1 SYSTEM TEST PLAN

1.1 UNIT TESTING

1.1.1 Microcontroller

The MSP432 MCU needs to be able to receive transit data from the server (via the GSM), as well as send commands and data to our display. In order to confirm that sending and receiving works correctly we will perform separate tests.

MCU Test 1: Receive and store data
Run a program that, after receiving data from the server through the GSM module, will take the data from the UART and store into RAM. The program will print sample data from memory to screen to check functionality.
Outcome: Sample data from the server will be printed onto the console successfully from the microcontroller from RAM.

MCU Test 2: Send command and data to display
Run a program that will send an upload image command from our MCU to the display with data stored on RAM previously.
Outcome: The display will update with a new image correctly shown.

The Microcontroller will be integrated with the GSM Module, Display, Sensors, and battery unit to ensure a full functioning system.

1.1.2 GSM Module

The Sim 800C GSM Shield needs to be able to send requests (AT commands) to connect with the server and get data to the MCU. In order to confirm the functionality of the GSM we will perform a test.

GSM Test: Send AT Commands
Send commands through a serial console to connect to the server and read data back to the console.
Outcome: The serial console will show correct data from the server on the console.

1.1.3 Display

The MPico Pervasive Display needs to be able to display a stable image, and update when refreshed.

Display Test 1: Display image
Upload image to the display through the SPI port
Outcome: Display controller will return the correct response code (0x9000).

Display Test 2: Refresh display
Send a refresh command to the display, with a new image on the display.
Outcome: Display will correctly refresh the screen, showing the new image.

1.1.4 Proximity Sensor

Proximity Sensor Test 1: Sense nearby people within maximum radius
Detect a nearby person within 6 meters radius facing forward of the system.
Outcome: Sensor correctly detects nearby person

Proximity Sensor Test 2: Sense nearby people outside maximum radius
Detect a nearby person outside 6 meters radius facing forward of the system.
Outcome: Sensor will not detect the person.

1.1.5 Light Sensor

Light Sensor Test: Sense when daylight is not available
During the night time, detect when there is not enough daylight to view the display
Outcome: Sensor correctly detects the lack of light and switches on the LED.

1.1.6 Battery Unit and Solar Panel

Battery Unit and Solar Panel Test 1: Battery lifetime
Without recharging, run fully charged battery (connect to all components) until battery depletes
Outcome: Battery will die after 144 hours.

Battery Unit and Solar Panel Test 2: Charging capability
During discharge (or operation of system), charge battery from solar energy through the panels.
Outcome: Battery will charge until full (within 7 hours) or until the absence of solar energy.

1.2 SYSTEM INTEGRATION TESTING

In order to integrate the separate components of our device efficiently and effectively, we will integrate components in a specific manner. Since the testing will involve having to integrate certain components, we will integrate them as necessary to perform the tests. These integrations will be tested according tests explained below.

1.2.1 Integration Test 1

As explained in the functional specification the overall proof of concept of Solarity is to a stable transit information image displayed on the pervasive display. The first integration test starts with a request from the GSM module and after the whole process is complete the microcontroller will transmit the image data to the display,

which will showcase the bus time arrival as an image. The required steps to satisfy the completion of this integration test are labeled in order as follows:

1. The Solarity unit will be properly plugged up and powered.
2. The Solarity unit (the unit will have a unique identification number) be placed at a designated transit station/stop in Metro Vancouver. The IT technician will input the unit's unique identification number to the server and match it with the designated transit station. The server will now be able to provide the Solarity unit with the applicable transit information.
3. The microcontroller will program the GSM with AT commands.

The steps above only occur during the initial setup of Solarity. The following steps (4-13) get repeated in order every 2 minutes and continue as long as respective Solarity device is in operation:

4. The GSM Module will send a request to the Solarity server via the 3G network in the form of GPRS data.
5. The server will send a request to the TransLink server to retrieve the bus information for the designated stop.
6. The server will convert the data received into an image that will be displayed.
7. The server will break the image into a byte-stream, where the bytes represent the ordered pixels that make up the image.
8. The bytes will be transformed into the EPD format, which is the format that the pervasive display controller can read.
9. The server will send the data to the GSM module.
10. The GSM module will transmit the bytes received to the microcontroller.
11. The microcontroller will transfer the bits to the pervasive display.
12. The pervasive display will produce the image on its screen.

To test the sensors:

13. The 2 sensors (proximity and light respectively) of the system have to be tested to ensure that the integrated system is working accurately:
    A. For the proximity sensor, we have to keep the system idle (in other words, have no one walk within 6m radius in front of it) for a given amount of time so that the display goes stops refreshing when not needed. Then let a person walk within the 6 meters radius facing forward of the system. The sensor will correctly detect the nearby person and switch the screen from idle mode and refresh the image.
    B. For the light sensor, we have to test the system during the night time to detect when there is not enough daylight to view the display. The sensor will correctly detect the lack of light and switches on the backlight LED on.

1.2.2 Integration Test 2

The first integration test covered the operation of Solarity without integrating the solar panel. The integration of the solar panel with the integrated system depicted in the previous integration test completes the Solarity unit. The system connected to solar panels will be tested according to steps 1 through 13 in the previous section. The solar panel in particular will be tested in parallel to steps 1 through 13. During discharge (or operation of system), the battery will charge from solar energy through the panels. The battery will charge until full (within 7 hours) or until the absence of solar energy. After this test, the fully integrated system will have been fully tested.