D. High-Level Requirements List

- **Omni-wheel Drivetrain**: The primary requisite pertains to the ability of the omni-wheel drivetrain to maintain heading according to a directional vector provided by our localization subsystems (UWB & Computer Vision). This system will receive a the direction vector in the form of an array with broken down direction and speeds for each motor. Consequently, upon determination of a directional vector by our software, our robot is expected to initiate movement in the designated direction within 500 milliseconds.
- **UWB Localization**: The subsequent criterion concerns the ability of the UWB localization system to produce precise positional data, with an accuracy threshold within 1 meter. We aim for continuous real-time updates regarding the child's whereabouts in relation to the robot.
- **Close-range Object Detection**: The third requirement entails the implementation of close-range object detection mechanisms to discern obstacles obstructing the robot's trajectory. It is expected to promptly notify users of impending collisions, thereby safeguarding the child's well-being and facilitating timely intervention for obstacle removal.

Drivetrain Subsystem R&V

Requirements	Verification
The electronic speed controllers power the motors at the command of the board control subsystem. The commands may be to set duty cycle, set current, or set RPM.	Ensure the OWB is at an idle state such that the wheels are not spinning. Then, send a command to both ESCs from the board microcontroller to either set a nonzero duty cycle, current, or RPM. Confirm that both motors begin spinning.
The drivetrain must be able to maintain a speed of at least 3 MPH (that fastest speed we are comfortable following a user), with a fully loaded oxygen tank (Full tank at 3.458 lbs + roughly 10 lbs of chassis, electronics, battery weight).	Ensure the OWB is at an idle state such that the wheels are not spinning. Load the oxygen tank onto the robot and add any extra weight to help simulate a full oxygen tank. Set up a specific distance for the robot to drive. Using a stopwatch, time the robot as it travels the specified distance. Using an external RC controller drive the robot forward until it passes the specified distance. Using the time measurement and the specified distance, measure the speed and ensure it is > 3 MPH.

UWB Subsystem R&V

Requirements	Verification
Utilize ToF or TDoA localization to create a triangle of known lengths with vertices defined by each UWB transceiver.	Ensure the OWB is at an idle state such that the wheels are not spinning. Then, place the UWB Tag one meter away from the robot. Utilizing the visualization program by MakerFabs, we will test our algorithm by comparing the location that was calculated for the tag with the real-world location. Confirm that the proper location is displayed.

Computer Vision Subsystem R&V

Requirements	Verification
Utilize OpenCV or OpenPose to determine $\Delta\theta$ of the child respective to the center of the robot.	Ensure the OWB is at an idle state such that the wheels are not spinning. Then, stand slightly off-center from the front of the robot. While having the Raspberry Pi plugged into the monitor, display the video feed with overlayed boxes for the tracked object. Using an additional overlay on the video, there should be an angle that defines the degrees from the center of our target. Compared with the measured value.

Close Range Object Detection R&V

Requirements	Verification
The bumper-style limit switch should trigger an alert system and stop all motors until the bumper is depressed.	Ensure the OWB is at an idle state such that the wheels are not spinning. Then, move the UWB tag such that the Robot must move to compensate for the change in the Tag's location. During the robot's travel, trigger one of the limit switches. The robot should stop and the alert system should trigger.

Control & Power Subsystem

Requirements	Verification
Control Subsystem: Using the direction and angle vectors from the UWB Subsystem, and angle from Computer Vision Subsystem we should compute a direction vector for the robot to move.	Connecting a serial output to our STM32, we should be able to read a direction vector in the frame of reference of our robot. Compare it to the location of the UWB Tag.
Power Subsystem: Must be able to regulate battery voltage to power components throughout the discharge cycle of the battery and automatically cut out power when battery voltage drops too low.	Connect the input of the voltage regulator to the voltage supply. Connect the output of the voltage regulator to a programmable load. Set voltage supply to maximum battery voltage. Check the voltage reading with a multimeter to make sure the output voltage does not fall outside of 12V $\pm$ 5% under no load and full load.

Block Diagram