

---

# Android Design Document

---

*Data Communications: Big Brother 3000*

*Manuel Gonzales, Aoo866174, 4O*

*Georgi Hristov, Aoo795026, 4O*

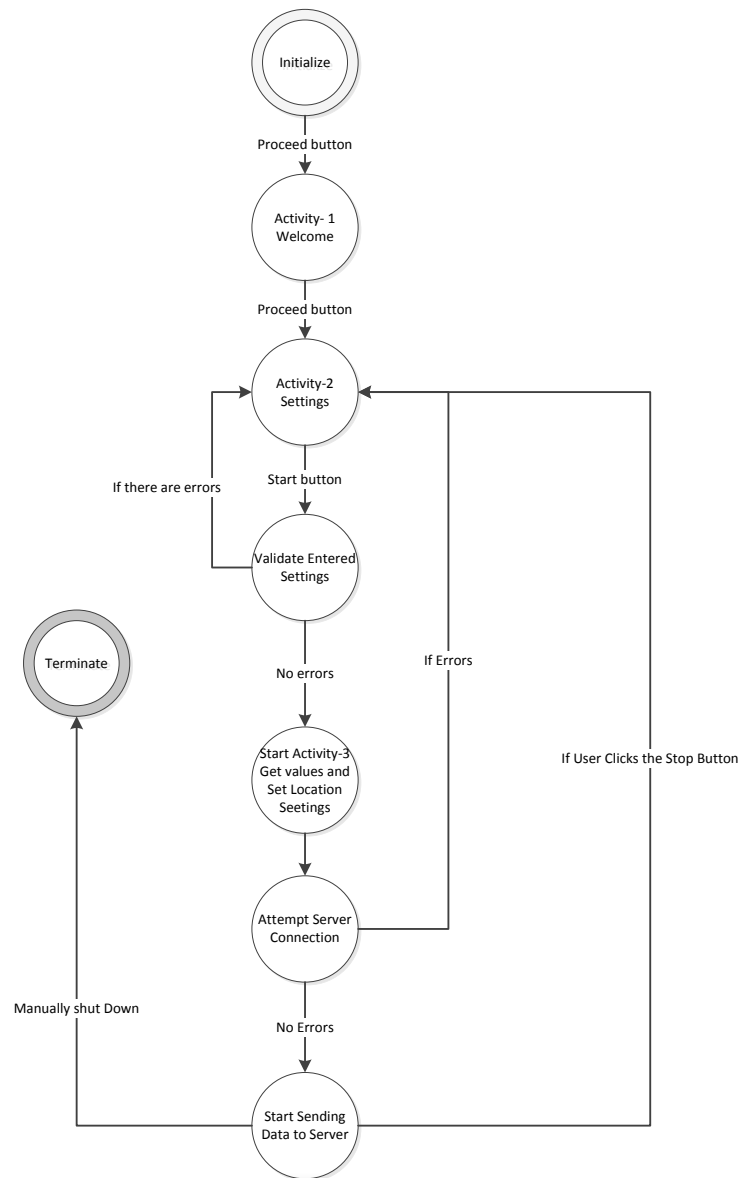
*Calvin Rempel, Aoo871348, 4O*

*Eric Tsang, Aoo841554, 4O*

## Table of Contents

State Transition Diagram.....	2
Android Pseudo Code .....	3
Main Activity.....	3
Layout.....	3
Activity.....	3
Settings Activity.....	3
Layout.....	3
Activity.....	3
Tracking Activity.....	4
Layout.....	4
Activity.....	4

## State Transition Diagram



The diagram above illustrates the states of the Android application:

- **Initialize;** the main activity is launched
- **Activity -1;** this is the welcome screen for the user and the press of the begin button will move the user into the second activity.
- **Activity -2;** this is the settings screen, will prompt the user for all necessary information before starting the watch and will validate all of the values entered before proceeding
- **Activity-3;** this is the main part of the program that will set all the user specified settings and will start sending the location updates to the server until the User presses the Stop button or if an error is encountered. In case of error the User will be sent back to the Settings activity.
- **Terminated;** the activity is no longer running.

## Android Pseudo Code

---

### Main Activity

---

#### Layout

---

- 1 A picture showing the Application name
- 2 Some text to lead the user
- 3 "Begin Watch" button to move to the next activity

#### Activity

---

- 1 Start the activity on Create
- 2 If (Begin Watch button is pressed)
- 3 {
- 4     Create an intent and move to the next Activity - Settings
- 5 }

### Settings Activity

---

#### Layout

---

- 1 Text indication Settings page
- 2 Text Fields for the IP Address and Port to be entered.
- 3 Radio Buttons so the user can choose between updates by Time or Distance
- 4 Text Field for the frequency desired by the user
- 5 Checkboxes to check if the user wants to use data (charges) or Wi-Fi
- 6 GPS is used by default.
- 7 "Start Watch" Button to continue, "Clear" Button to clear the fields

#### Activity

---

- 1 Start the activity on Create
- 2 Get all the values out of the text fields, radio button and checkboxes using their respective ID's
- 3 If (Clear button is pressed)
- 4 {
- 5     Clear all the text in the text fields.
- 6 }
- 7
- 8 If (Start Watch button is pressed)
- 9 {
- 10     Use all the values from the text fields, checkboxes and radio buttons to validate them
- 11     If (IP address is not a valid address) => toast Error and return
- 12     If (Port number is not a valid number) => toast Error and return

```

13     If (frequency is not a valid number) => toast Error and
        return
14
15     If all the values are okay put all this information into a
        Bundle/Intent.
16     Pass the intent into the next activity - Send Data
17 }

```

## Tracking Activity

---

### Layout

---

```

1 A picture showing that it is now sending data.
2 A gif image to mimic the sending of data to let the user knows
  it is running.
3 "Stop Watch" button to stop sending data and go back to the
  Settings
4 Activity.

```

### Activity

---

```

1 Start the activity on Create
2 Get all the values out of the bundle received from the
  Settings Activity. (IP Address, port, frequency,
  preferences)
3 Attempt Connection to the server.
4 If (Connection)
5 {
6     Start the Thread to send Data.
7     Get the mac address from the device and send it to the
        server to be used as a unique ID
8     Set the Location preferences criteria
9     Create the Location Listener
10    {
11        On Location Changes Send Data to the Server (New
        Location)
12    }
13    Request Location Updates based on the user Preferences
        (time/distance) and this will call the location
        listener on Updates.
14 }
15 Else
16 {
17     Let the user know
18     Return
19 }
20
21 If (Stop Watch button is pressed)
22 {
23     Close the Connection
24     Finish the Activity
25 }

```

```
26 Connection Thread
27 {
28     Connect to Server
29     If (there is a location update)
30     {
31         Send Data to Server
32     }
33
34     If (Connection is shut down/ Errors)
35     {
36         Free Resources;
37         Finish the activity;
38     }
39 }
```