

# Project Topics

This document is a portfolio of the project topics that are being offered by academic staff this year for the Innovation Project module, organised alphabetically by staff surname. Please note that projects must include the undertaking of practical work of some sort using computing/IT technology. This is most frequently achieved by the creation of an artefact as the focus for covering all or part of an implementation lifecycle. Dissertations based solely on literature review activity and/or user/market surveys are not acceptable.

**Action:** Arrange an appointment with the project supervisor of the projects that interest you during the first two weeks of the Autumn Term.

**Action:** Complete the Project Preferences Form (download from Moodle):

- List the top three preferences for your project. They should ideally each be with a **different** supervisor.
- Upload the **Optional:** If you want to propose your own project:
- Talk to your potential supervisor about your idea **as soon as possible**. They must confirm to you that the project is suitable and be satisfied that it will undertake practical work of some sort using computing/IT technology.
- Use the first option on the preferences form to list the project idea, making sure you state "Own Project Idea" with the title. You should still select two other projects from the portfolio.

Note that due to staff load you may be allocated a project that is not on your list. We will do our best to give you one of your preferences, however, this will always be resources permitting. It is important that your three preferences are each with a different supervisor where possible and you engage with those potential supervisors in advance of submitting the form, so they know who best to prioritise.

Staff will have the final choice on who is selected for their projects.

This key has been used to highlight which projects are suitable for different degree courses:

CS	SE	GD	Cy
----	----	----	----

CS      Computer Science  
SE   Software Engineering   GD  
Games   Development   Cy  
Cybersecurity

## Jules B

### Areas of Interest:

- Industry 4.0 R&D - Digital Product Design for Industry
- IoT Product Design
- HCD – Human Computer Interaction
- Accessibility Design
- VR & AR for use in Commercial Environments, i.e., Training in Industry
- UX – User Experience Design & Psychological Principles
- UI – User Interface Design
- Wearable Technology including Apparel
- Secure Cloud based Design & Implementation
- System Architecture
- Project Management
- Graphic Design
- Mobile Application Development

### 1. Design and develop a prototype (3D model or working model) for an IoT device.

CS	SE	GD	Cy
----	----	----	----

This would entail a detailed design process, using UX/UI techniques, software and system architecture and research into the marketplace for your product.

The theory is to design (and build if possible) a functional MVP (Minimal Viable Product) prototype for demonstration and consideration as a viable product for marketplace.

Consider the use of cloud-based technology into your design and use of 3<sup>rd</sup> party plug ins and library's, even the use of LLM's and RAG's, plus the security levels needed for that.

Other considerations are the use of microcontrollers and a controller interface such as a mobile application plus the overall, usability and safety in the design.

Skills to be used: UI/UX design, 3d modelling, systems architecture diagrams, practical physical computing, programming. Project management techniques and industry standards to be considered throughout.

## 2. Dynamic Web Application.

CS	SE	GD	Cy
----	----	----	----

Full stack design, build and deployment of a dynamic database drive web application.

There needs to be a certain amount of rationale for the decision on theme or product, so a thorough analysis of the market area and viability study should be undertaken.

Consider the following when completing a project like this re, web systems architecture drawings, UX theory applied and heuristically tested and full sets of UX, UI prototypes ready for demo as part of the viability study.

Exercise full SDLC project management skills to help with the development, and use of tools such as kanban boards, UX design tools and innovation boards, version control, security is paramount and various testing techniques and tools, especially prior to deployment. WCAG standards must be adhered to and fully researched and tested.

Skills to be used: UI/UX design, architecture diagrams, cloud technology, cyber security, programming, databases. Project management techniques and industry standards to be considered throughout.

## 3. Any other projects suggested by students are also considered.

## Dr Lee Bee

### Areas of Interest:

- Game design and development
- Level design process and implementation
- Augmented reality games (e.g. ARKit, ARCore and ARFoundation).
- Virtual reality games/implementations
- 3D asset creation

## 1. Developing a single player level for X game

CS	SE	GD	Cy
----	----	----	----

*Main Skills Required: Design (Heavy) and Programming (Medium)*

This dissertation topic aims to replicate the level designer role in a studio by asking students to develop a single player level for an existing game. Students will be required to research level design theory, plan out a level, prototype it, test/iterate and finally complete an art pass. Students will be expected to run several play test sessions throughout the process. Good problem-solving skills are required as it is also up to the student to learn the level editor and programming environment of the chosen game.

There are many games on Steam that support modded levels but not all allow for enough depth for a dissertation, therefore some recommended games are:

- Dungeon Defenders (UE3)
- Half Life: Alyx (Source 2 – requires a VR headset)
- SOMA (HPL Engine 3)
- Left 4 Dead 2 (Source)
- Dying Light (Chrome Engine 6)
- Hello Neighbour (UE4)
- Fallout 4 (Creation Kit)
- Starfield (Creation Kit)
- Thunder Tier One (UE4 – Campaign)
- Human: Fall Flat (Unity)

## 2. Developing an augmented reality game using ARKit/ARCore

CS	SE	GD	Cy
----	----	----	----

*Main Skills Required: Design (Medium) and Programming (Medium)*

Students will develop a game that makes use of augmented reality on a smartphone using either ARKit, ARCore or ARFoundation. Augmented reality requires a new way of design thinking and making effective use of the technology is essential to developing a successful AR game. The game

should be developed in Unity making use of the appropriate SDK for the desired platform or using Unity's AR Foundation package. Students will need to provide their own device, so please ensure your device is compatible with ARKit or ARCore.

### 3. Developing an animated game character

CS	SE	GD	Cy
----	----	----	----

*Main Skills Required: 3D modeling (Heavy) and Programming (Light)*

This topic requires students to research character modelling techniques and processes along with recent developments in the industry. Students will be expected to plan the character, create concept drawings, model the character (using modern techniques), rig the character and develop a set of basic animations. The character should then be implemented into either UE5 or Unity in an application that can demonstrate the different animations.

Students will need to research and develop their current modelling skills further to gain the skills required to develop a quality game character. The character can be of any style and can be based on an existing IP (as a new character).

### 4. Developing a virtual reality game for Meta Quest or similar

CS	SE	GD	Cy
----	----	----	----

*Main Skills Required: Design (Medium) and Programming (Medium-Heavy)*

This topic requires students to research virtual reality game design and usability concepts to develop a virtual reality game. Students are free to choose the game genre but are expected to make use of the input/output features of their HMD of choice. Students may wish to utilise the Unity XR Interaction Toolkit or code their own interactions from scratch, depending on their design or programming focus. There will also be the opportunity to integrate with my LevelEd VR software

It is recommended that students have their own VR equipment and capable PC, however, we do have several Meta Quest devices available, but these will be first come first serve and shared across the whole department.

### 5. Developing a networked multiplayer game in Unity/UE5

CS	SE	GD	Cy
----	----	----	----

*Main Skills Required: Programming (Heavy) and Design (Medium)*

This dissertation topic extends and builds on top of skills developed in CO5035 by focusing on developing a networked real-time game using Unity or UE5. Students choosing to undertake this topic will develop new skills and gain an understanding of an essential topic, such as multiplayer networked games. Students choosing this topic should ensure they are comfortable working in

Unity or UE5 and exploring how to use networking APIs, including 3<sup>rd</sup> party Unity plugins such as Photon, Mirror or the new Netcode for Game Objects from Unity.

Students choosing this topic are free to decide on a game genre in which to build the networking framework around. Past examples have included a first-person shooter, a turn-based RTS and a co-operative wave-based shooter.

## 6. Developing a new multiplayer experience for Fortnite using Unreal Editor for Fortnite (UEFN)

CS	SE	GD	Cy
----	----	----	----

*Main Skills Required: Programming (Medium) and Design (Medium-Heavy)*

In March 2023, Epic Games launched Unreal Editor for Fortnite. This software allows developers to create truly original content for the Fortnite game, rather than using the limited built in tools. The software uses a modified version of Unreal Engine that allows developers to bring in their own custom assets, write custom code, create cutscenes, etc. and then deploy this to Fortnite for testing and ultimately, publishing. A student who takes on this topic will be required to create their own game experience using UEFN, test it online with players and finally publish the game to the platform. This topic will require students to learn the new toolset, including the new Verse programming language to succeed and so the scope of the game idea will be important to consider.

## 7. Developing a Mechanics Gym

CS	SE	GD	Cy
----	----	----	----

*Main Skills Required: Programming (Heavy) and Design (Medium)*

This dissertation topic would be suitable for any student interested in becoming a game mechanic programmer. A mechanics gym features a playable space that demonstrates core character mechanics implemented to a high standard and with multiple variations. For example, you may programme multiple different ways for a character to open a door. The focus of this project is on developing polished mechanics that show you understand the polish required for commercial games. This project can be completed in either UE5 or Unity, however, it is recommended that game programmers complete this in UE5 utilising C++ as the primary language.

## 8. Personal Project Ideas

CS	SE	GD	Cy
----	----	----	----

I'm also happy to discuss and support personal project ideas if they are within my areas of interest listed above and demonstrate a new set of skills being learnt by a student. Please discuss these with me before putting in your Project Preferences Form.

<b>Dr Joe Collenette</b>	
<b>Areas of Interest:</b> <ul style="list-style-type: none"><li>• AI &amp; Law</li><li>• AI &amp; Ethics</li><li>• Multi-Agent Systems</li></ul>	



## 1. Formalising the Rules of the Road for countries other than the UK for use in a Recommendation System for Driverless Cars

CS	SE	GD	Cy
----	----	----	----

### Background

The UK Highway Code specifies how drivers should behave on UK roads. Most of its rules are in the form of recommendations rather than hard and fast rules. RoTRA is a tool which takes a description of a road traffic situation and returns the recommendations of the highway code that apply to that situation. At the moment RoTRA has been tested on very simple examples around road junctions expressed in a grid world based simulation.

### Description

The object of this project would be to investigate whether the rules of the road for a country other than the UK can also be implemented in RoTRA and extend both the theory and implementation of RoTRA as necessary to allow this. A student taking on this project should ideally be familiar with the relevant rules of the road and, if necessary, with the language in which they are written. RoTRA has a formal underpinning so the student should be confident in logical reasoning and representing knowledge using logical expressions.

### Outputs

1. An evaluation of whether the RoTRA framework can be adapted to road traffic rules of another country.
2. An implementation of the road traffic rules of another country in RoTRA (with suitable adaptation of RoTRA if necessary)
3. Evaluation of the ability of cars in a simple simulation to follow the road traffic rules as they change between countries

### Relevant Literature

Joe Collenette, Louise A. Dennis and Michael Fisher. Advising Autonomous Cars about the Rules of the Road. Fourth Work- shop on Formal Methods for Autonomous Systems (FMAS 2022)

### Useful Background

Familiarity with Prolog will be an advantage to this project since RoTRA is implemented in Prolog.

## 2. Helping applicants to the European Court of Human Rights correctly fill in the applicant form

CS	SE	GD	Cy
----	----	----	----

### Background

In 2021 the number of applications the European Court of Human Rights processed was 44250, of which 32961 were either inadmissible or struck out. In 2021 there were 70150 applicants waiting for a decision. It is imperative that applications are dealt with quickly to enable an effective access to justice. Given that the vast majority of applications are inadmissible it is vastly important that applicants get help in producing good quality applications for the court to consider.

### Description

The object of this project is to identify parts of the European Court of Human Rights application form that can be filled out from the statement of facts that is given by an applicant. A website tool has been developed to help applicants to the court fill in the application form correct, the project can extend this to allow the form to be prepopulated to ease the cognitive load on an applicant.

### Outputs

1. A module that takes a human written statement of the facts for a given ECtHR application and outputs a number of true/false/undecided statements that match to the argumentation framework
2. An adaption of the website that incorporates the module into the argumentation framework.

### Relevant Literature

Collenette, Joe, Katie Atkinson, and Trevor Bench-Capon. "Explainable AI tools for legal reasoning about cases: A study on the European Court of Human Rights." *Artificial Intelligence* 317 (2023): 103861.

### Useful Background

The web tool was developed in HTML/CSS/JS using an argumentation framework, familiarity with argumentation and web development would be an advantage.

## 3. Integrating the CARLA Autonomous Car Simulator with the RoTRA Highway Code tool

CS	SE	GD	Cy
----	----	----	----

### Background

The UK Highway Code specifies how drivers should behave on UK roads. Most of its rules are in the form of recommendations rather than hard and fast rules. RoTRA is a tool which takes a description of a road traffic situation and returns the recommendations of the highway code that



apply to that situation. At the moment RoTRA has been tested on very simple examples around road junctions expressed in a grid world based simulation.

### **Description**

The object of this project would be to link RoTRA to the CARLA Autonomous Vehicle Simulation Tool. The project does not involve implementing a driverless car algorithm since CARLA comes with a number of baseline tools for this. Instead it should focus on providing an interface to the RoTRA system that can, at the least provide warnings when traffic rules are violated. An extension would be to investigate how a sub-system could be developed to “take over” from the baseline using RoTRA and enforce some behaviour such as coming to a stop.

### **Outputs**

1. A link between RoTRA and the CARLA simulator that can flag up road traffic violations. This would involve theoretical work on how information within CARLA can be transformed into information relevant to RoTRA.
2. The development of a module within CARLA which enables information from RoTRA to be used to over-ride the underlying baseline algorithm

### **Relevant Literature**

Joe Collenette, Louise A. Dennis and Michael Fisher. Advising Autonomous Cars about the Rules of the Road. Fourth Workshop on Formal Methods for Autonomous Systems (FMAS 2022)

Carla Open Source Simulator for Autonomous Driving Research: <https://carla.org/>

Hilal Al Shukairi and Rafael C. Cardoso. ML-MAS: a Hybrid AI Framework for Self-Driving Vehicles. AAMAS 2023.

### **Useful Background**

Familiarity with Prolog will be an advantage to this project since RoTRA is implemented in Prolog.

## **4. Interacting Ethical and Legal Considerations in Driverless Cars**

CS	SE	GD	Cy
----	----	----	----

### **Background**

The UK Highway Code specifies how drivers should behave on UK roads. Most of its rules are in the form of recommendations rather than hard and fast rules. RoTRA is a tool which takes a description of a road traffic situation and returns the recommendations of the highway code that apply to that situation. At the moment RoTRA has been tested on very simple examples around road junctions expressed in a grid world based simulation with simple driving protocols that either obey all the recommendations, none of the recommendations, or only those recommendations that are hard and fast rules.

In her paper “Ethical Considerations for a Decision Making System for Autonomous Vehicles During an Inevitable Collision.” 2020 28th Mediterranean Conference on Control and Automation (MED) (2020): 514-519. Lalis Milan-Blanquel considers how an ordering of ethical concerns can

be matched to various ethical theories in order to help a driverless car decide upon an appropriate course of action in a situation where it can not avoid a collision.

### **Description**

The object of this project would be to implement a driving protocol that obeyed the RoTRA recommendations unless it perceived there to be an ethically imperative reason to disobey them. This would involve integrating the legal recommendations into the ordering of concerns in Milan-Blanquel's framework and then showing how this affected the behaviour of the vehicle.

### **Outputs**

1. A theoretical framework showing how RoTRA recommendations rank as concerns in the ethical theories identified by Milan-Blanquel
2. An implementation of her framework in a suitable programming environment and linking it to RoTRA and the Grid World in which RoTRA has been tested.
3. Designing scenarios to illustrate the behaviour of the system

### **Relevant Literature**

L. Milfan-Blanquel, S. M. Veres and R. C. Purshouse, "Ethical Considerations for a Decision Making System for Autonomous Vehicles During an Inevitable Collision," 2020 28th Mediterranean Conference on Control and Automation (MED), Saint-Raphaël, France, 2020, pp. 514-519, doi: 10.1109/MED48518.2020.9183263.

Joe Collenette, Louise A. Dennis and Michael Fisher. Advising Autonomous Cars about the Rules of the Road. Fourth Workshop on Formal Methods for Autonomous Systems (FMAS 2022)

The BDI language is part of the MCAPL Framework that can be found at: <https://autonomy-and-verification.github.io/tools/mcapl>

### **Useful Background**

Familiarity with Prolog will be an advantage to this project since RoTRA is implemented in Prolog. There is also an implementation of ethical concern ordering (similar to Milan-Blanquel's) in a BDI language tool which has many similarities to Prolog and might serve as a suitable implementation platform.

## Dr Stuart Cunningham

### Areas of Interest:

- Affective computing
- Computer music
- Data compression
- Human-computer interaction
- Sonic interaction
- User experience

## 1. VR Reaction Exergame

CS	SE	GD	Cy
----	----	----	----

### Main Skills Required

Game development [Unity or Unreal] (Medium-Heavy), Programming (Medium), Interface Design (Medium).

### Summary

This project requires you to design, implement and test a Virtual Reality (VR) game. Essentially, in the game to be created the player is challenged to hit one of several virtual lights held within a matrix configuration (8x8 is suggested to start). The object of the game is to hit the light as fast as possible over the period of play (maybe two or three minutes). Game difficulty may be manipulated by changing the speed at which the lights turn on/off, duration of illumination, number of concurrent lights lit, and so forth. Think about the game as 3D version of the arcade game *Strike-a-Light* [[Video](#)] or the exergame system *Twall 64* [[Web](#) | [Video](#)]. Crucially, since this game is VR, you can have the lights appear all around the player as well as being able to control their depth, relative to the user, as mechanisms to manipulate game difficulty. User testing should include research to evaluate whether players found a simulated 2D version of the game different to a 3D version.

## 2. Navigating the Real World with Spatial Audio Cues

CS	SE	GD	Cy
----	----	----	----

### Main Skills Required

Mobile app development / APIs (Medium), Sound Design (Light), User Experience (Medium).

### Summary

This project will explore how sound can be used to enhance navigation software, such as Google or Apple Maps, by providing the user with auditory cues when they are wearing headphones using Head-Related Transfer Functions (HRTFs). For example, the destination they are trying to

reach could be sonified in a 360-degree space around the head of the wearer, thus indicating the direction in which the destination is. Distance from the destination might be conveyed by factors such as rate of pulsing, changing pitch, or increasing volume. The project will examine the various methods for audifying distance and direction information (and even other hazards and information) and determining which are most user friendly and effective.

### Useful References

Brewster, S. A., Wright, P. C., & Edwards, A. D. (1993). An evaluation of earcons for use in auditory human-computer interfaces. In Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems (pp. 222-227). ACM.

Holland, S., Morse, D. R., & Gedenryd, H. (2002). AudioGPS: Spatial audio navigation with a minimal attention interface. *Personal and Ubiquitous computing*, 6(4), 253-259.

Strachan, S., Eslambolchilar, P., Murray-Smith, R., Hughes, S., & O'Modhrain, S. (2005). GpsTunes: controlling navigation via audio feedback. In *Proceedings of the 7th international conference on Human computer interaction with mobile devices & services* (pp. 275-278). ACM.

## 3. Classic Video Games as Audio Games

CS	SE	GD	Cy
----	----	----	----

### Main Skills Required

Game Engine / Game development (Medium), Programming (Medium), Sound Design (Light).

### Summary

Audio Games are a type of computer game that provide no visual feedback to their player. Instead, information is primarily communicated to the player using sound, music, and haptic or tactile means. In this project, you should select a classic video game (such as *Pacman*, *Space Invaders*, *Pong*, *Horace Goes Skiing*, or similar) and produce an audio game version of it. You will need to think carefully about how the sounds are designed and deployed to provide game state information, as well as entertainment, to the player. As part of the process, you are expected to perform evaluations and user testing of the game, to determine how it might be refined and how easily people are able to interact with it and, hopefully, enjoy it.

## 4. A User-Friendly Cricket Score Book

CS	SE	GD	Cy
----	----	----	----

### Main Skills Required

Databases (Medium), Programming (Light-Medium), Web Development (Medium), interface design (Medium).

### Summary

The traditional process of recording the scores during a cricket match is challenging, requiring constant attention to the match being played, knowledge of the game, umpiring signals, and

various notations and symbols. To the untrained eye, the traditional score book [looks intimidating](#) and not easy to pick up. In this project, you are to create a web-based digital cricket scorebook system with an easy-to-understand interface that is designed around user need. The choice of web systems and technologies is yours, but it must be responsive and support access from a variety of devices (such as laptops, tablets, and smartphones). The idea is that the final system could be used by scorers at all levels of the game, but particularly those who are inexperienced.

## 5. Automatic Selection and Synthesis of HRTFs

(NOTE: Three distinct projects under this topic)

CS	SE	GD	Cy
----	----	----	----

### Main Skills Required (specific, depending on project)

Algorithms / Optimisation (Medium-Heavy), Programming (Medium), Image Processing / Computer Vision (Medium), Machine Learning (Medium).

### Summary

The use of spatial audio via binaural reproduction (think wearing headphones whilst using a VR or AR headset or even playing a game with headphones on) is increasing rapidly. This allows the listener to hear sounds in the virtual world coming from all directions around them, just like in the real world, and is achieved with the use of Head Related Transfer Functions (HRTFs). A major challenge in creating an accurate and immersive experience is providing the user (listener) with the best spatial audio possible. This is often difficult because to be completely accurate a HRTF model would need to be created for the head of every listener. This isn't currently practical. So instead, generic HRTF models are used. But this leads to sub-optimal and often poor user experiences. However, there are already a good selection of HRTF libraries available for example, so there's plenty of data out there.

- ⇒ *Fast HRTF Finder*: The aim of this project is to find a way to provide a user with the optimal, or near optimal, HRTF from an existing large library. This might be done by playing the user a small number of test sounds and asking them to choose between them to identify subsets of HRTFs that might be most suitable. There is a clear trade-off between the amount of time this must take and the quality of the HRTF selected.
- ⇒ *Photo HRTF Finder*: The aim of this project is to allow the user to take a photo, or series of photos of their ear(s) and for these to be analysed using computer vision, before key anthropometric features are identified and these used to select the best HRTF available in the existing library. The quality of the resulting HRTF should be verified.
- ⇒ *Fast/Photo HRTF Synthesizer*: Either by providing the user with some simple listening tests OR by asking them to upload a photo of their ear(s), synthesise a personalised HRTF for that user based upon learnings that can be achieved from an existing dataset(s).

### Useful References

- Miccini, R., & Spagnol, S. (2021). A hybrid approach to structural modeling of individualized HRTFs. In *2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)* (pp. 80-85). IEEE.
- Pelzer, R., Dinakaran, M., Brinkmann, F., Lepa, S., Grosche, P., & Weinzierl, S. (2020). Head-related transfer function recommendation based on perceptual similarities and anthropometric features. *The Journal of the Acoustical Society of America*, 148(6), 3809-3817.
- Wang, Y., Zhang, Y., Duan, Z., & Bocko, M. (2021). Global HRTF Personalization Using Anthropometric Measures. In *Audio Engineering Society Convention 150*. Audio Engineering Society.
- Miccini, R., & Spagnol, S. (2020). HRTF individualization using deep learning. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)* (pp. 390-395). IEEE.

## 6. Design and Implementation of a Raspberry Pi-Based Open-Source Alternative to the "Frogbox" for Live Cricket Streaming

CS	SE	GD	Cy
----	----	----	----

### Main Skills Required

Shell scripting (Medium), Programming (Medium), Web Development (Light).

### Summary

This project aims to design, develop, and implement a low-cost, open-source solution for live-streaming cricket matches events online. This should mimic the features of the commercial “Frogbox” equipment (<https://www.frogbox.live/>). The solution will be built around a Raspberry Pi, the system will utilise a webcam plus microphone for video capture and be designed for ease of use, requiring minimal to no technical interaction from the end user. A key feature of the system will be the ability to integrate a live scorecard from an external API, such as Play-Cricket.com (<https://play-cricket.ecb.co.uk/hc/en-us/sections/360000978518-API-Experienced-Developers-Only>), for automatic display during cricket matches. The Raspberry Pi, while powerful, may have limitations with processing video at high resolutions. Ensuring a balance between performance and cost will be critical. The final system should be foolproof, requiring as little interaction as possible. It should be easy for non-technical users to operate the box seamlessly. This project will provide the student with valuable experience in hardware-software integration, API interaction, automation, and open-source software development. It also has practical implications, making it a meaningful contribution to grassroots sports broadcasting.

## 7. A Cross-Platform Game to demonstrate Fitts' Law

CS	SE	GD	Cy
----	----	----	----

### Main Skills Required

Game Engine / Game development (Medium), Programming (Medium), statistics (Light-Medium), interface design (Medium).

### Summary

[Fitts' Law](#) is used to describe and predict the way that pointing tasks are carried out in interfaces based upon the distance the user must travel to reach a target (often a button) and the target size. This is useful, for example, in providing a difficulty index or estimated amount of time it takes a user to press a button on the screen. The concept can be illustrated and testing by getting users to perform a task (essentially a mini game) where they must click on a randomly sized and positioned target multiple times, whilst the time taken for each is measured.

There are some [examples](#) of this [in action](#) available. This project involves recreating this type of game/test, but making it available on multiple platforms, thus supporting users playing the game with, for example, a mouse on a desktop or laptop computer, or using their finger on a smartphone or laptop. Ideally, this new version of the game should do some analysis of the user's performance, by examining the regression trend their data generates and plotting this comparatively against the established Fitts' Law model, to see how they correlate.

### Useful References

Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology*, 47(6), 381.

MacKenzie, I. S., and Buxton, W. (1992). Extending Fitts' law to two-dimensional tasks. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 219-226).

## 8. Repertory Grid Software Development

CS	SE	GD	Cy
----	----	----	----

### Main Skills Required

Programming (Heavy-Medium), Statistics (Medium-Light), Interface Design (Medium).

### Summary

[Repertory Grid](#) is a technique used by psychologists and human-computer interaction researchers to understand how people perceive things in the world around them. It relies on data often elicited by semi-structured interviews or questionnaires. Essentially, it consists of multiple scales (called *constructs*) that are used by participants to rate things (called *elements*), typically by using a numbered Likert-style mechanism. These are then entered into a matrix (the Repertory Grid) where the data can be statistically analysed, primarily by ranking/ordering, correlation, and dimension reduction techniques (commonly Principal Component Analysis).

The job of this project is to develop a software system where users can enter their Repertory Grid data and be presented with the various analyses, graphs, and other information to help with their research, which they can then save and/or export. There is a very out-of-date, often unreliable, web-based version of the software here: <http://webgrid.uvic.ca/> that you can try. There have been [more recent attempts](#) to develop replacement systems, but none of these offer the same level of intuitive, ease-of-use. The software to be developed in this project should be individual and easy for anyone to use. You have the option to develop the Repertory Grid system as a standalone application (in which case it must run on Mac OS and Windows) or as a web-based application.

## 9. Detecting Alerts in Virtual Acoustic Environments

CS	SE	GD	Cy
----	----	----	----

### Main Skills Required

Game Engine / Game development (Medium), Programming (Medium).

### Summary

This project requires you to design and build several virtual environments, based on common, real-world, physical spaces. For example, these might include: an office environment, a shop, a hospital ward, a train carriage, and so forth. These models should include a realistic representation of the real-world, 3-dimensional, sonic environment, considering sources of background noise (e.g., people talking, phones ringing, environmental elements) that you would encounter in these spaces as well as their acoustic properties (how reflective or reverberant the spaces are). Once built, you should recruit a group of human participants to explore and test your environment and see how well they are able to perceive a set of pre-designed audio alert sounds (provided by the supervisor) within them. There is potential for the environment to be explored by participants either using a virtual reality headset or by using headphones and a traditional display screen.

### Useful References

- Cunningham, S., & McGregor, I. (2021). Evaluating Use of the Doppler Effect to Enhance Auditory Alerts. *International Journal of Human-Computer Interaction*, 37(11), 1074-1087.
- Edworthy, J. (1994). The design and implementation of non-verbal auditory warnings. *Applied ergonomics*, 25(4), 202-210.
- Hellier, E. J., Edworthy, J., & Dennis, I. A. N. (1993). Improving auditory warning design: Quantifying and predicting the effects of different warning parameters on perceived urgency. *Human factors*, 35(4), 693-706.



## 10. Audio Forensics: Event and Object Detection

CS	SE	GD	Cy
----	----	----	----

### Main Skills Required

Machine Learning (Heavy-Medium), Programming (Medium-Light).

### Summary

The idea of this project is to develop a piece of software or processing pipeline that can take an audio stream as an input (probably in WAV or MP3 format) and for it to be able to identify key events of interest that take place in the stream and label them accordingly. For instance, imagine taking a recording during a lecture or lab activity at the University and for the software to be able to identify each occasion where a door was closed or when someone in the room had their phone ring or receive a message alert. Key challenges will include identifying or synthesising a suitable dataset, separating the events of interest from the background noise, and identifying suitable features and machine learning techniques to classify events accurately.

### Useful References

- Babae, E., Anuar, N. B., Abdul Wahab, A. W., Shamshirband, S., & Chronopoulos, A. T. (2017). An overview of audio event detection methods from feature extraction to classification. *Applied Artificial Intelligence*, 31(9-10), 661-714.
- Clavel, C., Ehrette, T., & Richard, G. (2005, July). Events detection for an audio-based surveillance system. In *2005 IEEE International Conference on Multimedia and Expo* (pp. 1306-1309). IEEE.
- Portelo, J., Bugalho, M., Trancoso, I., Neto, J., Abad, A., & Serralheiro, A. (2009, April). Non-speech audio event detection. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 1973-1976). IEEE.

## 11. Fake Me a Picasso

CS	SE	GD	Cy
----	----	----	----

### Main Skills Required

Machine Learning (Heavy-Medium), Programming (Medium).

### Summary

Imagine being able to give a piece of software a series of images as input and asking it to synthesise one, or more, images that are similar in composition, style, colour, and so forth, as the original ones. That is the focus of this project, where it is anticipated that techniques such as Generative Adversarial Networks may be used to perform such a task. Ideally, what the user should be able to do is to provide your software with a relatively small set of 'example' pieces of artwork and for the system to produce something new, but in a similar style. The small input set will be a distinct challenge, and so you may need to consider providing some more 'general' training data to the system for it to produce variety and diversity of outputs. As such, you may

want to consider looking at more intelligent systems, such as [Midjourney](#), for additional inspiration and ideas.

#### Useful References

Wang, L., Chen, W., Yang, W., Bi, F., & Yu, F. R. (2020). A state-of-the-art review on image synthesis with generative adversarial networks. *IEEE Access*, 8, 63514-63537.

Xue, A. (2021). End-to-end chinese landscape painting creation using generative adversarial networks. In *Proceedings of the IEEE/CVF Winter conference on applications of computer vision* (pp. 3863-3871).

## 12. Exploring the Capabilities and Limitations of Large Language Models

*[NOTE: This project has been completely specified by using ChatGPT (https://chat.openai.com/chat) on 30-Mar-2023]*

CS	SE	GD	Cy
----	----	----	----

### Main Skills Required

Natural Language Processing (Medium-Heavy), Python programming (Medium-Heavy), and Machine Learning (Medium-Heavy).

### Summary

In recent years, large language models like GPT-3 have revolutionized the field of natural language processing, allowing machines to generate human-like text with remarkable accuracy. However, these models are not without their limitations, such as biases, ethical concerns, and lack of generalizability.

This project aims to explore the capabilities and limitations of large language models, specifically GPT-3, by conducting a series of experiments and evaluations. The project will involve the following steps:

1. Preprocessing and preparing datasets for training and testing the language model.
2. Fine-tuning GPT-3 on specific tasks such as text classification, summarization, and question answering.
3. Analyzing the generated outputs of GPT-3 to assess its accuracy, coherence, and fluency.
4. Evaluating the performance of GPT-3 on different benchmarks, such as GLUE and SuperGLUE.
5. Investigating the ethical considerations and potential biases of large language models.
6. Proposing solutions and best practices for the responsible use of these models.

The project will require proficiency in natural language processing, Python programming, and machine learning, as well as strong analytical and critical thinking skills. The outcomes of the project will help shed light on the potential of large language models and their implications for the future of natural language processing.

## 13. Personal Projects

CS	SE	GD	Cy
----	----	----	----

### Summary

I'm happy to discuss and support personal project ideas provided they match my areas of interest listed above and demonstrate a new set of skills being learnt by a student. Please arrange to chat about these with me before submitting in your Project Preferences Form.

**Areas of Interest:**

- Computer Graphics
- Serious Games
- Autonomous cars
- Medical Imaging
- AI assistants

## 1. Using AI assistant for marketing purposes

CS	SE	GD	Cy
----	----	----	----

Creating an application that automatically posts on social networks using ChatGPT involves several key steps. First, define the scope of your application, including which social media platforms you want to support and the types of posts you want to automate. Next, obtain the necessary API access from OpenAI for using ChatGPT and from the social media platforms for posting content. Design your workflow by integrating ChatGPT to generate content, and implement features for scheduling and posting. Develop the backend to handle ChatGPT integration and social media APIs, and create a user interface. Ensure compliance with platform guidelines and implement error handling. After developing the application, thoroughly test it and deploy it on a suitable cloud service. Security and data privacy should be prioritized by securing API keys and protecting user data. Finally, set up monitoring for performance tracking and plan for regular updates to accommodate API changes and enhance functionality.

## 2. Facial expression recognition

CS	SE	GD	Cy
----	----	----	----

Facial expression recognition is an important part of our lives. The face and its expressions are a window to the soul—if you know how to read them. The good news is that we can tell a lot about someone just by looking at their face. Emotions help us communicate with others, for example, when we feel sad and need help. They can also help us act quickly in important situations. For instance, when you're about to cross the street and see a car approaching quickly, fear will make you jump back onto the curb. A facial expression recognition system can be used in many applications. It can be used to study or analyze human feelings. Many companies are implementing facial expression recognition systems into their applications to study the emotional levels of their employees. Some gaming companies use facial recognition systems to record the satisfaction levels of players during gameplay.

In this project, the student will design a method for emotion or facial expression recognition and implement it as a software application using Windows platform, with HD camera resolution and the libraries OpenCV, TensorFlow, and Keras. Program a model and use the trained model to recognize a person's facial expression from a real-time video stream. The FER2013 dataset will

be used to train the model using a convolutional neural network (CNN) similar to VGG. Create a set of videos containing various facial expressions. Evaluate and categorize them manually without using software. Then, have the application evaluate and categorize these videos. Compare both categorizations and assess the accuracy and efficiency of the developed application. Programming language: Python.

### 3. Visual counter of repetitive exercises in the gym

CS	SE	GD	Cy
----	----	----	----

The main objective of this project is to create a software application on the Windows platform for a visual counter of repetitive exercises in the gym (e.g., push-ups, pull-ups). The student must use visual information from a HD camera (video stream) connected to the Windows to determine the relative coordinates of parts of the human body (e.g., shoulders and elbows) and estimate their position accordingly. The student will then compare the coordinates of the shoulders with the coordinates of the elbows. For instance, if the shoulders are at the same level as or below the elbows, it will count as a push-up. Similarly, the number of pull-ups can be observed and counted. The main task for the student will be to study image processing from the camera, segment parts of the human body, and assign the corresponding skeleton to these parts. The student must implement the project in Python. Optionally, the student can connect a speaker to the Windows and program a voice countdown for the exercises. The result of the project will be a software application that detects body movements and counts repetitive exercises. At the end of the project, the student must evaluate the accuracy and efficiency of the software application, the correctness of the exercise counting, and compare different types of exercises.

## Ralph Ferneyhough

### Areas of Interest:

- Games Mechanics Programming
- Re-usable Games Components
- Procedural Generation of Game Content (“Roguelikes”)
- Logic / Puzzle / Factory Games
- Boardgames

*NOTE: All projects should be implemented using C++ and the Unreal Engine, as that is a critical skill to develop for the games industry.*

## 1. Design and implement a logic / puzzle game with an editor to submit levels

CS	SE	GD	Cy
----	----	----	----

*Main skills required: Programming (Heavy), Unreal (Heavy), Design (Medium)*

Many logic / puzzle games have relatively simple rulesets that are comparatively easy to code, but their success is then entirely dependent on the content available – i.e. how many levels or puzzles a player has to progress through, and a logical, incremental progression in difficulty.

To alleviate this, user content can provide an endless source of levels, but the creation of an in-game editor is then required, and, more importantly, one which is intuitive to use and can only be used to create puzzles which have an actual solution.

You will research, design and create a simple puzzle game, and then implement an in-game editor which the user can use to create levels for others to try and complete.

## 2. Create a procedural world generator

CS	SE	GD	Cy
----	----	----	----

*Main skills required: Programming (Heavy), Unreal (Medium), 3D modelling (Light to Medium)*

Many games nowadays rely on procedural generation to provide a (near-) infinite array of levels or worlds to explore by the player – indeed it is one way “art-light” teams can produce the content expected by the user.

This challenge is to create a 3D world generator. The scope of this “world” is for negotiation – one example might be a space station generator that could be used in a space flight-sim as somewhere a player can explore in full 6 degrees of freedom. Another example could be to generate a level suitable for a survival game to take place.

The generator should aim to combine simple components into interesting shapes and layouts, and ultimately feature both a seeding system (so the best layouts can be saved and replayed)

and a method to decide whether a generated level is suitably “playable” according to rules you will define for the game it is intended for.

### 3. Create an immersive, functional inventory and shop system in virtual or augmented reality

CS	SE	GD	Cy
----	----	----	----

*Main skills required: Programming (Medium to Heavy), Unreal (Heavy), Design / UI (Medium)*

Many games rely on an inventory system underlying the gameplay, and supporting components such as shop interfaces to buy/sell and even craft items. The rules for these are well established for traditional 2D user interfaces, but a 3D interface in VR or AR means new considerations must be understood.

Develop a re-usable system that can be used as an off-the-shelf component by other developers which could handle this problem, working from a data set that would contain such parameters as size, weight, value, 3D model, 2D icon etc., that a third-party designer or producer could adjust and see working in the system instantly.

### 4. Design and implement a fully realised text system for use in a game

CS	SE	GD	Cy
----	----	----	----

*Main skills required: Programming (Heavy), Unreal (Medium), Design (Medium)*

The job of a games mechanics programmer isn’t all about the playable parts of a game – somebody has to do just as essential work making sure the user experience is as usable as possible. Elements of Unity or Unreal do exist on which one can build simple text-based elements, but much more than this is needed to deliver the full-fat UX in AAA games.

You will research existing games, design a suitable framework, and deliver code which could be used as an “oven-ready” text and menu system for a game in Unity or Unreal. This can include methods for text justification, auto-splitting of lines, fitting text into a fixed size box, handling accented characters or icons within the strings (e.g. controller button icons), and handling foreign language translations seamlessly. Simple VFX such as sliding, fading, scaling or rotating text exposed to the programmer to use should be created too.

### 5. Create an educational game

CS	SE	GD	Cy
----	----	----	----

*Main skills required: Programming (Medium), Unreal (Medium), Design (Medium), 2D or 3D art (Light to Medium)*

Gamification of education is a popular call, so this project will involve looking for existing games that are used to help teach some topic to people, evaluating what components are needed to

make them successful, then designing and implementing your own idea. This needs to go beyond a simple quiz – the actions in the game must reflect the concept being taught.

Examples might be to teach a specific concept in physics, such as the reflection and refraction rules of light, or in chemistry the bonding of atoms into simple molecules.

## 6. Personal project idea

CS	SE	GD	Cy
----	----	----	----

I am happy to discuss personal project ideas before you fill in the form, especially in my areas of interest, but they must include a sizeable programming component within your implementation.



## Graham Logan

### Areas of Interest:

- See topics below

## 1. Computing in Schools Project

CS	SE	GD	Cy
----	----	----	----

The design and implementation of materials for the teaching of computing in schools, with an emphasis upon difficult concepts such as Object Orientation and the Fetch-Execute Cycle.

## 2. Big Data Analysis for Business

CS	SE	GD	Cy
----	----	----	----

Examine the concept of Big Data in Business, with special emphasis upon infrastructure, management and mining techniques. Use appropriate software to demonstrate how such techniques are applied.

## 3. 3 Optical Character Recognition for Mobile technology

CS	SE	GD	Cy
----	----	----	----

Design a piece of software than can identify handwritten characters and symbols.

## 4. Newton's Laws of Motion in Games Technology

CS	SE	GD	Cy
----	----	----	----

Describe and explain how appropriate laws of physics (i.e. Newton) are applied to Games technology.

## 5. Cloud Computing

CS	SE	GD	Cy
----	----	----	----

Examine the advantages of cloud computing in modern business, as well as the possible problems and issues that such technology can cause. Such issues can include:

- Security and integrity
- Privacy in multi-tenancy clouds
- Virtualisation
- Data recovery and backup
- Data segregation and recovery
- Secure cloud architecture

- Cloud cryptography
- Cloud access control and key management
- Integrity assurance for data outsourcing
- Trusted computing technology
- Failure detection and prediction
- Secure data management within and across data centres
- Availability, recovery and auditing
- Secure computation outsourcing

## Andrew Muncey

### Areas of Interest:

- Collection and exploitation of data from mobile and connected devices (such as wearables)
- Development of systems to automate the assessment (and/or teaching of) of programming (for example programmatically generated quizzes)
- Development of systems concerned with the internet of things (connected devices)

## 1. Android app to replace chords sheets for musicians

CS	SE	GD	Cy
----	----	----	----

Many musicians, perhaps more so those playing contemporary popular music, often use printed sheets which contain the words and chords to music, often copied from guitar chord websites ([example song here](#)). Apps already exist to provide an on-screen version of chord sheets, with features such as automatic scrolling and transposition. A popular app for such purposes is the iOS only [OnSong](#).

The project would develop the core features of such an application (catalogue of songs, ability to create set lists, displaying of chords) as an Android app. There is the potential for collaboration on this, either by making the project open source, and/or for multiple students to work on the same project. Similarly, there is scope to introduce a wide variety of extra features (such as importing file from other software).

## 2. Programming a Programming Game

CS	SE	GD	Cy
----	----	----	----

Programming is a skill that requires practise, but learners sometimes fail to put in the required time to become proficient. This project aims to gamify the process of learning programming, ideally learning the Java language, but other languages will be considered (the language the game is built in can be chosen by the student). There is scope to determine the nature of the game, for example it could be a single or multiplayer, 2D, 3D, web-based or even command line driven. Ideally the game will be aimed at novice programmers.

## 3. Dissertation topic creation, presentation, selection and allocation system

CS	SE	GD	Cy
----	----	----	----

In advance of the start of each academic year staff update the document you are reading, which is then provided to students. Students select preferences for their topics and then we meet and allocate student projects to staff with the aim that all students will be allocated a topic and supervisor of their preference (which is not always possible), and a secondary aim of

allowing that students can keep a topic if a preferred supervisor does not have capacity. The number of students per staff member varies based on other work commitments. This project would seek to automate as much as possible of the process – enabling staff to update projects using a web interface, enabling students to filter by programme of study, and tutor, and by attempting to allocate projects to staff members based on student preferences and staff availability, whilst optimising the number of students who get one of their preferences. The system should also be able to handle students requesting their own topic. There is scope to extend the project, for example by showing live popularity, extending the system to handle master’s students dissertations (some projects will only be suitable for Masters students, some only for undergraduate, some for both, and some staff cannot supervise masters students).

#### 4. Academic staff workload modelling (and management) system

CS	SE	GD	Cy
----	----	----	----

The workload of an academic is hard to model, but it is key that managers can ensure equitable distribution of workload across staff. Typically, workload is modelled on an annual basis. Whilst various systems exist to manage staff workload, they can be hard to use, and typically managers resort to using a spreadsheet. Some aspects of workload are calculated based on other variables, for example, timetabled teaching hours will require additional work hours for preparation, but the preparation is not duplicated if staff teach the same thing to multiple sets of students. Time allocated for marking work is a function of the number of students on a module, the level of the module and the number of credits for the module (and the staff member might share a module). This project would seek to develop either a system, or part of a system which can help staff and managers record workload.

Areas where the project could be expanded include accommodating activities that span multiple academic years (such as MSc dissertation supervision for students starting mid-year), allowing for the reallocation of workload to other staff, ensuring all known workload is allocated (for example, that the teaching of modules is not left unallocated), handling staff sickness or other periods of absence, or even distribution of workload across the year.

#### 5. Personal Project Ideas

CS	SE	GD	Cy
----	----	----	----

I’m also happy to discuss and support personal project ideas as long as they are within my areas of interest listed above and demonstrate a new set of skills being learnt by a student. Please discuss these with me (via an office hour booking) before putting in your Project Preferences Form.

## Dr Mike Morgan

### Areas of Interest:

- See topics below

## 1. How hard can it possibly be to pack a collection of boxes (the Bin Packing Problem)?

CS	SE	GD	Cy
----	----	----	----

The bin packing problem is easy to describe but notoriously difficult to solve. Given a collection of items with varying dimensions, arrange them into the smallest possible number of equally sized containers (known as bins). This has very important applications in environmental logistics. Fewer containers mean reduced fuel consumption and therefore reduced carbon emissions and available capacity is commonly underused throughout the industry. Your task is to design and implement heuristics to improve resource allocation.

## 2. Automation of CISCO device configuration with Python

CS	SE	GD	Cy
----	----	----	----

Incremental SDN (Software Defined Networking) is becoming an increasingly important technique in network management. Whilst many small to medium enterprise networks are not yet fully programmable, it is possible to take a step in that direction by scripting the configuration of legacy (e.g. CISCO) devices. You are to investigate and implement techniques to automate the configuration of some aspect of networking (e.g. routing, access control, IPv4 to IPv6 transition).

## 3. Develop a simple physics engine in C++

CS	SE	GD	Cy
----	----	----	----

Write a physics engine in C++. It is recommended that you keep it simple, e.g. stick to Newtonian mechanics with a bit of Hooke's law for springs/elastic.

## 4. Automated GUI interaction in Python

CS	SE	GD	Cy
----	----	----	----

Python currently has some basic GUI automation tools (such as pyautogui) which could have their functionality enhanced in combination with image processing tools (such as opencv) for enhanced image and text recognition. Your task is to build an intuitive python library to interact with simple GUIs such as web forms on behalf of a user and test it on one or more suitable systems. The software would be intended for use in circumstances where a web system does not have an API, restful URLs or any other meaningful structure for a programmer to exploit.

## 5. Implement a simple games server using C++ and boost

CS	SE	GD	Cy
----	----	----	----

We all love boost. Who can deny it? How about putting it through its paces and writing a network games server (and client) with it? Any games it hosts should be relatively simple (e.g. pontoon) as the networking is intended to be the technical focus of this project.

1. <Project Title>

CS	SE	GD	Cy
----	----	----	----

<Details of project>

2. <Project Title>

CS	SE	GD	Cy
----	----	----	----

<Details of project>

## 1. Automation of malware analysis

CS	SE	GD	Cy
----	----	----	----

There are various services that will allow you to upload either a file or URL and they will be scanned for presence of malware. E.g. Virustotal.com and Hybrid Analysis. In the case of a file, they will scan it against all the AV products they have access to, with the latest definitions.

The websites will also do automated analysis of the file, such as pulling out known IP addresses, strings, and APIs in use. Paid for access will do better analysis such as linking IPs and pieces of the code to known attacks and other malware in the wild.

### Proposal

Create an open-source version of some of the features given, so that individuals and companies can provide both a file and a URL and the website will do automation of the input. For files, this could be automatically detecting the type of file, whether it be a native PE executable for Windows, or whether it's packed in some way. If it is packed, it will try unpack it, then scan it.

The scanning itself can be large in scope. Bypassing AV is rather trivial so the student could write their own algorithms. Perhaps run the file in a sandboxed environment and monitor what happens on it, in the way that hybrid analysis does.

### Skills required

Programming (heavy) – The student should create a backend which takes an input in the form of a file or URL. The backend then uses a bunch of programs (existing and student created) which will scan the input and then perform various tasks. Such as unpacking it, running in a sandbox if its custom packed, analysing the file using a disassembler/decompiler and figuring out what API's are in use. Detect if the file is malicious and report back to the user on the frontend website.

## 2. Penetration Testing tool for ethical hackers

CS	SE	GD	Cy
----	----	----	----

For the use in sanctioned penetration tests. For many ethical hackers, the problem when performing a penetration test is that the same steps have to be performed. Is there a way to automate some of this to save you time?



## Proposal

Using Kali Linux, or a Linux distro of your choice, you are to create a program that takes as input a list of IPs, and it will automate the recon, vulnerability and exploitation of IP addresses.

The main implementation part here is the design of various scripts that will take the output from various programs then act on it. E.g. Imagine you are given an IP range which you are to try gain access to. You would need to give that IP range to a program to scan for hosts which are actually there and running. You then need to probe these hosts for the OS and programs that are running on there. From here you might decide to try find vulnerabilities in these services. This is the simplest part of it, but you can go into much more depth on this, using various algorithms to decide how to act. E.g. Do you then try exploit it and give the pen tester back a success message with information on how to proceed? From the end user's point of view, they just want to get some output so they can decide what to try next. Clearly this is the simplest case you could have. In a real pen test, it will be much harder than this and will require the pen tester to follow up on other methods. The scope here is large, and allows the student to decide on what else happens in the entire process, including lateral movement across the network.

## Skills required

Programming – Heavy. Knowledge of the cyber kill chain process – Heavy.

## 3. Incident Response SIEM

CS	SE	GD	Cy
----	----	----	----

For blue teams doing Incident Response, there is often a need to purchase a SIEM (Security information and event management) tool to sift through the noise of what's happening on the network. A SIEM will allow you to plug in various devices on the network, e.g. Firewall, Switch, PC? It will pull the logs off these and try to piece together what's happening on the network in a normal session, and detect anomalies when an attack is taking place.

## Proposal

Make an open source SIEM which will read event logs from the PC, parse them together and then like a SIEM would do, you are to correlate across the events to see if there is anything peculiar. For example, if an attacker were performing lateral movement, you would see a connection into the network once to a box, but you would see a lot of movement from this box to other boxes in the network. If this were being performed by someone who would never connect to another box, it's a sign of lateral movement.

## Skills

Network design and implementation – Heavy. Programming – Heavy.

## Dr Toyosi Oyinloye

### Areas of Interest:

- Software Exploitation
- Software Protection Measures
- Cyber-User Awareness for Social Engineering
- Cyber-Reconnaissance
- Cybersecurity-Data Visualisation

## 1. Control Flow Integrity (CFI) for Securing Vulnerable x86 Binaries

CS	SE	GD	Cy
----	----	----	----

Control Flow Integrity (CFI) in protecting vulnerable programs from Control Flow Hijacks involves the use of software or hardware mechanisms to restrict (monitor) and enforce the flow of execution in a program to the intended path.

### Proposal

A software-based protection to be developed, separate from the vulnerable program and inserted at compilation time or activated at runtime. The first step would be to carry out extensive software exploitation to determine the critical spots in the execution path and then generate Control Flow Graph (CFG) to use in guiding the monitoring process. Buffer and Stack overflow, UAF & DF, are vulnerabilities that attackers use in exploiting applications. Existing codes have also been used by attackers as gadgets chained together in Return Oriented Programming (ROP) attacks. An ideal protection here would consider any/all of these factors towards securing vulnerable targets. Here are possible options that could be explored in this study.

#### Option 1:

Write assembly code or programs in C or C++ with zero user-interaction to manage the enforcement of CFI without any further interference. Protection can be set for binaries compiled and run on X86 platform or any other CPUs.

#### Option 2:

Create a framework which would use either hardware or software or a combination of the two as solution in combatting CFH.

### Skills required

- For software-based solutions: Programming (Heavy, Assembly code, C/C++)
- For hardware-based solutions: Ability to identify or build relevant/effective hardware materials (Heavy)
- Software Engineering (Heavy)

## 2. Inter-process Communication for Execution Monitoring

CS	SE	GD	Cy
----	----	----	----

Software-based protection measures can be implemented, in-line, or added at compile time, however, attacks are often executed at runtime. In-line protections could be migrated into non-vulnerable processes so that monitoring can be achieved by constantly checking of how a process is being executed, particularly in comparison with the CFG of each process. This is beyond the normal task management done by operating system. Inter-process communication (IPC) can be used to establish parallel communication between specified processes to set up a monitoring system. Possible means of IPC are through Inter-Process Signalling (IPS), In-kernel monitoring, or Hypervisor monitoring.

### Proposal

Use a program, with zero user-input to monitor a suite of vulnerable programs through the mechanism of IPC. This could also be achieved in collaboration with the kernel or a hypervisor. IPS is very effective when established through shared memory. It is important to include atomic operations to prevent race conditions. **\*\*Don't create another problem while trying to solve one.** Use either C or C++ to create shared memory and effectively monitor execution flow to detect errors, hijack, or deviation.

### Skills required

- Programming (Heavy, Assembly code, C/C++)
- Software Engineering (Heavy)

## 3. Cyber-User Awareness System to Combat Social Engineering Attacks

CS	SE	GD	Cy
----	----	----	----

In recent times, social engineering has been on the rise in the cyber-attack chain, in addition to phishing. As cyber-attack techniques continue to evolve, the user awareness system that corresponds with prevalent attacks must be deployed. This may include relevant and effective awareness training tools in form of GUIs. These includes gamification, awareness training systems, etc.

### Proposal

The focus here is on Social Engineering. Designing and Implementation of Cyber-User Awareness system/tool. Consider cost effectiveness, reliability, and sustainability. Expected product from this research is Game/App.

### Skills required

- Social Engineering/Cyber Psychology Research (Heavy)
- Design and prototyping (Heavy, involves programming)

## 4. OSINT/Cyber-Reconnaissance for identifying and Mitigating Attack Vectors

CS	SE	GD	Cy
----	----	----	----

OSINT can be used in information gathering and then for further cyber-attack by applying other attack vectors. Organisations can apply proactive approach in securing their web-based information.

### Proposal

Design an effective tool/technique for ethical Cyber reconnaissance to give organisations an advantage over attackers. Begin by identifying risky exposures of information and how attackers could use information gathered at the detriment of the target. A proper design would involve some automated information gathering and matching the information with existing attack vectors.

Option 1:

Design an application to achieve the goal.

Option 2:

Create a framework that effectively simulates how OSINT can be applied to protect data/information from any of the prevalent cyber-attack vectors.

### Skills required

- Social Engineering/Cyber Psychology Research (Heavy)
- Database management (Medium)
- Design and prototyping (Heavy, involves programming)

## 5. Cybersecurity-Data Visualisation for Incident Response

CS	SE	GD	Cy
----	----	----	----

Data visualization is relevant in cybersecurity as it provides an easy way of viewing data gathered from various sources. This could include data from activity logs, security logs, alerts, notifications, etc. When there are lots of data from various sources, they might appear as a clutter of jargons to the average user, whereas vital information is hidden in the clutter. Some clarity can reveal valuable information while further refinement can make the data/information more meaningful to investigators. According to Fry (2007) once data has been acquired, other steps to processing the data for visualisation are:

1. Parse the data
2. Filter- out unnecessary elements
3. Arrange the data statistically or mathematically
4. Use a visual model to represent the data
5. Refine the model to improve clarity





