SIT310 – Final Project Description

In this unit, you have learnt how to use ROS2 to build robotics applications. You have built robotics applications involving a mobile robot and a drone.

The final project consists of 3 parts:

1. Project Proposal
2. Project Demo
3. Project Report

For your final project, you will select a project idea. Check out the example projects in the task resources. You are encouraged to propose your own project idea that may be a combination of features you see in the example projects list.

Your mark for the final project will depend on what level (P/C/D/HD) you attempt to solve and how good of a job you have done. The tasks are accumulative i.e., if you are targeting D mark, you will need to do P and C tasks as well. Note that the difficulty of the projects differs, and it will be considered when your final mark is being determined. This means that you shouldn't be deterred to take on more advanced projects – your efforts would still be rewarded even if you take on a project and you can't have it work perfectly. Even when you can't complete a given mark level – you can still be given a "complete" mark if teaching staff thinks that you have shown the effort.

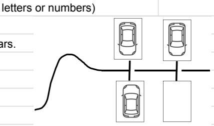
Note that the final project is individual – no groups! Even if you pursue the same project topic with a fellow student, you are expected to write your own code and your own report.

In helping you to complete this assessment, you can use any of the material in the unit task sheets, including code provided etc. It is sensible to use the work from the labs as a basis for your solution. You can use code that is publicly available, but only to assist you in achieving other tasks – you will only gain marks for work you have done yourself. If you are uncertain, talk to the teaching staff.

You should implement mostly using ROS2 functionality (unless noted in the project description). You can make improvements to the Arduino side as well, if you believe it is necessary.

Final Project Proposal Task (Last updated: April 21)

- Choose your project from the project list that's provided in the final page of this task sheet.
- Write a short proposal that includes the following:
  - The proposed project ID/title
  - The proposed robot platform
  - Your target mark for the final project
  - A brief discussion of your proposed technical approach to solve the problem.
  - List any 3rd party software/libraries you are planning to utilise for your final project.
  - If you have chosen "Simulation Guru" or "Independent Renegade" projects:
    - Propose features up to your target mark just like in the provided project examples.

#	Hardware	Title	Mark	Capability	Notes
1	Zumo or Tello	Gesture Control Wizard	P	A fixed number of Hand Gestures are recognized. The hand gestures will be recognized from a live video of the operator. I suggest using a camera that is connected to your PC (i.e. webcam)	
			C	Robot Moves according to simple hand gestures: go/stop/turn right/turn left .	
			D	Robot Communicates back information via motion gestures (i.e. shaking head) as well obstacle/robot visualization on rviz	
			HD	Robot Can handle "go forward/backward X meters" or "rotate left/right Y angles", input with hand gestures	It is up to you how to design these gestures. For instance, left hand can determine the command type and right hand can determine the number/level.
2	Zumo or Tello	Voice Control Commander	P	A fixed number of Voice Commands are recognized via Speech Recognition.	You can use the mic on your headphone/laptop/PC
			C	Robot Moves according to simple voice commands: go/stop/turn right/turn left.	
			D	Robot Communicates back information with speech and/or buzzer noise as well as obstacle/robot visualization on rviz	Robot won't speak through on-board speakers. The voice can come from your PC's speakers.
			HD	Robot Can handle "go forward/backward X meters" or "rotate left/right Y angles", input with speech comands	
3	Zumo or Tello	Teleoperation Phenom	P	Teleop with a Gaming Controller. Obstacles and virtual robot are displayed in rviz. User able to teleop by looking at rviz only	For accurate displaying of the obstacles, you may have to know where the ultrasonic sensors are located with respect to the robot's base (Zumo). Ideally, ROS2 tf library can be used for this purpose.
			C	Assisted Teleop: Robot executes user's input as much as possible while avoidsing obstacles.	
			D	User is able to teleop the robot using a GUI. Robot's speed is adjustable via the GUI.	
			HD	Teleoperation with gesture or voice	Look at Project#1 or Project#2
4	Tello	Selfie Drone Innovator	P	Robot is able to detect people. Robot keeps the person in the middle of the image by rotating in-place.	
			C	Robot is able to keep a given distance to nearest people by rotating and moving	
			D	Robot is able to land on user's hand when requested	
			HD	Person Recognition: Robot follows only the user, not other people.	
5	Tello	Reactive Flight Prodigy	P	Artificial Marker detection. Odometry is estimated and published to /odom topic. Detected marker poses and robot's model shown in rviz	I suggest that you find a ROS2 implementation of ARUCO markers. Mission Pads from Tello could be an option too.
			C	Robot is able to avoid a single object	
			D	Multiple, variable sized objects are avoided. Different markers are used to represent different objects.	
			HD	A simple local navigation algorithm is implemented, i.e. Dynamic Window Approach or Bug Algorithm	
6	Tello	Aerial Stunt Master	P	Robot is able to draw straight letters (i.e. X, M, N) and circular numbers (i.e. 3, 6, 9) in the air on the horizontal plane (fixed height)	
			C	Robot is able to draw letters and numbers at different scales	
			D	Robot is able to draw consecutive letters and numbers (i.e. C-3PO, T8000, R2D2, your initials)	
			HD	You have an external camera that is looking at the drone. You detect the drone's position when it moves in vertical plane, and you generate an output image that represents drone's path (it should look like the drawn letters or numbers)	
7	Zumo	Parking Pro	P	Robot can find empty spaces, executes simple forward parking	It is up to you how to design a "parking lot" and represent other parked cars.
			C	Robot can execute a reverse parking in between two parked cars (obstacles)	You can use a line follower course as follows: 
			D	Robot can execute parallel parking in between two parked cars (obstacles)	
			HD	Human-in-the-loop parking: User can choose among parking options. Input can come from a controller, voice or gesture	
8	Zumo	Maze Runner	P	Robot able to reasonably explore a maze without hitting walls	You are responsible for building a maze. We don't have any in the lab.
			C	Robot can explore most of the maze	
			D	Robot capable of getting out of a simple maze	The "outside" can be encoded via a different ground colour (i.e. black vs white in inside the maze) so contrast sensors can be utilized
			HD	Robot "learns" the shortest route to get out of the maze. Takes shortest path after re-start.	
9	Zumo	Object Pushing Phenom	P	Robot is able to push an object consistently without dropping it (i.e. object stays in front of the robot)	No major modifications to the robot's front side
			C	Robot is able to push different sized objects	
			D	Robot can distinguish between pushable and non-pushable objects using its sensors	
			HD	Robot is able to collect multiple objects in a "goal" area	This feature may require localization to some degree. Alternatively you can design a black/white arena so contrast sensors can be utilized
10	Zumo	Reactive Navigation Jedi	P	Robot is able to circumnavigate a static, polygonal obstacle via wall following without touching it.	
			C	Odometry-based localization is implemented. Robot is able to recognize when it makes a full lap around the object.	
			D	1 bug algorithm is implemented. Choose from Bug0, Bug1 or Bug2	
			HD	2 bug algorithms is implemented. Choose from Bug0, Bug1 or Bug2. Performance is compared.	
11	Zumo	Line Follower Expert	P	Follows continuous lines reliably. Robot is able to handle 90 degree turns	
			C	Obstacle Avoidance while following the line	The robot is expected to leave the line when the obstacle is detected, traverse around, and find the line again, and continue following
			D	White on black and Black on White following both works	Consider a track like this: https://www.youtube.com/watch?v=QEkxuvSierQ
			HD	"Learn" the arena and create a metric map of it, and output in the form of an image	
12	Zumo	IMU Wizard	P	Kalman filter or similar implemented for Gyro and Accelerometer. When an impact is detected, the robot makes a noise.	This project should be implemented on the Arduino side only, due to the fast reaction needed
			C	Implement "Always Face Uphill" or "Resist Being Rotated" feature	0:59 in this video: https://www.youtube.com/watch?v=Jtwu7-T1a9o
			D	The robot can reliably estimate the direction of the impact.	
			HD	Inverted Pendulum via PID Control	1:42 in the same video: https://www.youtube.com/watch?v=Jtwu7-T1a9o
13	Zumo	Mini Sumo Champ	P	Offense: The robot should push static objects out of the ring without going outside itself.	We don't have a dojo. You'll have to build yourself if you take on this project.
			C	Robot takes user teleop inputs (keyboard or controller) but stays in the dojo no matter what input is provided.	Since the robot needs to operate untethered, the logic should only be implemented on the Arduino side.
			D	When an impact is detected, the robot makes a noise. Defense: The robot can reliably estimate the direction of the impact. Robot tries escaping the impact.	
			HD	Implements an advanced Mini sumo Competition strategy or a mechanical solution. Tournament winner (if multiple students participate) or wins against Unit Chair's robot (if available). In the absence of competitors, Implement "Always Face Uphill" or "Resist Being Rotated" feature.	
14	Zumo or Drone	Robot Learning Visionary (Advanced)	P	Implement Teleop on the robot. Record a bunch of "expert demonstrations" data.	
			C	Read a paper about Deep End-to-End learning paradigm and write a 2 page report on what you understood in your own words.	The papers should be a well-cited one (>500 citations on Google Scholar) published at a reputable conference/journal.
			D	Train a neural network that takes robot sensor data as input (Ultrasonics or Camera) and Expert action as output (Robot Velocity). Analze the accuracy of your model.	
			HD	Run the robot autonomously with the actions that come from the neural network you have trained, in real-time.	
15	Camera	Computer Vision Specialist (Advanced)	P	Run a pre-trained neural network. i.e. face detection, object detection etc	A good GPU is needed for this one.
			C	Read the paper for the network you trained and write a 2-page report on what you understood in your own words.	The papers should be a well-cited one (>500 citations on Google Scholar) published at a reputable conference/journal.
			D	Train a computer vision neural network from a dataset.	
			HD	You collect data, label it and train your own network or fine-tune a network.	
16	RGB-D Camera	Point Cloud Enthusiast (Advanced)	P	Point Cloud visualization, filtering, downsampling, outlier removal.	We have Kinect cameras in the lab available for lending
			C	Planar surface (table, walls, ground) detection in Point Clouds	
			D	Object Detection via Clustering Point Cloud Data	
			HD	An Advanced Feature i.e. ICP, Point Cloud Registration, Object Detection, Surface Reconstruction, 3D Object Detection	
17	None	Simulation Guru	P	Propose a P-level task from any other project in a simulator (subject to Unit Chair Approval)	
			C	Propose a C-level task from any other project in a simulator (subject to Unit Chair Approval)	
			D	Propose a D-level task from any other project in a simulator (subject to Unit Chair Approval)	
			HD	Propose a HD-level task from any other project in a simulator (subject to Unit Chair Approval)	
18	Any	Independent Renegade	P	Propose a P-level feature (subject to Unit Chair Approval)	
			C	Propose a C-level feature (subject to Unit Chair Approval)	
			D	Propose a D-level feature(subject to Unit Chair Approval)	
			HD	Propose a HD-level feature (subject to Unit Chair Approval)	