# LOCATION FINDER

## Design Document


### Abstract

This document overviews the design and pseudocode considerations used when building the Location Finder application.



Justin Cervantes

Cervantes.jfa@gmail.com

# Contents

## Overview

This document outlines the design and pseudocode used to create the Location Finder application.

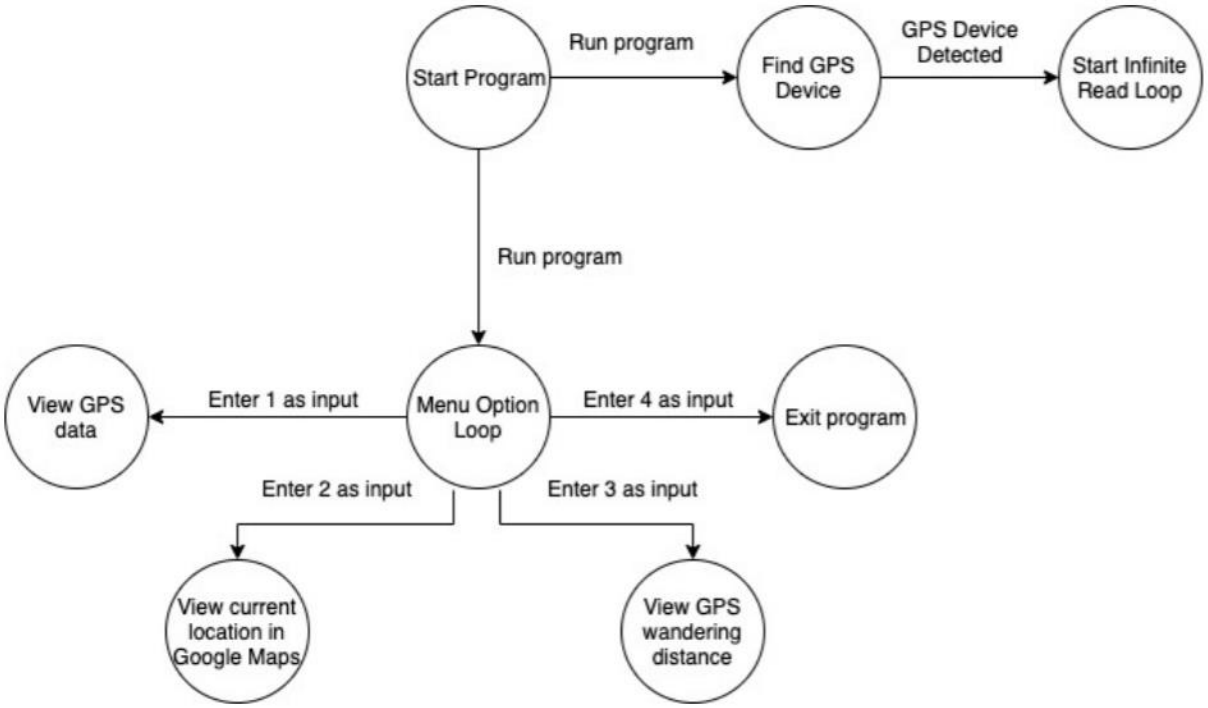
The Location Finder application is implemented with the following features:

- Menu Option Screen
- Viewing GPS satellite data and user coordinates
- Viewing current location in Google Maps
- Viewing total wandering distance

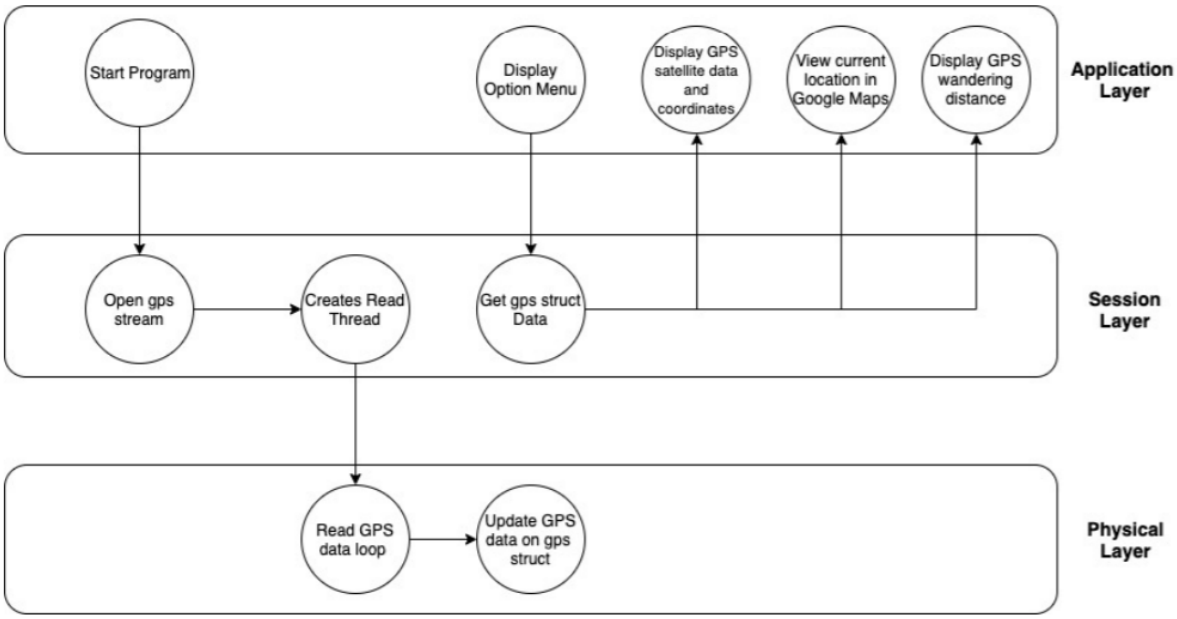
## Design

The following section details the diagrams we've created to implement the RFID Terminal Emulator program. We've produced a State Diagram and an Architecture Diagram during the design process.

### State Diagram



## State Implementation Diagram




# Pseudocode

The application is implemented based on the following pseudocode.

***dcgps.c***

```
main( )
{
        Malloc gps_data_t structure
        Open stream to gpsd
        Start stream

        Create read thread
        Create menu options loop

        Close thread
        Close stream
        Free the memory allocated by malloc
}
```

***gps-utils.c***

```
myThreadFun( )
{
        Infinite loop {
                Get all satellites
                If (gps satellites are in view) {
                        Read the gps satellites' data

                        If (a GPS fix is available) {
                                Update gps fix availability flag to true
                                Track total meters wandered
                        }
                }
        }
}

openBrowserToCurrentLoc( )
{
        Loop until gps fix has been found
        If gps fix has been found {
                Open browser window to google maps
        }

}

printGPSData( ) {
```

```
        Call print function for printing gps satellite data and coordinates
    }
    getCurrentGPSWanderingDistance( ) {
        Print total distance wandered
    }
```

***gpsprint.c***

```
    printResults( ) {
        if (satellites are visible) {
            Print number of satellites
            Get the list of satellites
            Print each satellite's PRN, Elevation, Azimuth, and Used Flag
            Print timestamp
            If (GPS fix is available) {
                Print GPS coordinates
            }
        }
}
```