**18/10/24** **Progress:**

* Team roles were split: Mechanical design (Sam, Dylan) and electrical/computing (Oscar, Trystan, Tye).
* ROS, YDLIDAR, and Hector SLAM were successfully set up on the Raspberry Pi for real-time mapping using RViz.
* Development started on both manual and autonomous control of a mobile robot platform for mapping.
* Oscar focused on motor control, Trystan and Tye on implementing wireless manual control, Dylan designed and 3D-printed a holster for the LIDAR and Raspberry Pi.

**Issues:**

* Initially attempted to use RPLIDAR library; later switched to a compatible (but outdated) YDLIDAR Git repo.
* SSH setup failed using Pi hotspot; resolved by switching to standard WiFi, which is incompatible with university WiFi. Modem to be acquired.
* USB aliasing between the LIDAR and Pi required debugging; resolved with ChatGPT guidance.
* Setup.bash not sourced in each terminal session led to unnecessary reinstallations.
* RViz failed to install due to a typo ("htp" instead of "http") in source.list. Fixed using a search script and manual correction via nano.
* Motor driver pin conflicts between Pi and Arduino were resolved by reassigning digital pins and using analog pins as digital outputs.

**05/11/24** **Progress:**

* Velcro applied to tidy up the robot's layout.
* Team attended soldering induction, gaining access to a dedicated lab.
* Wireless motor control now functional using laptop keyboard inputs.
* First saved room map using Hector SLAM.
* Due to aggressive motor startup (cut-in speed), the robot's rotation affected SLAM accuracy. Currently mitigated by using low-speed maneuvers to improve map quality.

**Issues:**

* Use of hardware with limited support (e.g., YDLIDAR on Ubuntu 20.04 with ROS Noetic) led to frequent troubleshooting.
* Multiple ROS scripts required porting from Python 2 to Python 3 for compatibility.

**20/11/24** **Progress:**

* TF tree reconfiguration complete; node and transform alignment working.
* Awaiting recharged battery from University to proceed with tests.

**04/12/24** **Progress:**

* Obstacle avoidance node added: robot now slows or stops based on LIDAR proximity data in a 245° frontal arc.
* Began tuning PID parameters for smoother acceleration and braking.
* Created a launch file to initiate all core nodes: SLAM, odomtransformer, teleop, and serial connection.

**Issues:**

* Occasionally dropped packets in ROS-serial connection caused motor stutter. Mitigated by improving cable shielding and baud rate tuning.

**12/01/25** **Progress:**

* Integrated Arduino Nano IMU (yaw) with ROS odometry via odomtransformer node.
* Map localization improved dramatically when rotation data fused with scanmatch\_odom.
* Started implementation of artificial potential field (APF) navigation logic on Raspberry Pi.

**30/01/25** **Progress:**

* Added goal setting functionality through RViz: user can click-to-set goal.
* Coordinates transformed into local reference frame, then sent via serial to Arduino for directional logic.
* Boolean "goal reached" flag added.

**Issues:**

* Small delay in transform broadcast occasionally causes offset between visual goal and true robot trajectory.

**10/02/25** **Progress:**

* Tested full integration of LIDAR + IMU + motor control + goal setting.
* Conducted indoor navigation trials using APF logic.
* Began writing technical documentation for group report.

**05/03/25** **Progress:**

* Refactored codebase to improve modularity: nodes are now structured into packages.
* Added README and project documentation draft for GitHub.
* Final lab tests performed and recorded: obstacle avoidance and basic navigation successful.

**Issues:**

* RealSense RGB-D camera integration deferred due to time constraints.
* Final mapping performance adequate, though occasional loop closure failure noted.

**20/03/25** **Progress:**

* Finalised written report for submission.
* Group rehearsed for technical viva with demo scripts and division of speaking roles.

**28/03/25** **Milestone:**

* Final Report Submitted ✅

**11/04/25** **Milestone:**

* Technical Viva Completed ✅
* Robot demonstrated successfully; navigation and obstacle avoidance presented live.