Fall Emergency Distress System
Test Plan

Project Team: Cyrus Chi Chung Chan
                     Yuvin Ng
                     Benjamin Sia
                     Janet Mardjuki
                     Daniel Lei
                     WelsonYim

Contact Person: Cyrus Chi Chung Chan
                     cyrusc@sfu.ca

Submitted to: Andrew Rawics
                     Steve Whitmore
                     School of Engineering Science
                     Simon Fraser University

Issued: November 29, 2015

Revision Number: 1.0

System Test Plan

This test plan will be split into two sections. The first part involves testing the FEDS device, focusing on whether or not it can detect falls and ignore other daily activities. The second part concentrates on the server side of the system where the database and operator application are required to be tested.

FEDS Device Tests

The FEDS device test will focus on a couple of key ideas:

1. Battery life of FEDS
2. The device should be able to detect a fall
3. The device should be able to ignore daily activities such as walking and sitting
4. The device's user intractable hardware components/interfaces (Buzzer, Buttons, LED, Wi-Fi, On/Off switch and casing)

Note: When testing for falls, the tester will be falling onto a bed or a gym mat for safety precautions.

Device test 1: Powering the device and battery life

> **User Input:** The user should connect the battery to the Raspberry's 5V micro USB input
>
> **Expected Result:** The Raspberry should power on. Because the battery is modular, it can easily be replaced with a different battery with a different capacity; using the default battery should allow the FEDS device to operate up to 13+ hours with one full charge.

Device Test 2: Detecting a forward fall and checking status indicator

> **User Input:** The user should fall forward towards a bed or mat
>
> **Expected Result:** After the user falls forward, a low frequency buzz should trigger and the status indicator should turn yellow

Device Test 3: Detecting a sideways fall and checking status indicator

> **User Input:** The user should fall sideways towards a bed or mat
>
> **Expected Result:** After the user falls sideways, a low frequency buzz should trigger and the status indicator should turn yellow

Device Test 4: Detecting a backwards fall and checking status indicator

> **User Input:** The user should fall backwards to a bed or mat
>
> **Expected Result:** After the user falls backwards, a low frequency buzz should trigger and the status indicator should turn yellow

Device Test 5: Testing for false alarms while walking

> **User Input:** The user should walk normally for 1 minute
>
> **Expected Result:** The device should not detect a fall and the status indicator remains green

Device Test 6: Testing for false alarms when the user sits down normally

User Input: The user should sit down on a chair in a "normal" fashion
Expected Result: The device should not detect a fall and the status indicator remains green

Device Test 7: Testing for false alarms when abruptly sitting down
User Input: The user should abruptly sit down on a chair
Expected Result: The device should not detect a fall and the status indicator remains green

Device Test 8: Cancel button and status indicator
User Input: The user should fall to activate the buzzer and then press cancel within 30 seconds
Expected Result: After pressing the cancel button, the buzzer should stop and the status indicator should turn from yellow to green

Device Test 9: Help button and status indicator
User Input: The user should press the help button
Expected Result: After pressing the help button, a high frequency buzz should start and the status indicator should turn from green to red

Device Test 10: On/off switch and status indicator
User Input: The user should turn the device on, wait a minute and then turn the turn off
Expected Result: After turning the device on, the status indicator should turn green. When the user turns the device off, the status indicator light should turn off.

Device Test 11: Structural integrity of the case
User Input: The user should wear FEDS properly and fall on it
Expected Result: The case should be able to withstand the fall and not break

Device Test 12: Connecting to Wi-Fi
User Input: The user should connect a HDMI cable to a monitor and keyboard and mouse to the device. The user should go to the Desktop and find their Wi-Fi signal and connect to it. The Wi-Fi signal should have internet connection and not only have local area connection (LAN).
Expected Result: The device should connect to the Wi-Fi signal and be able to send data to the Server (which will be tested in the next section). It is also possible to test if there is internet connection by going on the web browser on the device.

Server/Operator Application Testing

This section will focus on a few key ideas related to the server/operator application section:

1. FEDS device should be able to send data to 4Life's database
2. Operator application should be able to pull data from the database and view the status of the devices
3. Operator application should be able to add, edit and remove devices

In these tests, the operator interface must be opened.

Server Test 1: Sending FEDS info to database and view on operator interface

    User Input: The user should fall forward and not press the cancel button for 30 seconds after the low frequency buzzer starts buzzing

    Expected Result: After 30 seconds, the device should send an emergency signal to the database. This emergency signal should be shown in the operator interface

Server Test 2: Sending the GPS Location

    User Input: The user should fall forward and not press the cancel button for 30 seconds after the low frequency buzzer starts buzzing.

    Expected Result: After 30 seconds, the device should send a location signal to the database. This location should be shown in the operator interface.

Server Test 3: Add contact in operator interface

    User Input: The operator should use the add contact function and fill out the necessary data (Device ID, Client Name, Contact Number, Medical History and Emergency Number).

    Expected Results: After adding the contact, the data should be pushed to the database and the operator application page will be refreshed to show the changes.

Server Test 4: Edit contact in operator interface

    User Input: The operator should use the edit contact function and change the necessary data (Device ID, Client Name, Contact Number, Medical History and Emergency Number).

    Expected Results: After editing the contact information, the data should be pushed to the database and the operator application page will be refreshed to show the changes.

Server Test 5: Remove contact in operator interface

    User Input: The operator should use the remove contact function on a contact.

    Expected Results: After removing the contact, the removed contact should be deleted from the database and the operator application page will be refreshed to show that the user is deleted.