Tracuz BLE Communication Protocol

**Phone to bag:**

1. Predefined Animation/Frame Select:

**Function**: To select a predefined animation/frame to display on the bag. Animation\_id is for showing which animation ID is selected, each animation has one number. Display\_mode has two things 1 for making the animation scrolling and 0 for static.

**JSON Format**:

{

command\_id: 0,

animation\_id: (Integer) (0 - 50)

display\_mode:(1- for scrolling, 0- static)

}

2. LED Custom Frames Transfer:

**Function**: To allow users to send the frames which they have made using the editor to the bag.

Method 1: LED Frames Sending

**JSON Format**:

{

command\_id: 1,

led\_frames: (hex string array)(eg: [‘0x1111111111111111’, ‘0x2222222222222222’ ...]),

intensities: (Integer Array) (This is for setting the intensity of each frame how bright the frame should look),

led\_ring\_str: (hex string array) (eg: [‘0xFFFFFF’, ‘0x000000’, …]),

display\_mode: (Integer) (0 - 1), (Display\_mode has two things 1 for making the animation scrolling and 0 for static.)

frame\_delay: (Integer Array) (This parameter tells after what time each frame should appear in other words it tells what should be the time delay between each frame in an annimation),

num\_frames: (Integer) (Number of frames in the animation)

}

Method 2: Text Sending

**JSON Format:**

{

command\_id: 2,

text: (String),

num\_chars: (Integer),

intensity: (Integer) (0 - 15),

frame\_delay: (Integer) (0 - 5),

led\_ring\_str: (Hex String) (eg: ‘0xFFFFFF’)

}

3. Mode Select:

**Function:** To Select the mode from the app.

**JSON Format:**

{

command\_id: 3,

mode: (Integer) (0 - 1)

}

**Modes:**

(0 - Navigation Mode OFF)

(1 - Navigation Mode ON)

4. Inner Illumination Control:

**Function:** To control the inner light in the bag

**JSON Format:**

{

command\_id : 4,

inner\_illumination: (Integer) (0 - 1) (0 - OFF, 1 - ON)

}

5. Secure Mode Control:

**Function:** To turn on/off the secure mode in the bag.

**JSON Format:**

{

command\_id: 5,

secure\_mode: (Integer) (0 - 1)(0 - OFF, 1 - ON)

}

6. Deep Sleep Control:

**Function:** To turn on/off the deep sleep mode in the bag.

**JSON Format:**

{

command\_id: 6,

State: (Integer) (0 - 1)(0 - OFF, 1 - ON)

}

7. BLE Beacons Address Update

**Function:** To send all the BLE Addresses of the beacons **to be tracked**. Send this **everytime** there is an update in the devices to be tracked. (Add/Enable/Disable)

**JSON Format:**

{

command\_id: 7,

addresses: (String array) (eg: [‘yy:yy:yy:yy:yy:yy’, ‘xx:xx:xx:xx:xx:xx’, …]),

num\_beacons: (Integer)

}

8. Update Settings:

**Function:** To Update settings to the bag.

**JSON Format:**

{

command\_id: 8,

direction\_led\_color: (Hex String) (eg: ‘0xFFFFFF’),

missingitem\_notif: (Integer array [3]), (eg: [0, 1, 1]), (Tells whether missing item notif should be enabled in vibration or ringing)

secure\_notif: (Integer array [3]), (eg: [0, 1, 0]), (Tells whether secure notif should be enabled in vibration or ringing)

power\_notif: (Integer array [3]), (eg: [1, 1, 1]), (Tells whether power\_notif should be enabled in vibration or ringing)

vibration\_intensity: (int) (0 - 255),

ble\_scan\_interval: (Integer), (Beacon scanning interval),

schedule\_time: (String array), ([schedule\_start\_time, schedule\_end\_time]),(There is a feature in app to set switch On time and switch off time which will make the bag ON and OFF automatically like our mobile phone)

}

9. Power Settings and Manual Charge:

**Function:** To update the power settings to the bag.

**JSON Format:**

{

command\_id: 9,

charge\_mode\_doc: (Integer) (0 - 1),

charge\_capacity\_doc: (Integer)

}

10. Time Update:

**Function:** Periodically sending current time to the bag.

**JSON Format:**

{

command\_id: 10,

time: (String) (When the app gets connected to the bag each time the time will be sent to the bag)

}

16. Navigation Update:

**Function:** Sending navigation update to bag.

**JSON Format:**

{

command\_id: 16,

navigation: (int) (0-straight, 1-left, 2-right, 3- U-turn, 4-Slight left, 5-Slight right)

}

**Straight, continue straight, round about continue straight-** Both the straps vibrates

**Left, circle left-** Left strap will vibrate continuously without any delay

**Right, circle right-** Right strap will vibrate continuously without any delay

**U-turn-** vibrate both straps with some delay

**Slight left lane, keep left-** vibrate left strap alone with some delay

**Slight Right lane, keep right-** vibrate right strap with some delay

21. Bag Status:

**Function:** Getting Bag Status.

**JSON Format:**

{

command\_id: 21,

All\_Update:1(Get Status of All) (Update status will provide you the status of inner illumination, secure mode, deep sleep, animation ID)

}

**Bag Side:**

1. Missing Item Notification

**Function:** To alert phone in case an item is not found.

**JSON Format**

{

c: 11,

d: 1; (1 - Device missing Flag)

}

2. Secure Mode Disruption

**Function:** To alert phone in case bag is not secure while in secure mode.

**JSON Format**

{

c: 12,

z: (Integer) (0 - 1)(Zip alert)

m: (Integer) (0 - 1)(Motion detected)(This is when the bag is in secure mode and not in motion and all of a sudden someone tries to lift your bag the bag owner will be alerted)

}

3. System state

**Function:** To update the phone with system state after any update occurs.

**JSON Format**

{

c: 13,

s: (Integer) (0 - 1),(Secure Mode ON OFF)

m: (Integer) (0 - 1),(Mode)(0 - Navigation Mode OFF, 1 - Navigation Mode ON) (0 will represent secure mode).

}

**JSON Format**

{

c: 18,

i: (Integer) (0 - 1),(Inner Illumination)

}

**JSON Format**

{

c: 22,

d: (Integer) (0 - 1),(Deep Sleep)

}

**JSON Format**

{

c: 19,

A: (Integer) (0 - 50)(Animation Id)

}

4. Pollution Level

**Function:** To periodically update the phone with gas sensor values

**JSON Format:**

{

c: 14,

d: (float32)

}

5. Charge Level

**Function:** To periodically update the phone with charge dissipated level

Discharge data:

**JSON Format:**

{

c: 15,

p: (Integer) (0 - 1)()

}

**JSON Format:**

{

c: 20,

d: (float32) [current capacity]

}

Charge Data:

{

c: 17,

ch: (float32)

p: (Integer) (0 - 1)

}

Animation Indexes:

GIF Indexes:

0 - coffee battery

1 - univ

2 - star blink

3 - compass

4 - gps

5 - phone charge

6 - pingpong

7 - plane

8 - heart

9 - weather lightning

10 - running

11 - zip

12 - lock

13 - music

14 - yo

15 - radar

16 - smiley

17 - sunrise

18 - left arrow

19 - right arrow

20 - up arrow

21 - logo