Computer Science, Maths & Computer Science, Computer Science & Philosophy
Part A

Group Design Practical

Briefing Document Hilary Term 2020

All second-year undergraduates reading Computer Science, Mathematics and Computer Science, or Computer Science and Philosophy take part in a group design exercise in their second year.

Goal

The goal of the group design practical is to give you, the student, a chance to understand and learn the skills required to succeed in delivering a real-world project, not just writing a program.

In addition to using the skills learnt during the four core programming courses to actually code, to succeed in this project you'll need to

- engage with your customer to understand what their needs are
- agree with the customer what your solution will and won't do in order to solve their needs (and update/negotiate changes with them as necessary)
- self-organise with your team to distribute work *and* make sure the whole solution works together
- plan your time to deliver what the customer needs when they need it
- provide the solution to the customer in a form
    - they can easily deploy (installation, configuration)
    - they can easily use (documentation, briefings, maintainability, good diagnostics)
    - they won't break (careful testing).
    - and considers the ethical and societal impacts of the solution.

These are all solid basics that any tech company or research group expects of any good contributor – this is your chance to develop these skills as part of your degree.

Key Dates

Briefing Meeting	Wednesday 29th January, 2pm	Week 2
Deadline to fix team members	Monday 3rd February, 12pm	Week 3
Responsible Innovation Seminar	Wednesday 5th February, 2pm	Week 3
Deadline to fix project choice	Friday 7th February,12pm	Week 3
Briefing on specific projects	Wednesday 12th February, 2pm	Week 4
Project Co-ordination Seminar	Wednesday 19th February, 2pm	Week 5
Project Management Seminar	Wednesday 26th February, 2pm	Week 6
First progress meeting with customer/academic sponsor – agree customer requirements and project specification and plan	Week 6 - Time and day to be confirmed by team and sponsor	Week 6
Software version control seminar	Wednesday 4th March, 2pm	Week 7
Software testing seminar	Wednesday 10th March, 2pm	Week 8
Second progress meeting with customer/academic sponsor – Progress report, initial implementation and test results	Week 8 - Time and day to be confirmed by team and sponsor	Week 8
Third progress meeting with sponsor – delivery and final report	Week 2 - Time and day to be confirmed by team and sponsor	Week 2
Presentation to Department and industry guests	Wednesday 13th May	Week 3

Computer Science, Maths & Computer Science, Computer Science & Philosophy Part A

Organisation

Students are allocated to teams of around 6 people at the Briefing Meeting. Swaps between teams are allowed until the deadline to fix team members – to confirm a swap both students involved should email Oxfordgroupprojects@gmail.com.

Each team will then need to choose 3 projects from the list of available projects and email their first, second and third choices to Oxfordgroupprojects@gmail.com, with a copy to all team members. Each team will be notified which project they have been allocated, and who is sponsoring that project. Further information about each project, and the materials and equipment available, will then be given to the relevant teams.

The team's sponsor acts as both:

- The (friendly!) customer with whom the team engage to understand any detailed requirements for the project and to whom the team deliver their finished product.

- A mentor, ready to help the team deliver their project.

It is the team's responsibility to drive the project, requesting help from the sponsor as necessary.

As part of the project, each team *must* do the following.

Arrange an initial planning meeting with their sponsor in Week 6 of Hilary Term. At this meeting the team should agree what they will be delivering, when, to the customer and agree with their mentor a plan for how they'll achieve that.

Arrange a progress report meeting with their sponsor in Week 8 of Hilary Term. At this meeting, they'll present a progress report to the customer on their module implementations, test plans and test results. If the team and their sponsor agree, the team may instead provide a brief written report on their progress by the end of Hilary Term.

This meeting is also a good point to negotiate changes to the scope or timeline for what they are delivering (if for example it turns out that their project is significantly harder than originally expected, they can negotiate to reduce the scope). It is absolutely fine for teams to agree with the customer to change the scope of what they're delivering as long as the customer has reasonable warning (while it is absolutely not fine for example for the team to propose a reduction in scope 24 hours before the final delivery date!)

Arrange a delivery meeting with their sponsor in Week 2 of Trinity Term. At this meeting they deliver the agreed project to the customer, including the product, installation instructions, documentation, etc. as previously agreed. Each student will also deliver to the sponsor a one-page summary of their individual contribution.

Teams are encouraged to engage with their customer (sponsor) more frequently than these three mandatory meetings.

In Week 3 of Trinity Term each team will present their work to students, members of the Department and industry guests. This will take the form of a demonstration session, followed by a seminar where groups will take turns to describe their projects in no more than 7 minutes each. A panel of judges will attend the demonstration session and seminar to judge the best groups, and cash prizes will be awarded.

Computer Science, Maths & Computer Science, Computer Science & Philosophy Part A

Overview and Marking

The group practical exercise is designed to take 20-30 hours of student time, mainly during Hilary Term (so a team of 6 delivers a project of 120-180 hours of work). The contribution of each student will be based on the group report, demonstration, and individual contribution sheet. This mark will count for approximately one third of the practical marks for the second year. Students must pass the group practical in order to pass the practical component of the course. The group design exercise will be marked on a scale of S-, S(pass), S, S+. These marks should be converted to a numerical mark using the following scale:

S+	100
S	60
S (pass)	40
S-	20

This exercise gives students experience of working against rigid deadlines, with a team of colleagues not of their own choosing, using externally supplied tools to undertake a fixed project. This will give some idea of the problems encountered in normal professional practice. Groups will be expected to exhibit professional skills in design, quality and management. Specifically, they will have to show that the work has been carefully planned, that components and systems have been properly tested, and that members of the group have cooperated effectively. The review meetings with sponsors provide an opportunity to monitor group progress and for general discussion, but sponsors will not be expected to provide technical advice or resolve technical issues. The responsibility for organising and completing the work lies squarely with the group members.

Our intention is that all group projects should be successful, and all students are expected to contribute to their own project in accordance with the timetable. Groups often encounter problems, and students are expected to manage these as they arise. However, if serious problems are encountered, such that members of the group are unable to resolve them, students should contact the project organizers immediately rather than waiting until a deadline is missed. The contact address for the group project organizers is: oxfordgroupprojects@gmail.com.

Computer Science, Maths & Computer Science, Computer Science & Philosophy
Part A

Group Design Practical

Deliverables

Each part of the project gives rise to a set of deliverables *which must be given to the project sponsors at the review meetings.* The project sponsors will forward all group and personal reports to the examiners for inclusion in the practical work portfolio of all group members.

Specification and plan

Project topics are presented in the form of an outline design brief. Part of the work is to undertake a proper requirements analysis for the chosen project. However, it is important not to develop an over-elaborate specification which commits the group to more work than is necessary. The first major task is therefore to turn a relatively open and informal design brief into a more detailed project specification, setting out the major components of the system you will produce, what documentation you'll deliver, what diagnostics you'll provide, etc. Once this is agreed with your sponsor, you must produce a project plan which sets out who will do what, the time needed for developing and testing each module, dependencies between modules, etc.

When planning and executing the project, it is important to work to a budget. The entire project should take no more than 40 hours per team member, and records should be kept of time invested. Groups should set realistic targets and achieve them; there will be no additional credit for over-elaborate projects, or for individuals who offer or do more than is required. Each member of the group is expected to gain experience of programming in the course of the project – this may possibly involve test harnesses or scripts, data conversion utilities, a tutorial system, external interfaces, demonstration examples, installation packages, or other code and associated materials as appropriate to the project and the individuals in the team.

Module implementation and testing

The next task is to implement the components and test them. This is likely to require the construction of special test harnesses for separate classes. The deliverable for this task is a written ***progress report*** describing the testing procedures and results.

Project delivery and report

The final task is to piece the whole system together, test it and ensure that it is adequately documented. At the final meeting, the group must:

- Demonstrate the solution to the sponsor.
- Deliver the solution and associated documentation in a format that the sponsor can use.
- Submit a brief (approximately 4 pages long) group report describing the project aims, outcomes, successes and failures, and any lessons learnt. This report should include a short (approximately half a page) discussion of responsible innovation, exploring (i) the ethical issues you needed to consider during your design and development activities and (ii) the potential societal impacts (both positive and negative, intended and unintended) of the solution that you have developed.

Each team member must also submit a one-page summary of their own individual contribution.

Computer Science, Maths & Computer Science, Computer Science & Philosophy
Part A

Project A: Virtual Reality Finance

[Project in collaboration with **Oxford Asset Management]**

We are broadly interested in investigating potential applications of VR in Finance. The era of Big Data has brought about an explosion of information about publicly-traded companies, often in the form of high-dimensional datasets. They might be timeseries (e.g. real-time stock quotes), relational (e.g. company industry sectors), or network (e.g. company supply chain data). By leveraging the natural human ability to spot patterns and anomalies, we believe sophisticated audio-visualisations of feeds such as order book flow and portfolio evolution have the potential to find data errors, uncover low-level correlations, and pick out regime changes.

This project will involve surveying the current state-of-the-art in VR, big data, and machine learning. You will implement visualisations that will immerse users in real-world financial datasets, allowing them to explore and interact with the data from within. This should provide a platform for testing the hypothesis that such an experience can aid the human understanding of large, complex systems.

Project B: Tradeteq: Adversarial attacks and defences for credit scoring systems

[Project in collaboration with **Tradeteq]**

Many financial decision making systems are increasingly reliant on machine learning. Credit scoring for companies and individuals in one of the common applications. Ample incentives for unscrupulous market participants to game those systems create the need for robust credit scoring resilient to realistic attacks.

In this project, you will be playing the role of a “white hat” hacker of a financial machine learning system. We will provide you with a large pre-processed UK limited company dataset. You will calibrate a simple machine learning model to predict company status transitions for these companies and will then develop adversarial attacks on this model. Some company features are much more prone to manipulation than others. The attacks will seek to improve the credit score by changing those features. The attacks may be developed while using the target model as an oracle or with full access to target model coefficients. After creating a successful attack, you will need to investigate possible defences against this attack.

Project C: Web-based Editors simulation of C/C++ code and Python

[Project in collaboration with **Micro:bit]**

Teaching people to code with blocks is great, but we ultimately want them to learn to code with text-based languages. When moving to text from block-based coding, where syntax errors and typos are impossible, giving very fast feedback to students when they make these errors is essential. Sadly, we currently do not do this in our Python editor! Therefore when the micro:bit is running MicroPython, we want to be able to run their code immediately in the browser, without needing to run it on the device to find out it’s broken.

We’ve got a prototype “Python Simulator” that builds the MicroPython C/C++ code using ‘emscripten’ so that it can run in the browser. This has resulted in a proof of concept for simulating Python… we need a team that can understand the requirements of users to and work across many levels of the software stack: C/C++, compilers, javascript, UI/UX in order to deliver an awesome simulator. There’s scope to turn this project into a more generic simulator for C/C++ projects on the micro:bit, perhaps by attaching a compilation backend or running firmata on ht micro:bit.

Computer Science, Maths & Computer Science, Computer Science & Philosophy Part A

Project D: Automatic Tagging of F1 Photographs.

[Project in collaboration with **Ferrari Formula 1]**

Over the course of F1 race events, thousands of photographs are taken of all the F1 cars (eg. see here). As part of the competitive nature of the sport, there is value in looking at these photos to identify alternative ways of solving the same design problems, highlight upgrades and ensuring that all participants comply with the corresponding technical regulations. In order to allow an efficient use of the engineering time, the objective of this project is to develop an algorithm capable of automatically do the following to each photo:

1. Assign a tag with the name of the main car component/s visible in the photo from a set of predefined list (I.e.: Front Wing)
2. Cropping the image around the corresponding component.
3. Assigning an 'orientation' tag to define the approximate yaw position of the camera relative to the car from a set of predefined orientations: 0° (front), 45°, 90°(side), 135°, 180°(rear)
4. If possible, a 'certainty metric' (I.e. 70% chances of this photo showing a front wing) should be associated to each tag.

The photo tagging is to be stored in a simple MS Access database. Students participating in this project will be required to sign a non-disclosure agreement (NDA).

Project E: Micro:Bit Educational Tools for Micro:bit

[Project in collaboration with **Micro:bit]**

The micro:bit is an ARM-based embedded development board intended for STEM education (https://microbit.org). To date over 4.5 million micro:bit devices have been manufactured. In this project, you will work with the Micro:bit Educational Foundation to extend the capabilities of this device.

Potential projects include (i) developing interaction mechanisms to facilitate local large-scale multi-player games and simulations (e.g. modeling virus contagion throughout a classroom or synchronisation of fireflies) using the existing I/O and peer-to-peer networking capabilities of the micro:bit board, (ii) a data-logging application that allows micro:bit devices to collect experimental data, exposes that data to a webpage through a WebUSB interface, and presents useful data analysis and plotting tools, or (ii) deploying small convolution neural networks on the device to perform image recognition tasks using a low resolution camera made from an optical mouse sensor. You will be encouraged to explore other ideas of your own. The focus is to provide a complete end-to- end solution, balancing the needs of both students and instructors, that can ultimately be incorporated into publicly available micro:bit educational resources.

Project F: Explainable Machine Learning in Trade Surveillance

[Project in collaboration with **Morgan Stanley]**

There has been a surge of research in Explainable Artificial Intelligence (XAI), partly driven by industries where AI/ML algorithms are being used to make legal, medical or financial decisions and explanations of the inner workings of the models are required by regulations (GDPR Right to Explanation). We investigate XAI in the area of trade surveillance where algorithms are used to detect illegal market manipulation attempts.

The task is to build a ML algorithm that is explainable for predicting trading pattern using publicly available data. An abundance of generic explanation methods is available per model type (linear, tree-based, non-linear/DNN, model-agnostic) with different notions of explanations. We propose to put the explanation methods in context by adapting them to the specific domain of trade surveillance. Further improvements could be producing explanations understandable by non-ML-experts or enhancing the 'quality' of the explanations. Specific topic can be decided at the first meeting with the students and we are also open for suggestions by the team. The aim of the project is to produce a working MVP and potentially as a basis for a short paper. More details and resources are available on request.

Project G: Emergent behaviour from AEAs

[Project in collaboration with Fetch.ai]

AEAs encapsulate the functionalities of an autonomous software agent. However, it is up to the developer of an AEA to decide how intelligent their AEA is going to be.

In this project, you are required to create a collection of (almost) zero-intelligence AEAs who achieve tasks which a) none of the agents in isolation could achieve, and b) it would be extremely difficult to develop a single sophisticated, (super) intelligent AEA that could achieve those tasks.

The project participants must quantify the difficulty of solving tasks using a swarm of zero-intelligence AEAs versus a single sophisticated AEA, and make suggestions for the AEA architecture to follow, if one is to implement either of the two approaches, in general settings.

Project H: User preference representation and elicitation for autonomous economic agents [Project in collaboration with **Fetch.ai]**

Autonomous Economic Agents (AEAs) can represent humans in executing their narrowly defined plans and objectives. For instance, an AEA might purchase a number of public transport tickets to get its user from A to B. In order to achieve this to the user's satisfaction, the AEA needs to be aware of the user's relevant preferences. This means the AEA needs a formal representation to capture its user's preferences and an associated mechanism for eliciting preferences from its user.

This project is centered around creating a formal preference representation for a well defined use case as well as a tool for eliciting such preferences. The use case can be specified by the project participants after discussion with the Fetch.ai mentor.

To help with how the preference representation problem could be approached, project participants could refer to conditional preference networks (cp-nets).

Project I: Automated Real-time Global Events Data Collator and Persister

[Project in collaboration with **APEX:E3**]

APEX:E3 has created a global event based financial instrument backtesting framework, where users can answer questions like:

What impact does a hurricane hitting the American Eastern coastline have on insurance stocks? Or How did the victory of Boris Johnson affect FX rates and Banking stocks?

The events are stored as time lines and our users can backtest trading strategies around these timelines e.g. Buy GBP/EUR and GBP/USD 1 month before the election and sell 2 months after the election. The idea is to create an automated real-time global events data collator and persister, where the following type of events are consumed from publicly available sources on the internet then classified, tagged by sentiment and persisted to a database or index for future querying:

1. Geo Political - e.g. US Trade war timeline, Brexit timeline, Oil tanker issues, Trump tweets
2. Financial - e.g. Companies earnings report timeline, key company announcements, IPOs, mergers & acquisitions
3. Sports - e.g. European / US / Asian soccer/baseball/cricket/Formula 1 teams that are listed on stock exchanges or have sponsors that are listed on stock exchanges. Example events include premier league match results, Formula 1 race wins/losses
4. Extreme Environmental Events - e.g. hurricanes, earthquakes, tsunamis, droughts, landslides
5. Entertainment - Film releases, Game launches, Stadium events like music concerts and boxing matches, Music releases

The outcome of this project is to have a functioning prototype which automatically collates and persists events as described above, with means for further extension. Bonus points will be awarded if the prototype can also detect fake news/tweets/content. Historical data can be provided as required for this project.

Project J: Earth Trust

[Project in collaboration **with Earth Trust**]

Trees are the lungs of the planet absorbing carbon dioxide and sequestering the carbon. However, the extent and timescale of predicted climate change will impact trees and woodlands extensively.

Earth Trust, an environmental learning charity in South Oxfordshire, owns Paradise Wood, a national research woodland. Paradise Wood contains the largest genetic collection of hardwood forestry trials in Britain. They have raw data on the growth of different tree species in the woods and what to use this data to show the importance of the woodland and the impact of climate change of different trees.

The task for this project is to build a prototype system to correlate this data with metadata from the Met Office to visualise which trees have grown best despite the more extreme weather events we’ve had in the last 20 years. They are looking for an output with a wow factor - the system will need to be able to produce an appealing front-end visualisation that engages people in the importance of the Paradise Wood and why it is vital to build reliance into our woodlands for the future.

Project K: Information Representation

[Project in collaboration **with Fetch.ai**]

Autonomous Economic Agents (AEAs) interact with each other to generate economic value for their owners. In this context, AEAs have several application areas, including but not limited to a) agents representing hardware devices, b) agents as API interfaces for interaction between different digital ecosystems (e.g. legacy vs current tech, web2.0 and web3.0 etc.), c) agents representing users and acting as their “digital assistants”, and d) agents brokering access to data. The different applications require agents to represent different types of information (e.g. services) so they can be registered and queried in search engines.

This project is centered around proposing one or several models for information/knowledge representation suitable to the machine economy the AEAs inhabit. The project should discuss the tradeoff between structured and free form information representations as well as other tradeoffs encountered. It should also be discussed how the proposed model(s) lends itself to decentralization/distribution of the databases in which knowledge is stored.

Computer Science, Maths & Computer Science, Computer Science & Philosophy Part A

Project L: Finding Technical Talent

[Project in collaboration with Metaswitch]

A software telecoms company are looking for ways to attract and identify new technical talent. They're looking for some kind of game or puzzle that will run in a web browser, with twin underlying goals of (a) get lots of students interested in and applying to the company and (b) automatically flagging any particularly good looking candidates to the company to fast-track to interview. The company has some ideas for puzzles and games that might work, but are open to ideas and suggestions from the consultants (you!) – they're really looking for something with wow factor. There's no restriction on the set of languages/packages/back-end that you use.

Project M: Put Your Phone to Work

[Project in collaboration with G-Research]

When people browse social media sites on their phones for hours every day, most of the CPU power goes unused. The old desktop equivalent of this problem was the screensaver, which did little of value until it was co-opted for distributed computing projects such as SETI@home. Your task is to make a platform that can perform useful computation in the background on a large number of mobile phones, while the owners are on social media - or even while they are asleep. It will have to run cross-platform, perhaps using JavaScript, but must also give the appropriate incentives to users - will it drain batteries or incur network charges? If so, what kind of application would customers pay to run on such a platform? Would phone sensors offer any specific value? You need to demonstrate an end-to-end solution including servers, mobile clients and an example application, keeping in mind the security implications if either customers or phone owners try to cheat the system.

Project N: AI Racing Market

[Project in collaboration with G-Research]

As AI-controlled vehicles like Google's Self-Driving Car become more common, the person who gets to work fastest will be the one with the best algorithms. Your task is to create a competitive market in which members of the public can submit algorithms to see which is the best. You'll need to define a simple scripting language and API suitable for creating the entries. Users should be able to enter their script directly into an interactive game then see its performance in an actual real-time car race created with the Unity (https://unity3d.com/) graphics engine and physics model. The best entries should be stored and ranked in a leader board, with new players able to see existing code and tweak it for better performance. In future, this kind of algorithm market could be applied to other problem domains such as finance.