**Project Summary (Group 11)**

**Internet of Things (IoT) in Unmanned Ground Vehicle**

**Objectives:** An unmanned ground vehicle is one which functions while in touch with the ground but without the presence of a human on board. Unmanned ground vehicles (UGV) can be employed in a variety of situations when having a human operator present is inconvenient, dangerous, or impossible. The project focuses on the following objectives:

* Remotely controlled UGV over the Internet
* Live stream from the UGV
* Attain serialization between Raspberry Pi and Ardiuno
* Store and analyze sensor data
* Semi automate the UGV
* Create a web application to control the UGV

**Requirements:** *Hardware*: Raspberry Pi, Arduino Uno, Camera, Gyroscope, Breadboard, Jumper Wires (MF, MM and FF), Multi-meter, Display Screen, HDMI Cord, Micro USB, Keyboard, Mouse, DC Motor, GPS, Sensor, Ultrasonic Sensor, Servo Motor, Density, Humidity and Temperature sensor, Air Quality sensor etc. *Software* : Arduino IDE, Python, Google Firebase, HTML, CSS, Javascript, C++

**Implementation:** The rover consists of two processing systems: a Raspberry Pi single board computer and an Arduino UNO micro- controller board. The Raspberry Pi serves as the eye of the system because it consists of a Pi camera that broadcasts a live stream around the rover for users to navigate. The Arduino acts as the body of the system because it controls the movement of the four motors that rotate the wheels of the rover.

To achieve remote access and control of the rover over the Internet, we needed to connect to a cloud database. Therefore, the Raspberry Pi is connected to the Google Firebase cloud server. When the user clicks a navigation button in the web app to move the rover in a specific direction, the navigation variable is set to 1. Updating the value of the navigation variable does the same thing in the cloud database in real time. The transfer command is then read by the Raspberry Pi on the rover and immediately sends the serialized command to the Arduino micro-controller responsible for controlling the motors on the four wheels of the rover. Upon receiving the serialized command, the Arduino will execute the forward code on the hobby motor and the required motor on the rover. Some sensors connect to the Arduino microcontroller while others connect to the Raspberry Pi. The data that is generated by the sensors is consistently displayed in the web application. The data generated by the sensor connected to the Arduino is first sent to the Raspberry Pi in serialized format. Once all the sensor data is stored on the Raspberry Pi, the data will be uploaded to Google Firebase’s cloud storage and then the data will be successfully transferred to your website.

When connected to a local WiFi network, the rover responds fairly instantly. The response time increases dramatically when utilizing the Internet to operate the rover through a cloud server with a public IP address. Acquisition of video for live streaming content can only be done on local networks. Live streaming is not possible over the WiFi network.





