9. System Test Plan

Our system is safety oriented since any errors or confusions regarding the signals can cause serious accidents as both the helmet user and other road users depend on accurate functionality of the system. In order to ensure high performance of the Smart Helmet and for easier debugging process, unit testing will be conducted, as well as regression testing after adding any modules to the system. For instance, after the first stage of the development, thorough testing will be conducted to ensure that the implemented features work perfectly as expected. Indeed, the triggering circuit should be able to send correct signals and the microcontroller should be able to decode the signal and switch on the correct LED section.

After the second stage of development, the whole system should be working without interference or degradation of the previously tested modules. It is worth noting that not all the unit tests will be executable after integration.

9.1 Unit Testing

Unit testing will be conducted on individual components separately as follows:

9.1.1 Triggering Unit Test Plan

- **Signal Buttons**
  - Use two LEDs connected to the switch to confirm correct functionality of the signal switch:
    - When left button is pressed, the left LED will turn on
    - When right button is pressed, the right LED will turn on
    - The top buttons when pressed will turn both LEDs off.
- **RF Transmitter**
  - Use the RF receiver and four LEDs. Each LED is turned on by one of the four combinations of sent signals.
- **Integrated Triggering Unit**
  - Once the individual top components are working as described, the whole triggering circuit is assembled. Turn signals should be successfully sent via the RF transmitter and the corresponding LED should turn on.
- **Brake Detection:** The accelerometer will be used to detect braking event. We will manually simulate the movement of the accelerometer when assembled to the helmet. When a brake is detected, turn an LED on and display the data to the serial monitor of the arduino.

9.1.2 LED Unit Test Plan

To test the LEDs, a software test framework should be written to simulate the triggering and choose different colors and different patterns of the LED strip.

When the above is tested, the LED and the triggering should be integrated and tested to obtain the first working stage of the proof-of-concept.

9.1.3 Microcontroller Unit Test Plan

To test the microcontroller unit and explore its full capability, sample projects will be run first to get used to the programming style and the connections of the unit. In addition, the microcontroller should be integrated with the LEDs and the triggering circuit to make sure that decoding is correctly done and that the microcontroller unit is capable of driving the LED unit.

9.1.4 Speaker and Microphone Unit Test Plan

Testing the speaker and microphone unit can be done separately by pairing it with a Bluetooth capable audio device and making sure that audio can be received and transmitted without issues.

9.1.5 Fall Notification Unit Test Plan

This is to test that the accelerometer combined with the algorithm can detect free fall and send appropriate signal to MCU. To be able to test this unit we have to make sure that the unit is shock resistant and will not be damaged by testing. After proper packaging we will drop the accelerometer and check that the algorithm detects the free fall.

9.2 First Phase Regression Testing

This is to test correct functioning after integration of different units. We divided our development to two stages as explained in the functional specification document and will carry on testing in two stages accordingly. The product of 'development stage I'- the proof of concept performing our main system functionality as explained in the functional specification document- would be required to detect correct signal inputs through the triggering system, wirelessly transmit the input to microcontroller unit on the helmet, decode inputs and finally produce corresponding output by showing signals through LEDs.

Once the unit testing on all components is carried out, we will integrate units to ensure correct data transfer between them and ensure that the functionality of the product as a whole is as expected. To achieve better debugging, we will start by integrating small numbers of units and then putting the whole system together.

We will connect the Triggering circuit to the MCU to ensure correct transfer of inputs to MCU through the RF module. To test the output produced according to the input received by MCU, we will use three single LEDs indicating possible outputs: right signal; left signal; and stop signalling. Then we will add the LED unit to it to ensure that the output from MCU drives LED unit accordingly. At the end we will put all the systems in the helmet and test the whole system by giving all possible combinations of signal inputs through triggering unit and checking the resulted output for each. In this part of testing, the power unit is being tested to power all different units.

9.2 Second Phase Regression Testing

The second phase of product is after adding extra features to the first phase. The features we wish to add to the main functionality are: fall notification and Bluetooth speaker and microphone. Similar to phase one, after unit testing we will start adding one unit at a time to make debugging easier.

In this part of testing we will integrate the accelerometer with microcontroller unit to test that free fall can be detected and correct output is produced after detection. Then we will add the LED unit and test that free fall would result in flashing of the brake signal. Then we will put all the units together, adding the Bluetooth speaker and put them in the helmet. Then we will test all possible combinations of inputs from triggering unit and check their results, we will test free fall of the helmet in addition to testing appropriate functioning of the Bluetooth speaker. The final stage would be installing the system on the bike to test usability in addition to correct functionality.