Vedo Systems Lunar Bot Challenge Description:

**Summary**:  
Challengers will be developing an interface for ground teams to control remote rovers on the moon for science and exploration. Users require the ability to visualize 3D environmental data, interact with 3D environmental objects (click an object to see details), view system telemetry, and modify system settings and controls on the fly. As the rover has autonomous capabilities, users need to place target markers in order to have the rover plan out a path.

**Prize**: $500.00

**Requirements**:

* Interface must be cross platform (in the case of a client server design, only the client must be cross platform).
* Interface must consume incoming 3D data from the rover and display it for user interaction
* Interface must support panning of 3D environment for user inspection
* Interface must support a camera view of the rover (this will simply show the 3D environment from the perspective of the rover and not actual camera footage).
* Interface must provide common controls for robotic systems, i.e. pan tilt of camera or movement in 3D space. Use your imagination for what you think would be useful.
* Interface must provide capability for adding target markers to the environment.
* Interface must provide a button to initiate route planing.
* Interface must provide a button to start executing a given rout/plan.
* Interface must provide an emergency stop button.
* Interface must consume data from the rover. Data can either be read from a JSON file, or over the network via a ROS2 topic. (ROS2 is preferred)
* Interface must send data to the rover. Data can either be written to a JSON file, or published over the ROS2 network. (ROS2 is preferred) Data must include...  
  - Target marker positions.  
  - Control system data (i.e. orientation of camera based on user inputs or other data related to user inputs)  
  - Action data (i.e. initiate route planning, start, route, stop)

**Provided Resources:**

* Environmental terrain file. (.stl, challengers are free to modify file format as desired)
* 3D environmental objects (.obj and .stl, challengers are free to modify file format as desired)
* Sample ROS2 node for emulating communication with rover.
* Rover telemetry example (.json format)
* Rover environmental data file (.json format)
* Python script to emulate communication with rover (if ROS2 is not used).
* Resource links
* Documentation on input and output data protocols.