SIT310 – Project Assessment
Final Project

Overview

This document describes the final project for SIT310. The assessment is split into two parts, 1) a report, and 2) prototype and presentation.

You are asked to plan, build and present a prototype smart-car based on the robotics platform and technology you have being using during the unit.

Project Aim

Using ROS as the foundation, you should research, plan, build and present a prototype of a smart car based on the Zumo 32U4 platform. Note that you should limit the Arduino on the robot to just message passing with no real logic. The decision making should be performed in ROS. It is also expected that the robot will be tethered using USB. You should choose and justify an appropriate architecture, including centralised, decentralised, subsumption to other style.

From the labs, you have been given code that allows the robot to:

- Be fully controllable using programs through ROS topics
- Automatic movement and fixed obstacle avoidance
- Visualisation of movement and obstacles and post using rviz

You should select three of the following advanced features. You need to research your chosen features and implement them using ROS and your robot. A good solution will integrate all three at the same time.

- Navigation using line sensors
- Automatic changing of lanes (using a printed road and the line sensors)
- Improved accuracy of robot pose and movement using e.g. gyroscope and other sensors
- Generate a persistent map in rviz as the robot moves
- Moving obstacle avoidance – predicting movement of moving objects.

- Wireless communication (will need hardware such as Zigbee or Bluetooth modules)
- Vision recognition for object detect/navigation (will need to use a camera, does not need to be on board the robot)
- Using the navigation stack and planning to navigate effectively
- GUI to control the robot effectively
- Conflict resolution (if you can't slow down, how do you choose which car to hit?)
- Acceleration management - after a period of time with no obstacles
- Interaction with traffic lights and smart motorway sensors
- Interaction with other cars
- The addition of sensors on the robot to add functionality
- Something else that is suitable – use your imagination

Some of these options may require the use of equipment not used so far in the lab tutorials. You can use actual hardware, simulation, mobile phone or internet sources of data, manual interaction, or any sensible approach. Please talk to the unit chair if you are unsure.

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 unit chair.

The "Report" is broadly focused on identifying the various issues smart-cars need to solve and providing a design of a ROS-based solution to solve them using your robot. You need to implement these features. The "Prototype and Presentation" is for you to provide evidence of how much you have implemented. You should write the report first in case you don't implement all the features from the report, this is fine.

Assessment – Report

You are required to provide a report on the design of a smart-car using the ROS platform. You should present a report of 1500-2000 words that contains the following information:

1. A description of the problem of automated driving, and the tasks that smart cars have to do. You should research the minimum base requirements that a smart car needs to achieve and describe how ROS can help it achieve these. You should include a ROS process graph.
2. A detailed discussion of advanced feature 1. Including the background research, description of how to use ROS to achieve the feature and a process graph.

3. A detailed discussion of advanced feature 2. Including the background research, description of how to use ROS to achieve the feature and a process graph.
4. A detailed discussion of advanced feature 3. Including the background research, description of how to use ROS to achieve the feature and a process graph.

Your report should be of professional quality, with cover page, executive summary, contents page, page numbers and references section. You will submit your report as a PDF or Word (.doc, .docx) document to the unit site.

In undertaking this report you MUST source appropriate reference materials, such as books, scholarly magazine articles, journal articles or conference papers that provide an account of research in this area. You will need to *read*, *interpret*, *summarise* and *convey* a basic understanding of this material in your report, and **appropriately reference** your sources using in-text citations (https://www.deakin.edu.au/students/studying/study-support/referencing). You are required to apply the **Harvard reference style** in your report (http://www.deakin.edu.au/students/study-support/referencing/harvard) and adhere to best practices of academic integrity when using the work of others. This task is as much about your ability to read and use information sources appropriately as it is about your ability to investigate an open problem in robotics.

Reference list entries DO NOT count to the word limit of this report.

Assessment - Prototype and Presentation

The assessment for this part is for you to provide evidence of what you have achieved. You will need to complete the following, which will be submitted to the unit site assessment folder.

1. Your code.
2. A short (e.g. 1 minute) Video with audio-narration demonstrating your prototypes functionality (both Hardware and Software).
3. A PowerPoint presentation with audio-narration (around 5 minutes).
4. A tick sheet indicating what you achieved

Due Date

The final submission deadline for this task is **9:00 AM AEST Monday 27th May 2019**. You must upload your submissions to the appropriate assignment submission folders on the Unit Site no later than this deadline. Penalties apply to late submissions in accordance with University assessment policy (5% penalty per day up to 5 days, then 100% penalty thereafter). See the section below on Assignment Marking for more information regarding late submissions. Extensions to this deadline will only be granted where documented evidence of hardship or serious illness is provided, in accordance with Faculty guidelines. Extension requests must be made using the appropriate form (available for download from http://www.deakin.edu.au/sebe/students) and submitted via email to the Unit Chair no later than

12:00 noon on Saturday 25th May 2019. You MUST also forward all work completed for this task up to that time. Except in the most extreme cases, failure to demonstrate any progress toward the assessment goals when requesting an extension will result in your request being denied.

Assignment Marking

This project contains two parts, the prototype and presentation, and the report. These are worth 30% each of the overall unit grade. Each assignment is marked out of 30 and contributes 30% toward your unit grade.

Rubrics are provided at the end of this document to indicate the criteria upon which your submission will be assessed and the standards that will be applied. In accordance with Faculty assessment policies, late submissions will incur a penalty of 5% of the total available marks per day, up to five days total (25% penalty). Submissions will not be accepted or marked more than five days after the final submission deadline, except in cases where an extension has been approved prior to the deadline.

Getting Help and Support

Students are required to complete their submission individually and all submitted work must be the work of the respective student. However, students are encouraged to work together to discuss the task, as well as to assist in developing an understanding of the problem they are investigating, and the solution methods documented in the literature.

Additionally, students should seek guidance and advice from the teaching team, either during classes or through the discussion forums on the unit site. Additionally, there are a range of University-wide support services available to students that are likely to be of help in undertaking this task:

Library Study Support: http://www.deakin.edu.au/library/study
Library Help: http://www.deakin.edu.au/library/help
Writing Mentors: http://www.deakin.edu.au/students/study-support/writing-mentors
Deakin Guide to Referencing: http://www.deakin.edu.au/students/study-support/referencing

Feedback

Students will receive written and/or audio feedback on their final submission, particularly with respect to how their work could be improved. Prior to submission, students are encouraged to discuss their chosen problem and investigation with teaching staff, who will provide advice, guidance and feedback where appropriate.

Students may submit a draft report no later than 5:00PM AEST Friday 10th May 2019 via email to the unit chair. Written and/or audio feedback will be provided no later than 5:00PM AEST Friday 17th May 2019. Students should then use such feedback to guide improvements and corrections to their report, before final submission prior to the aforementioned final deadline.

Marking Grid - Report - (30% of Unit)

Criteria	Basic (1-2)	Intermediate (30)	Advanced (4)	Complete (5)	Marks Awarded
Problem Description	Problem is not well defined or description lacks significant detail needed to understand problem features or why it is a problem	Problem description lacks detail and demonstrates insufficient understanding of problem features	Description conveys major features of problem and relates relevant dimensions that make the problem hard to solve	Description presents cohesive and thorough understanding of the problem and why it is difficult to solve given current knowledge and capabilities in computing and/or engineering	/5
Requirements of smart-car	List of unstructured system requirements	Functional and none-functional (quality) requirements listed. Good attempts relate these to high-level system requirements.	Ranked requirements based on an estimation of effort or time needed.	Relationships between requirements clear including risks of not completing features.	/5
Overall ROS design and ROS Process graph	Simple description of how the base system can be built using ROS. Simple ROS process graph.	Identified and justified ROS technologies to use. Annotated ROS graph.	Description conveys a clear design of how to implement the solution in ROS.	Description presents cohesive and thorough understanding of how to apply ROS to a problem.	/5
Advanced Feature 1	Appropriate feature identified. Basic description of feature. Justification of the need for the feature.	Details of how it can be implemented in ROS. Figures or diagrams to illustrate the feature.	Description conveys a clear design of how to implement the solution in ROS.	Description presents cohesive and thorough understanding of how to apply ROS to this feature.	/5
Advanced Feature 2	Appropriate feature identified. Basic description of feature. Justification of the need for the feature.	Details of how it can be implemented in ROS. Figures or diagrams to illustrate the feature.	Description conveys a clear design of how to implement the solution in ROS.	Description presents cohesive and thorough understanding of how to apply ROS to this feature.	/5
Advanced Feature 3	Appropriate feature identified. Basic description of feature. Justification of the need for the feature.	Details of how it can be implemented in ROS. Figures or diagrams to illustrate the feature.	Description conveys a clear design of how to implement the solution in ROS.	Description presents cohesive and thorough understanding of how to apply ROS to this feature.	/5
Overall:					/30

Note that poor use of presentation and English will reduce marks.

Marking Grid - Prototype and Presentation – (30% of Unit)

Criteria	Basic (1-2)	Intermediate (30)	Advanced (4)	Complete (5)	Marks Awarded
Advanced Feature 1	Some aspect of feature implemented	Feature implemented and demonstrated working.	Complex feature complete and working flawlessly.	Well implemented more than just working. For example, creating python classes that expose an API for use – or ensuring the feature is scalable – or testing it extensively.	/5
Advanced Feature 2	Some aspect of feature implemented	Feature implemented and demonstrated working.	Complex feature complete and working flawlessly.	Well implemented more than just working. For example, creating python classes that expose an API for use – or ensuring the feature is scalable – or testing it extensively.	/5
Advanced Feature 3	Some aspect of feature implemented	Feature implemented and demonstrated working.	Complex feature complete and working flawlessly.	Well implemented more than just working. For example, creating python classes that expose an API for use – or ensuring the feature is scalable – or testing it extensively.	/5
Integration	Features as part of same code base.	Features all usable without any changes in code.	Seemless and well-integrated features that are well integrated in the system.	Features exposed using a consistent API or system structure – such as a subsumption architecture.	/5
Execution (evaluated through video demo)	Demonstration showing features working.	Fluid demonstration of features. Well executed and smooth movement.	Demonstration of integrated features working seemlessly	Impressive demonstration of complex features.	/5
Presentation	Basic Powerpoint with some issues	Clear explanations of features implemented, including good explanations	Presents cohesive and thorough understanding of the problem and the solution.	Presents unique insights into the use of ROS to implement robotics applications.	/5
Overall:					/30