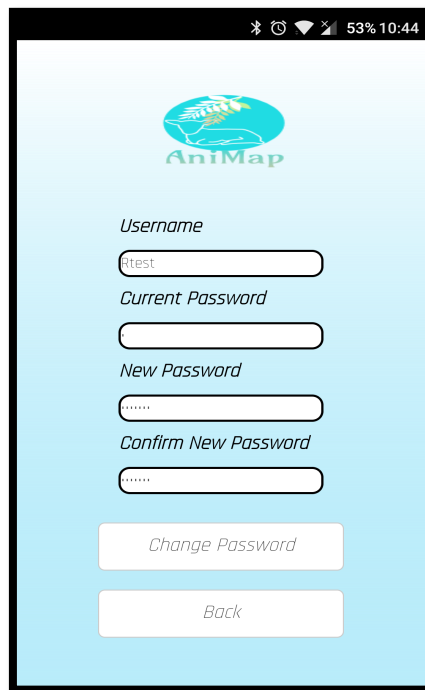


*Figure 2. Register screen*

### 2.3 Change Password

If the user clicks the Change Password button they are redirected to the Change Password screen. They are then required to enter their username and password. Following this the user may enter a new password which they must match in the confirm password field. The validation on this screen consists of checking the user has not left any fields blank, their username and current password are correct and that the new password and confirmation of the new password match. In this case a successful scenario would result in the users password

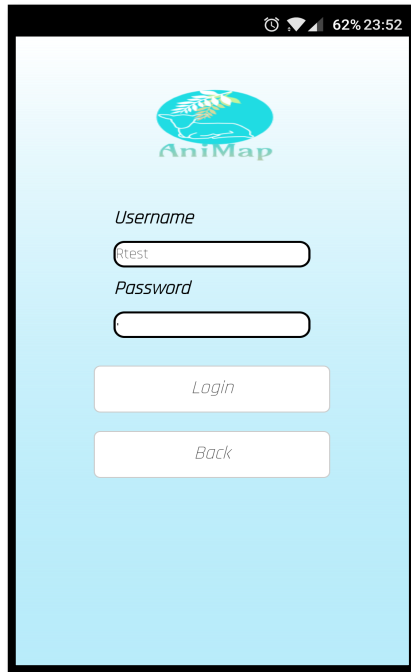
being updated in the register table. The user would also be automatically logged in and redirected to the Menu screen. Note all passwords are hashed before they are stored in the database.



*Figure 3. Change Password screen*

## 2.4 Login screen

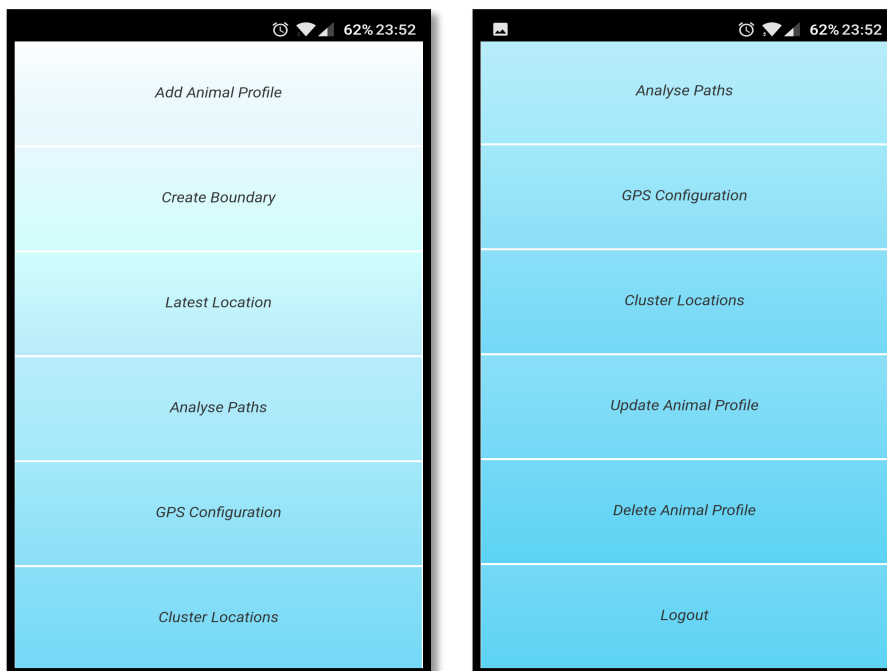
Assuming the user already has an account and has selected the login option the user then advances to the Login screen. The login in form is quite small which makes for easy access for an already registered user. The validation undertaken on this screen consists of checking that no fields are left blank, checking that the user exists in the database and that the entered password is correct. On successful login the user is redirected to the Menu screen.



*Figure 4. Login screen*

## 2.5 Menu screen

As mentioned in the previous use cases the success scenario of each results in a redirect to the Menu screen which will be displayed over two images as the user is required to scroll through the menu to see all the available options.



*Figure 5. Menu screen*

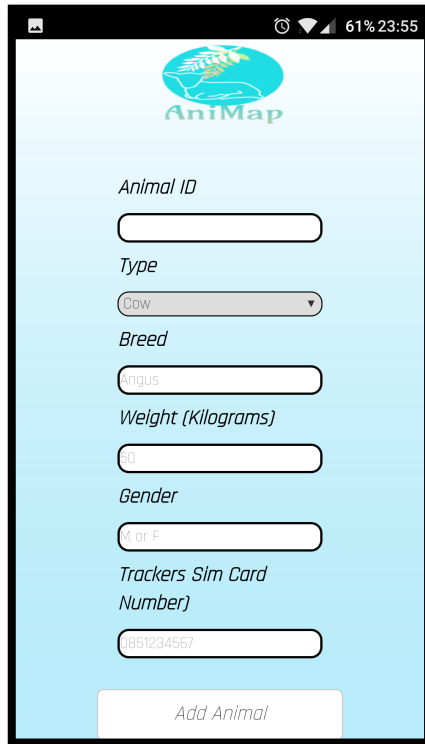
### 2.5.1 Add Animal Profile screen

On selection of the Add Animal Profile option the user is redirected to the Add Animal screen. This particular screen is one of the most densely populated screens in the application in terms of user input. As shown in Figure 6, there are multiple fields for the user to fill in. Firstly the user is required to enter a unique name for the animal, in the case of a herd of cattle, a cows tag number would be an appropriate option. Following this the user is required to select the type of animal from a dropdown list, enter the breed of the animal, enter the animals weight, their gender and the number of the sim card used in the NanoTracker.

Each input field is validated in the backend. As both the weight and tracking number are input fields of type number when the user clicks on this field a numeric keypad is displayed by default to prevent users entering letters.

Similarly to previous functionalities, on submission of the form it is validated to ensure no fields are empty. In a success scenario the animal details are added to the Animal table in the database.





*Figure 6. Add Animal screen*

### 2.5.2 Create Boundary screen

On selection of the Create Boundary menu option the user is redirected to a screen that is split to show a map and some input fields as shown in Figure 7. If there are no animals associated with their account then they are prompted to add an animal to their account and redirected to the Add Animal screen.

On the map there is a marker which the user can drag to any location they desire. This point on the map will then be used as the center of a boundary that will be defined by the user.

After selecting a point on the map the user is then required to select an animal from the drop down list of animal names associated with their account. Following this they then enter the radius of the boundary they wish to create. When the user clicks the Create Boundary button a text is sent to the NanoTracker containing the coordinates of the boundaries center point and the size of the radius.

Note, although the Create Boundary use case functions as documented above, the entire functionality could not be completed due to incorrect documentation provided by Round Solutions for the NanoTracker.

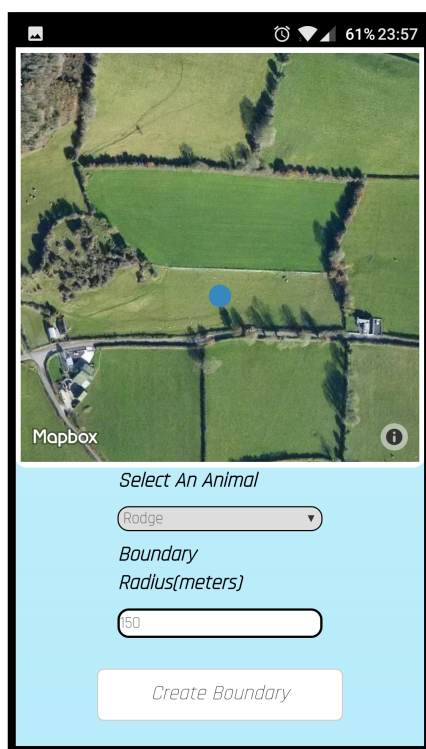
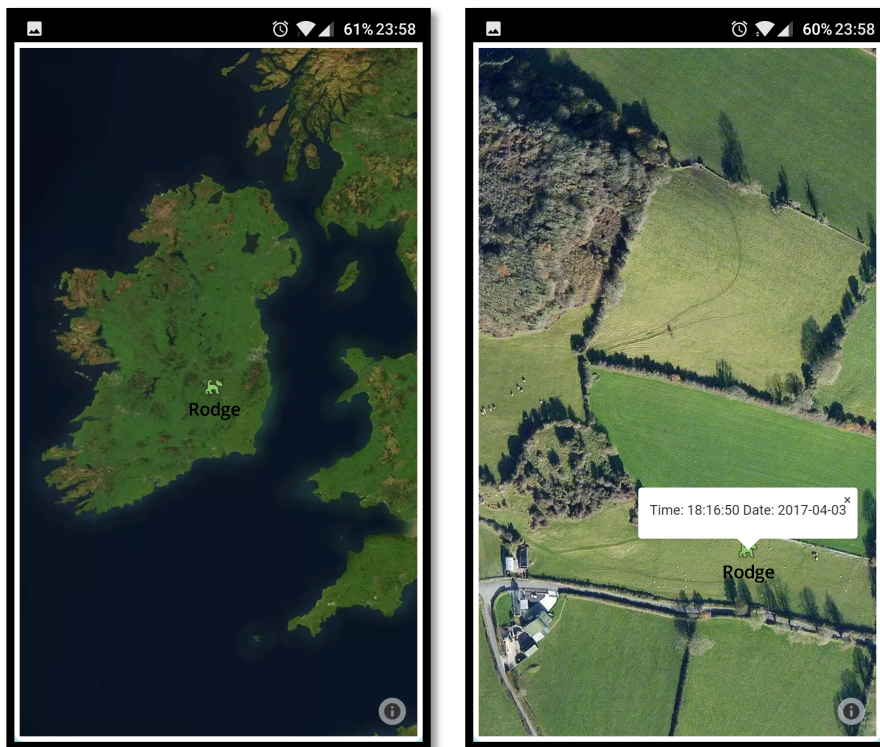


Figure 7. Create Boundary screen

### 2.5.3 Latest Location screen

On choosing the Latest Location menu option, a request is sent to the Flask API that retrieves the latest location of all the animals associated with the user. If no animals are associated with the user they are prompted to add an animal and redirected to the Add Animal screen.

Assuming there are animals associated with the user, a satellite map that is implemented using Mapbox is displayed with each of the animal(s) latest locations as shown in Figure 8. If the user clicks on the animal icon a popup displaying the date and time the location was recorded is displayed.



*Figure 8. Latest Location screens*

#### 2.5.4 Analyse Paths screen

If the user clicks the Analyse Paths option, they are then redirected to a page containing four drop down boxes. Like previous use cases if there are no animals associated with the users account then they are prompted accordingly.

In a successful scenario where the user has one or more animals associated with their account they are required to select two animals, a start date and an end date.

If for example the user only wants to view one animals path, they must then select the same animal in both animal selection drop downs as shown in Figure 9. The user must also select a start and end date that will be used to fetch the data. If the user sets an end date that occurs before a start date then they are prompted to re-enter the dates. Similarly to previous use cases a check for empty fields is also carried out.

Once the data has been successfully retrieved a satellite map is displayed which shows the specified animals paths. If the user clicks on an animal icon a popup displaying the date and time the location was recorded is displayed.

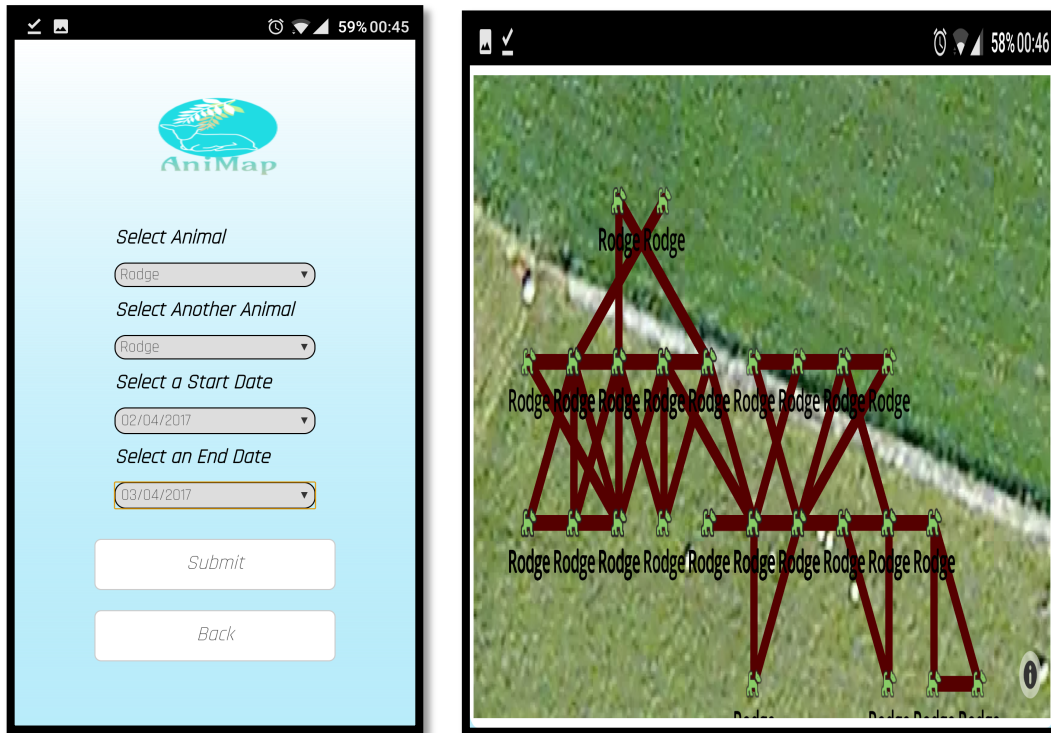


Figure 9. Analyse Paths screens

### 2.5.5 GPS Configuration screen

The purpose of this screen is to remotely configure the NanoTracker via text using a PhoneGap SMS plugin. When the user clicks this menu option they are redirected to a screen that requires them to select an animal, whose tracker they wish to configure, select a time metric and the amount of time they want between GPS requests. An example of user input is displayed in Figure 10.

In a successful scenario the application sends an SMS to the NanoTracker in the form **interval:120**, for example. The NanoTracker then receives this SMS and parses the message to extract everything after the colon and before comma. This

value is then used to force the device to sleep, in this case for 120 seconds, between each GPS request.

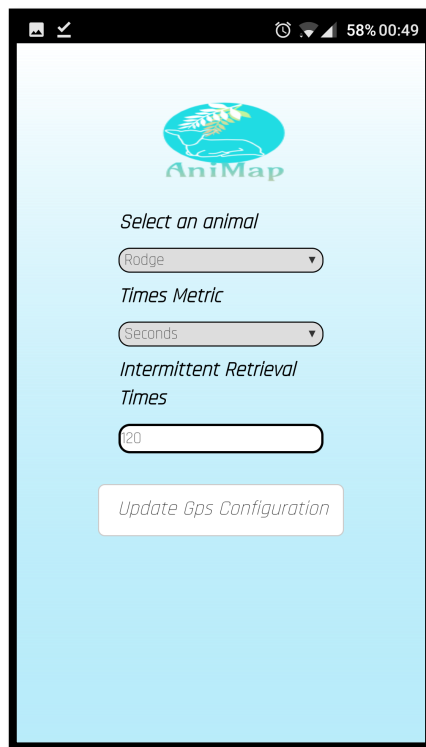


Figure 10. GPS Configuration screen

#### 2.5.6 Cluster Locations screen

This functionality could potentially be incredibly useful in determining the source of TB in a herd of cattle, in conjunction with some of the previously mentioned use cases. When the user selects the Cluster Locations option in the Menu, if there is not a sufficient amount of data for the clustering algorithm to take place the user will be alerted to this.

Assuming the user has a sufficient amount of data to perform the clustering of locations they will be redirected to a screen where they will be required to select from two drop down lists. They will also be required to select a start date for the data retrieval. On submission of these inputs the user will be presented with a satellite map which will display the location of each cluster. If the user clicks on a

cluster, information about each cluster will be displayed in a pop-up as shown in Figure 11.

In terms of the clustering algorithm, that will be discussed in more detail when documenting the API.

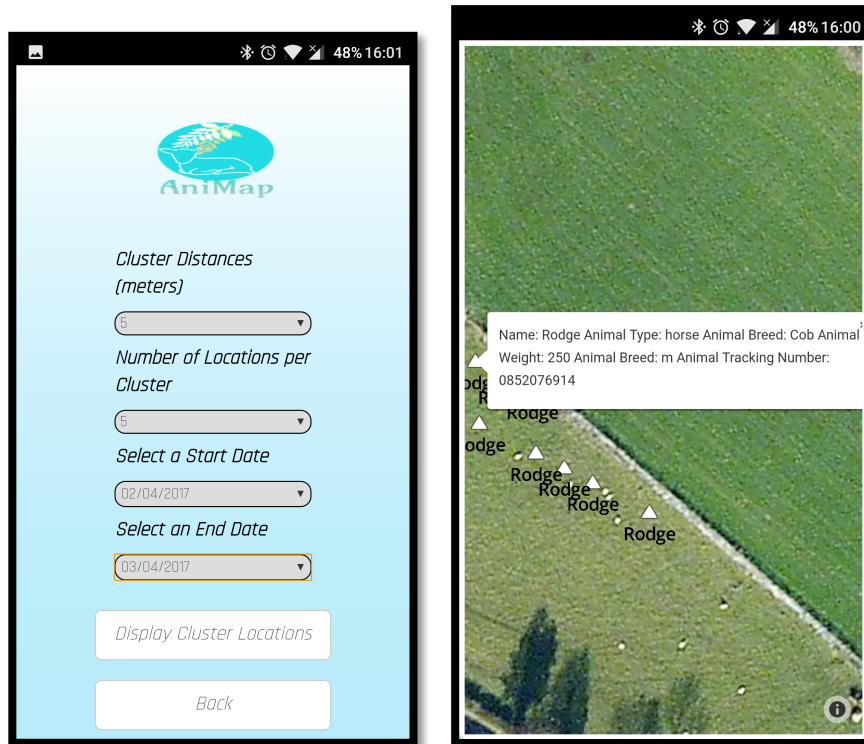
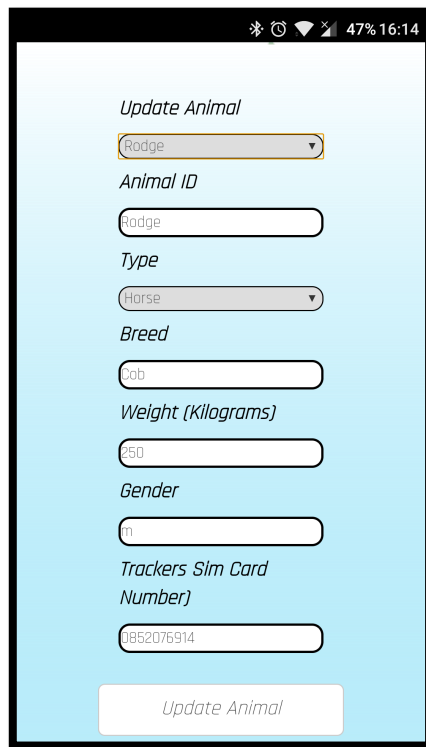


Figure 11. Cluster Locations screen

### 2.5.7 Update Animal Profile screen

As information relating to the animal may change over time, the user has the ability to update the animal's profile by selecting the Update Animal Profile option from the menu. The user selects one of the animals associated with their account from the dropdown list which then populates all the fields with the animal's information. The information in each of these fields can be changed as deemed appropriate by the application user as shown in Figure 11.

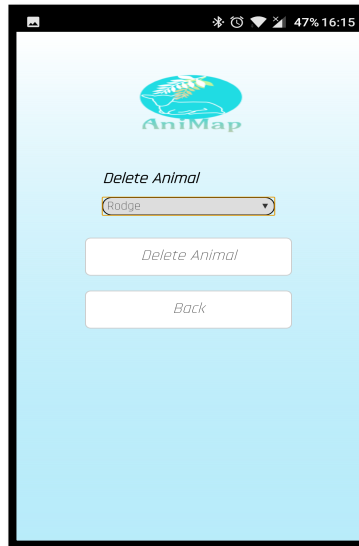
Similarly to previous use cases the user will be alerted appropriately if they attempt to update the animal's profile with blank fields.



*Figure 11. Update Animal screen*

### **2.5.8 Delete Animal Profile screen**

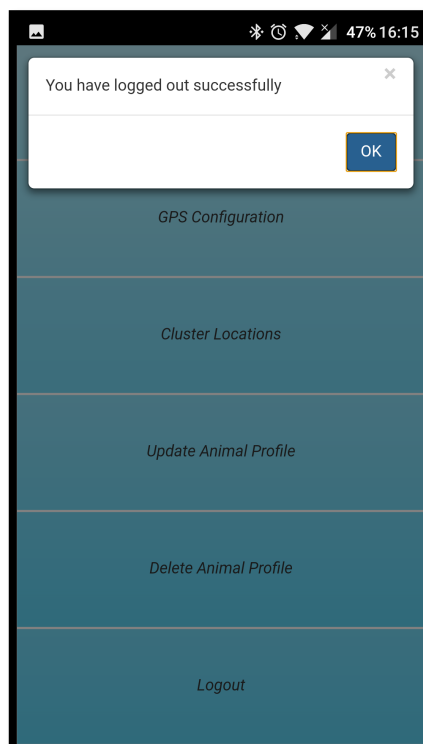
The Delete Animal Profile use case is very simple in its approach. Once the user selects the Delete Animal Profile option from the menu they are redirected to a screen which requires them to select an animal associated with their account from a drop down list. Once the user clicks the delete button all data relating to this animal from all database tables is removed. This use case is displayed in Figure 12.



*Figure 12. Delete Animal Profile screen*

### 2.5.9 Logout Menu Option

Like the Delete Animal Profile use case the Logout use case quite simple in its approach also. Once the user click this button a logged in flag in the Register database table is set to 0. The user is then redirected to the Home screen.



*Figure 13. Logout Menu Option*



### **3 NanoTracker**

The usage instructions for the NanoTracker are quite simple. Essentially all the user has to do is open the devices cover remove the contents of the case and plug the NanoTracker into a USB port for charging.