**FisheriesNET: A Complete User Manual & Guide**

**User Manual:**

**Tracker(s) Setup:**

Lithium Batteries must be placed properly into battery holders. It is EXTREMELY important that the positive side of all batteries be on the left side with respect to the tracker or the side closest to the Printed Circuit Board (PCB).

**Start Up:**

Switch on the trackers, the switch should be moved to the right for it to be on. If working properly, the Green LED will flash periodically.

Once GPS signal has been acquired, the Red LED will flash periodically and the Greed LED will turn off.

Once Red LED is flashing, unless preset, take tracker(s) to area where the base station will be placed. Hold switch at the bottom of PCB until there is a solid Green LED flash. The home base station coordinates have been set.

The trackers are now ready to be deployed.

**Base Station Setup:**

Make sure the Xbee module is connect properly to the USB on the Raspberry Pi.

Plug in power cable to Raspberry Pi and a power outlet.

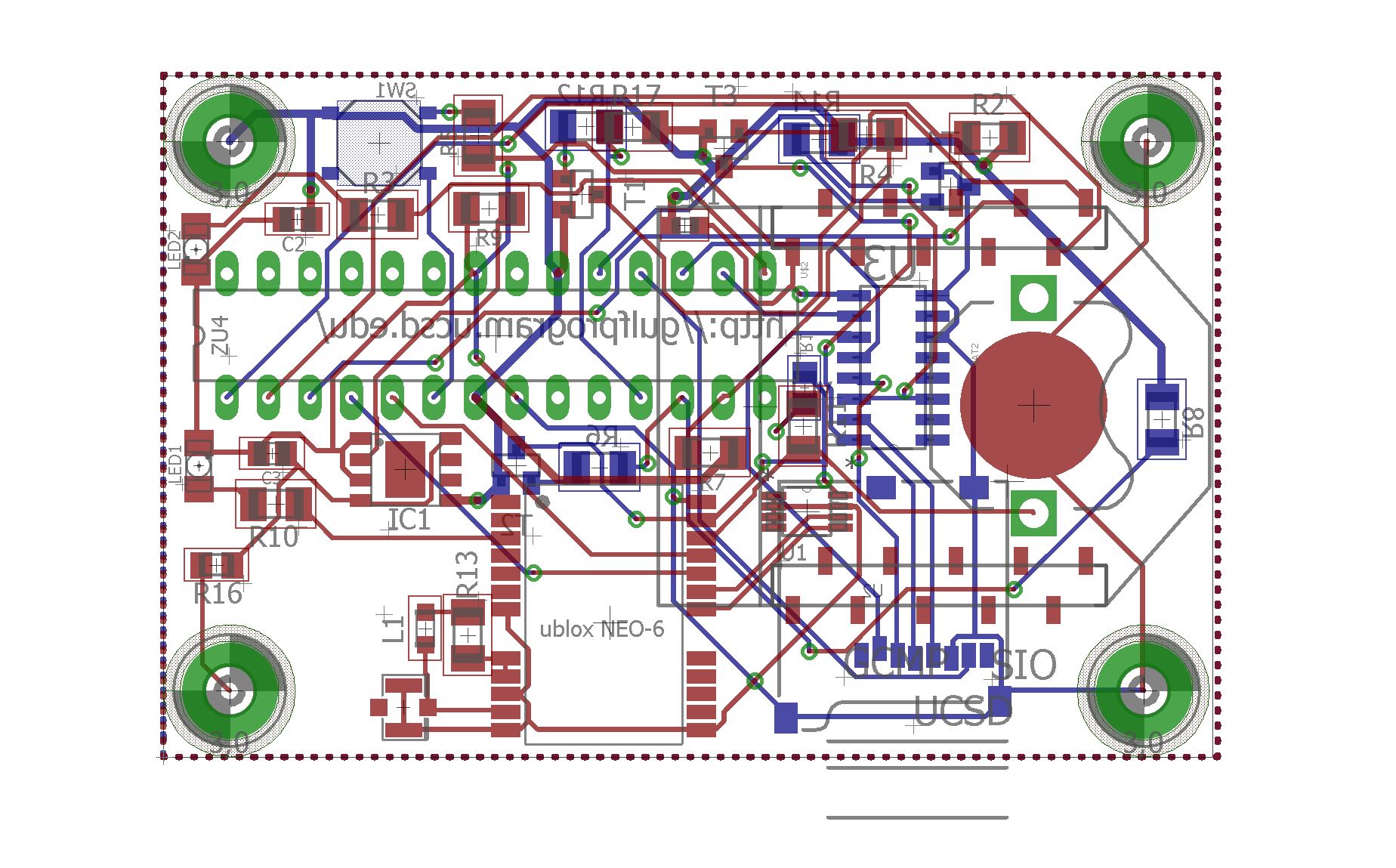
The Base Station should now be setup.

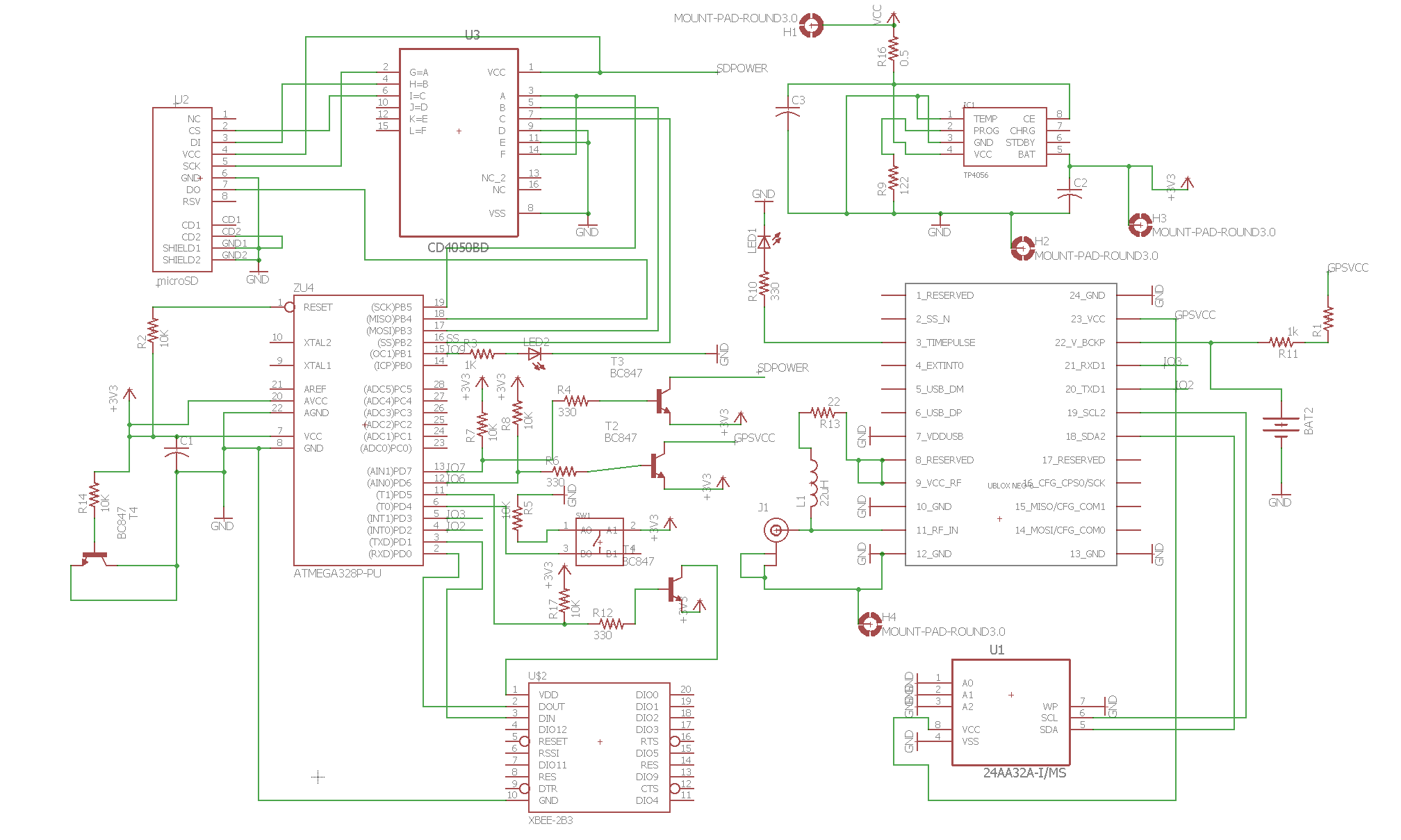
**Retrieving Data:**

Retrieving Data should be as simple as removing the Ultra SD from the Raspberry Pi and plugging it in to a computer.

If you see the data has not properly been loaded to the Ultra SD, retrieve trackers and remove microSD’s manually from each.

**Schematics**





**Technical Details and Guide:**

**Trackers**

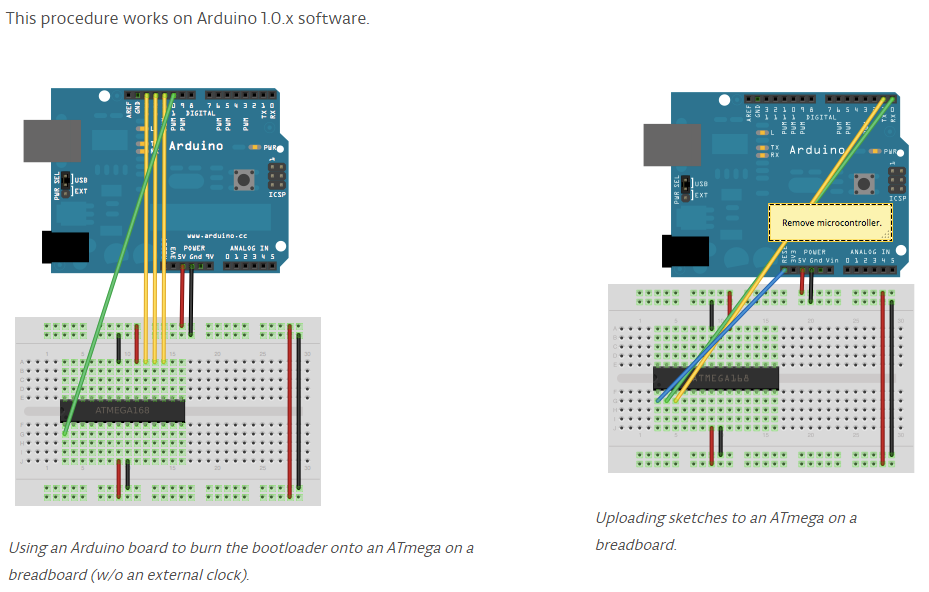
|  |  |
| --- | --- |
| PCB Part | Actual Part |
| ZU4 | ATMEGA328P-PU (Micro-controller) |

**Functional Description:**

*The ATMEGA328P-PU is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega328-PU achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.*

**Application:**

When designing, prototyping platforms are used to test a system. In our case we used an Arduino Board to test the design. After having a working design, it is ideal to migrate to a smaller platform, like the ATMEGA328P-PU Microcontroller. Initially a bootloader is burned to the microcontroller so that it functions as a standalone microcontroller.



After you burn the bootloader, you can now upload sketches or the program onto the microcontroller using the Arduino interface. You can find the current program on GitHub here: <http://bit.ly/2c6fSwk>

The microcontroller directly controls and interacts with the Xbee, the home base switch sw1, LED2 (green), and the HEX Buffer/Converter CD4050BD. It’s is the computer of the trackers.

|  |  |
| --- | --- |
| PCB Part | Actual Part |
| U$2 | Xbee |

**Functional Description:**

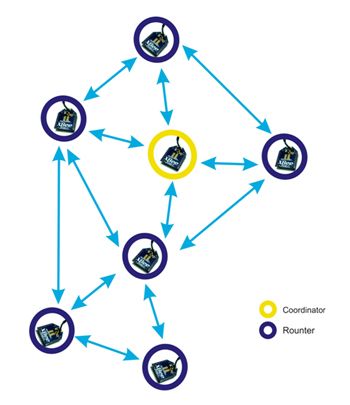
*XBee and XBee-PRO ZigBee RF modules provide cost effective wireless connectivity to electronic devices. They are interoperable with other ZigBee PRO feature set devices, including devices from other vendors.*

**Application:**

Our Xbees are programmed using the XCTU Program Interface. For tutorials, see here: <http://bit.ly/2bQjjJG> After adding the radio module, parameters are set. More details here: <http://bit.ly/2c6lAhs>

Currently our Xbees are in AT Mode (“Transparent” mode) in which only the end point devices can communicate to a single coordinator and not the other way around. In the future, it would be more ideal for the coordinator to communicate back or End Devices be able to communicate between each other to prevent wireless communication issues.

**Now**  **Ideal**

|  |  |
| --- | --- |
| PCB Part | Actual Part |
| U3 | CD4050BD (Buffers/line drivers) |

**Functional Description:**

*The CD4049UB and CD4050B devices are inverting and non-inverting hex buffers, respectively, and feature logic level conversion using only one supply voltage (VCC). The input-signal high level (VIH) can exceed the VCC supply voltage when these devices are used for logic-level conversions.*

**Application:**

The CD4050BD is directly connected to the ATMEGA Microcontroller and is a in between connection to the microSD where all the tracker GPS data is written to a .csv file. It serves as an amplifier and converter.

|  |  |
| --- | --- |
| PCB Part | Actual Part |
| EXT | microSD card |

**Functional Description:**

*MicroSD is a type of removable flash memory card used for storing information. SD is an abbreviation of Secure Digital, and microSD cards are sometimes referred to as µSD or uSD .*

**Application:**

Two types of files are stored here for our purposes: .txt and .csv files. A “settings.txt” file is pre-stored with default Latitude and Longitude coordinates, SD\_boat tracker name, and SD\_trip to mark the current trip. The home base switch (SW1) can be used to update the Latitude and Longitude Coordinates. A different tracker name must be set for each tracker.

|  |  |
| --- | --- |
| PCB Part | Actual Part |
| GPS | NEO 6m GPS |

**Functional Description:**

*The NEO-6 module series is a family of stand-alone GPS receivers featuring the high performance u-blox 6 positioning engine. These flexible and cost effective receivers offer numerous connectivity options in a miniature 16 x 12.2 x 2.4 mm package. Their compact architecture and power and memory options make NEO-6 modules ideal for battery operated mobile devices with very strict cost and space constraints.*

**Application:**

In our case, our GPS module receives the GPS input signal from the antenna through pin RF-IN. It transmits and receives Data and Clock with the EEPROM Microchip. LED1 (Red) is also controlled by the TIMEPULSE pin.

|  |  |
| --- | --- |
| PCB Part | Actual Part |
| U1 | 24AA32A-I/MS ( Microchip Serial EEPROM) |

**Functional Description:**

*The 24XX32A supports a bidirectional, 2-wire bus and data transmission protocol. A device that sends data onto the bus is defined as transmitter, while a device receiving data is defined as a receiver. The bus has to be controlled by a master device which generates the serial clock (SCL), controls the bus access and generates the Start and Stop conditions, while the 24XX32A works as slave. Both master and slave can operate as transmitter or receiver, but the master device determines which mode is activated.*

**Application:**

In our case, the UBOX NEO-6 GPS Module controls the sending and receiving of data by this device through pins SCL2 I/O DDC Clock and SDA2 I/O DDC Data.

|  |  |
| --- | --- |
| PCB Part | Actual Part |
| IC1 | tp4056 (Chips) |

**Functional Description:**

*The TP4056 is a complete constant-current/constant-voltage linear charger for single cell lithium-ion batteries.*

|  |  |
| --- | --- |
| PCB Part | Actual Part |
| T1, T2, T3 | transistors |

**Functional Description:**

*Supertex’s vertical DMOS FETs are ideally suited to a wide range of switching and amplifying applications where very low threshold voltage, high breakdown voltage, high input impedance, low input capacitance, and fast switching speeds are desired.*

**Application:**

► Logic level interfaces – ideal for TTL and CMOS ► Solid state relays ► Analog switches ► Power management

|  |  |
| --- | --- |
| PCB Part | Actual Part |
| L1 | 22uH (Inductor) |

**Functional Description:**

*Inductors used to resist changes in electrical current.*

|  |  |
| --- | --- |
| PCB Part | Actual Part |
| C1, C2, C3 | 100Nf,10uF |

**Functional Description:**

*Capacitors are used to store charge and energy.*

|  |  |
| --- | --- |
| PCB Part | Actual Part |
| R1-R17 | Varies Values |

**Functional Description:**

*Resistors resist the flow of electrons in circuits, resulting in changes in voltage levels. Voltage can be regulated through resistors.*

**Base Station:**

|  |  |
| --- | --- |
| Part | Raspberry Pi |

**Functional Description:**

*The Raspberry Pi is a small, barebones computer developed by The Raspberry Pi Foundation, a UK charity, with the intention of providing low-cost computers and free software to students.*

**Application:**

Raspberry Pi 3 is being used to power up and control the Xbee coordinator that receives information from the Xbees on the trackers wirelessly. Data is stored into a removable Ultra SD, which is then accessible locally. A local host will connect in the future to the FisheriesNET website, where it will update the Data folder when the Raspberry has received new Data.

Currently, Ubuntu is the OS that is being used by the Raspberry Pi 3 and can be found here: <http://bit.ly/2ccYTvA>

A program like Windows 32 Disk Imager can be used to write the OS to a microSD, and then inserted to the Raspberry Pi. This program can be downloaded from here: <http://bit.ly/2bYw5pS>

The Raspberry Pi is programmed through shell scripts to allow for the automatic execution of commands upon powered up.

Code and Details can be found on GitHub here: <http://bit.ly/2bYxsVI>

**Extras:**

**MATLAB**

The Local MATLAB based software can be found here: <http://bit.ly/2bQujGU>

It allows for visualization of trips and the generation of statistics and databases.

**Web Interface:**

The online site for tracking can be found here:<http://bit.ly/2cd0KQN>

Source code will be found here: <http://bit.ly/2cd1Lbw>