Project for Gesture Based UI Development

Due Date: 12th April 2019 (or earlier is acceptable)

Develop an application with a Natural User Interface. There are a number of options available to you and this is an opportunity to combine a lot of technology that you have worked with over the past four years.

At the very least, this should be a local implementation of the application using gestures to interact with it. For example, a voice controlled application fits the parameters of gesture based control. You can expand out to include real-world hardware and use this as an opportunity to prove a concept. The Internet of Things is a common phrase, so you could implement a solution taking advantage of hardware like the Raspberry Pi, using the cloud for data transfer and creating a real-world scenario through this medium.

You can **reproduce** a classic game or system using a gesture-based interface. For example, a platformer game or a navigation application using Kinect or voice control. Maybe Tetris using the Myo armbands to control the blocks, or Flappy Bird using the Kinect as the controller. Applications with multiple users are also acceptable.

Voice control applications need to be more complex and achieve something. Creating a skill in Alexa for the sake of creating a skill is not enough. You need to take the application further than this. You could, for example, implement a simple learning mechanism that will build a conversational skill as time progresses and demonstrate this. You could use the voice control to progress through a game or achieve a task. If you are doing this, then you need to distinguish the code you write from the samples available.

The programming language is your choice.

Hardware available

- (9) Myo Armbands, (2) Leap Motion Controllers, (2) Kinect V2, (2) Hololens, (6) Durovis Dive which are similar to Google Cardboard
- Raspberry Pi, Arduino, Lego Mindstorms (open to you)
- Other stuff you may have……

Requirements

Write up the project under the following headings including all references as evidence of your research.

Purpose of the application – design of the application including the screens of the user interface and how it works. The application can be an experimentation process for you, testing how pieces of hardware could interact or be combined with gestures. You don't have to solve the world economic crisis just yet.

Gestures identified as appropriate for this application – consider how gestures can be incorporated into the application, providing a justification for the ones that you pick. This is an important research element for the project and needs to explain how the gestures fit into the solution you are creating.

Hardware used in creating the application – You are not limited to the hardware listed above. If you have your own hardware, or hardware simulator that you wish to use, then feel free. The purpose of each piece of hardware should be given with a comparison to other options available.

Architecture for the solution – the full architecture for the solution, including the class diagrams, any data models, communications and distributed elements that you are creating. The architecture must make sense when the gestures and the hardware are combined. Justification is necessary in the documentation for this. You need to include a list of relevant libraries that you used in the project.

Conclusions & Recommendations – Conclusions are what you have learned from this project and the associated research. Recommendations are what you would do differently if you were to undertake the project again. The Reflective Piece – what I learned and "enjoyed"! This gives scope for a critical evaluation of the project and the objective that you tried to achieve.

Marks Awarded

Project is worth 60% of the module marks and is to be submitted through Git, with marks awarded in the following areas

- Architecture of solution implemented and working (the code part) – 35%
- Gesture consideration and use – the rationale for the gestures used needs to be well documented, explained and supported 30%
- Documentation – of the architecture and implementation, conclusions etc. 20%
- Class presentations – 15% (must stay for the others to be considered for these marks)
- Video – a short video demonstrating your project. This is considered part of the documentation.

Marking Rubric for Implementation & coding

0 - 35%	35 – 75%	75 – 100%
A selection of the basic requirements is implemented to a basic level Implementation may contain some syntax and/or run-time errors Implementation code is poorly documented and/or formatted Application will not be tested properly Implementation code will not follow applicable coding conventions	Requirements have been implemented to an acceptable level Implementation will not contain syntax and/or run-time errors Implementation code will be reasonably commented and/or formatted Application will be tested to a reasonable degree Implementation code will follow appropriate coding conventions Little or no evidence presented of research for the design and implementation of the application	Implementation requirements have been implemented to an advanced level Implementation will not contain syntax and/or run-time errors Implementation code will be well commented and/or formatted Application will be comprehensively tested Implementation of code will follow coding conventions Evidence of significant research in the design and implementation of the application

Marking Rubric for Documentation

0 – 35	35 – 75	75 – 100
Poor use of grammar, structure and content with little evidence of knowledge of the problem domain Limited development and test plan presented Limited evidence of research, critical analysis and conclusions	Good use of grammar, structure and content with satisfactory evidence of knowledge of problem domain Acceptable plan presented to support the implementation and testing of the application Satisfactory evidence of research, critical analysis and conclusions	Written and structured to a high standard with content that exhibits a comprehensive knowledge of the problem domain Thorough development and test plan presented to support the implementation of the application Extensive evidence of research, critical analysis and conclusions