

1 L4 project staff guidance (2020-2021)

1.1 General

Level 4 individual projects run through both semesters of every Level 4 students' course. Each project is supervised by members of staff in the School. Project proposals need to be entered into the LTC system by 19th September. Students will bid on projects in Week 1 (Bidding closes 25th September) and project allocations will be published in week 2.

The individual project is worth 40 credits, or 33% of the standard 120 credit Level 4 programme.

Supervisors provide guidance throughout the duration project, via weekly meetings with their students. The final product is assessed via a dissertation, a short video presentation component, and an evaluation of student professional conduct.

1.2 Allocation

Academic staff each need to propose $2 \times (t_A - s_A)$ projects that they would be willing to supervise, where t_A is the 'Level 4 Project Supervisees' allocation from your WAM Work Allocation (2020-21) and s_A is the number of student-defined projects that you ***know*** you will be allocated.

Project proposals are entered on the LTC system <https://gaua.dcs.gla.ac.uk/studentprojects>
See the end of this document for example project proposals.

During the selection period students rank projects. Automatic allocation is performed, based on preferences and staff load-balancing. This assigns projects to supervisors, keeping the number of projects assigned to supervisors as equally balanced as possible. The allocation is **final** and will not be changed after the fact.

Allocation to students is done by algorithmic means at the start of Level 4. All projects must be proposed and in the system by 19th September, i.e., before the projects open to students at the start of Week 1. Reasonable adjustments to project titles after allocation are permitted before the project begins, if both supervisor and student agree.

1.2.1 Project allocation types:

Most projects follow the rules above. However, there is an option for students to define their own projects and agree supervision with a member of staff. The possible project types are:

Proposed: standard project model, as above, supervisors give a short outline, students rank projects and are allocated by algorithm.

Student-defined: student proposes a specific project and agrees supervision with a member of staff *before* project allocation phase.

The last type guarantee the allocation of a specific student to a specific supervisor. *They must be fully agreed and entered into the system **before** the project allocation starts to stop “sniping” of the allocation process.* Student-defined project types are exceptional, and most projects will be of the proposed type. Students must approach staff themselves and negotiate a project description for the “student-defined” projects. Matchmaking is not offered for student-defined projects. Supervisors are not obliged to take on self-defined projects.

2 Supervision and content

2.1 During Level 4 Project

There is an expectation that you have weekly meetings with each student, of typically half-hour duration. You may agree alternative arrangements with students (e.g. one hour every two weeks), but this should be by mutual consent and must not disadvantage students.

Projects run from Week 2 until the final week of Semester 2, typically around 20th-25th March.

Supervisors are free to decide on the structure and organisation of these meetings.

Group meetings *can* be used, but special care should be taken that individual supervision is available when needed, and that students are not disadvantaged (e.g. with very shy students).

Implementation work should be well under way by the end of Semester 1. If there are any difficulties in accessing the required equipment (or indeed any other difficulties) the Projects Coordinator should be informed. Implementation work taper off by the middle of Semester 2 in order that sufficient time can be given to writing up.

Final submission will be during the last week of Semester 2. This deadline is intended to give the students sufficient time to revise for the exams.

Please note: Supervisors are not authorised to grant any extension to Level 4 projects.

Project styles

There are two broad styles of projects:

Product-style projects, where the output should be a significant software artifact (strong design, programming and testing focus)

Research-style projects, where the output should focus on a research contribution (strong analysis, often empirical work)

All projects must have an evaluation portion. This is specifically assessed in the marking scheme. *Students should be made aware of this early in the project.*

2.2 Marking

Note carefully: the dissertation is primary evidence of student output. No matter how well a student does in the supervisor's eyes, **it is the dissertation that is marked.** It must be possible to justify the mark awarded from the artifact presented for marking (the dissertation) without reference to the student's other achievements.

All other evidence (e.g. source code) is supplementary and markers have the option, but no requirement, to take it into account when marking. Students should be made aware that marks will be awarded on the basis of the dissertation. The dissertation is the lens through which the product is viewed.

2.3 Ethics

If a project involves *any* form of study with human participants, ethical approval must be sought and obtained by the respective student *before* any experimental work begins.

There is an self-certification ethics checklist for simple cases (see the Moodle page); if this is not sufficient, the School Ethics Coordinator **must** be contacted to seek approval. Note that prior ethical approval for a research project does not apply to student projects. Separate approval is always required for student projects. It is the joint responsibility of the student and the supervisor to ensure ethical approval is obtained.

2.4 Status report

At end of autumn semester (Week 12), students submit a short progress report to their supervisor and to the project co-ordinator. This is **not** marked, but must be submitted to keep a check on student progress. Verbal feedback on the report should be provided at the start of Semester 2.

2.5 Dissertation

There is a page limit of 40 pages (30 pages for 20 credit Joint Honours projects).

There are some example dissertations on the Level 4 Projects Moodle page ("Hall of Fame").

Encourage students to start early on the dissertation, and submit drafts for review well before the deadline.

2.6 Video Presentation

Each student is required to submit a video presentation on their project work. Videos should be ~10 – 15 minutes length. The video presentation is worth 5% of the total grade for the project. Markers should ignore the categories related to questions and body language in the marking scheme.

3 Marking

Before marking begins, the supervisor will enter a **project context** that should state:

- Any work done before the project started
- Any problems that were encountered during the project outside of student control (e.g. hardware issues, lack of access to data) that might have adversely affected the project
- Any other relevant background for the marker.

This should typically be one paragraph at most. The context should **not** indicate the supervisors judgement of the quality of the project and must not include phrases like *“this is clearly an A grade project”*. The context is a dispassionate analysis of external factors that should be taken into account when marking. In many cases, this will be a very brief statement to the effect of “everything went smoothly”.

The *Supervisor* will mark the project according to the marking scheme (available on the Moodle page). Every project is second marked by a *Reader*, which will always be a UoG academic. The marker will see the project context before marking begins.

3.1 Marking process

- the supervisor and reader must first assess the student’s work independently
- both markers must mark the project in detail, by deciding a band for each criterion (from which the overall band is computed automatically);
- bands are awarded in the manner described within the University’s code of assessment https://www.gla.ac.uk/media/Media_124282_smxx.pdf.
- **In particular: when deciding on a mark for a project component, markers should first decide on a grade (for example, “B”). Having decided on the grade, the default mark is the centre of the grade (A3, B2, C2) etc. Markers should only deviate from this default if there is justification to do so;**
- both markers must justify their marks by addressing each criterion explicitly in their written comments. In arriving at bands, supervisors and readers should exercise their judgement regarding the difficulty level of the project. (There is no explicit criterion in the marking scheme for this issue.)

3.1.1 Third marking

When third markers are engaged, they should provide a single summary grade and a single written justification. Third markers do not need to provide a component-wise breakdown of the marks. Third marking is conducted independently without sight of the initial marking.

3.2 Reconciliation and Arbitration

The supervisor's and reader's independent assessments will be reconciled as follows:

- If their overall bands differ by 0 or 1, the supervisor's band is taken as agreed.
- If their overall bands differ by more than 1 but do not cross two grade boundaries, the supervisor and reader must confer to decide the agreed mark. If agreement is not possible, the Projects Coordinator will invoke arbitration.
- If their overall bands differ by 2 grade boundaries (e.g. a B and a D), the Projects Coordinator will engage a second reader (third marker), who will independently mark the project. The supervisor and both readers must then confer to make a final decision.
- If third marking has been undertaken, the default starting point for negotiations should be the median grade awarded.
- The projects coordinator has the discretion to invoke arbitration for any other reason, for example: where the two marks fall on either side of a critical borderline; where the agreed mark is a fail; or where the agreed mark seems unreasonably high or low.
- Arbitration can entail requesting that the supervisor and reader discuss their assessments and try to reach an agreement. Alternatively, arbitration can entail the Project Coordinator engaging a second reader (third marker), who marks the projects independently. The supervisor and both readers must then confer to make a final decision.

3.3 Feedback

Students may ask you for feedback on their project after marks have been released. Please do not provide *any* feedback until after the Exam Boards have been held and marks have been released to students.

- Students can request feedback directly from their supervisor.
- You do not have to provide feedback pro-actively.
- The feedback to be provided is comments from the the supervisor's marking sheet, and the partial mark for each of the entries:

Marking Notes

Marking Criteria

- Analysis

- Product (Research or Software)

- Evaluation

- Dissertation
- Professional Conduct

- You would not ordinarily be expected to release reader feedback unless specifically asked for, nor the provisional marks and discussion in case negotiation was needed. Please note that students could be granted this and access to all related discussion records (e.g. emails) in case of a formal appeal to Senate.
- You can find the complete mark sheets for all student projects on <http://sql16.dcs.gla.ac.uk/Reports>
 - Log in with DCS\your_dcs_user_name

Please note that students *cannot* dispute the mark awarded. Grades are final once the Exam Board has been held. This is University policy.

4 Reader allocation

Every project must be second marked by an UoG academic. Readers are allocated to projects using a bidding system. This lets staff rank projects to help match marking to staff interests. The bidding system will be opened before the L4 project deadline and will be available by the time marking is due to start.

4.1 Bidding process

The system is accessible at: <https://webapps.dcs.gla.ac.uk/LTC/bidproject.cfm>

Logged in as Williamson,JH. Hi, John. Information: the current academic session started Sep 17, 2018. This is week 26.

Public Page
 LTC Online
 YEAR HEAD DASHBOARD
 Projects

PROJECT BIDDING PAGE
INDICATE AT LEAST 23 PREFERENCES (REMAINING CAPACITY + 50%)
(READING + SUPERVISING TARGET: 24 - YOU ALREADY HAVE 9 ALLOCATED)
YOUR CHOICES : Last Updated: 14/3/2019 16:37

Title	Cohort	Rating	Re-Order	Remove
(Singapore) A machine learning based threat detection platform	Hons	1		
(Singapore) Interactive Data Visualization of Industry Auto-Generation Code	Hons	2		
(Singapore) Passive Localization of Drone	Hons	3		
(Singapore) VRSafety: Prototyping a VR experience for Safe Cycling	Hons	4		
(Singapore) GPS Anti-spoofing using Machine Learning	Hons	5		
(SINGAPORE) Data Analytics on Emergency Department Health Services	Hons	6		
(Singapore) A Virtual Reality Overlay for the Physically Challenged	Hons	7		
(Singapore) Laser Sentinel against Biting pest - Image processing Part	Hons	8		
(Singapore) Ship Detection from Satellite Image	Hons	9		

Add	Title (num bids)	Cohort	Supervisor
	Force-directed layout algorithms for Python (0)	Hons	Chalmers,M
	Game for supporting learning mathematics (0)	Hons	Jose,J
	Modelling UAVs in Controlled Air Space (0)	Hons	Johnson,C
	(Singapore) A Condominium Management Portal using Web and IoT technologies (0)	Hons	Keoh,S
	(Singapore) A Dashboard- Monitoring and Alert Triggering System for Venture (0)	Hons	Lin,D
	(Singapore) A Mobile App for Seafood Purchase (0)	Hons	Low,M
	(Singapore) A Warehouse Automatic Reporting System (0)	Hons	Lin,D
	(Singapore) A Web App for Hotel Revenue Management Forecasting Simulation (0)	Hons	Detlev,R
	(Singapore) An Manufacturing Inventory Tracking and Control System for Venture (0)	Hons	Lin,D
	(SINGAPORE) Attack Graph Generator from Heterogeneous Logs (0)	Hons	Loh,P
	(Singapore) Augmented Reality Navigation on Campus (0)	Hons	Kiat,S
	(Singapore) Automated land cover mapping (0)	Hons	Radice,G

WHO TO CONTACT

- Ethics: Brewster,S
- Exchange Coord: Camarasa,G
- HoS: Johnson,C
- L3 Projects: Omoronyia,I
- L4 Projects: Williamson,JH
- Level 1: Pezaros,D
- Level 2: Foster,ME
- Level 3: Norman,G
- Level 4: Omoronyia,I
- Library: Williamson,JH
- M Projects: Jose,J
- MobSE Prog: Murray-Smith,R
- MSc: Poet,R
- MSci: Williamson,J
- MScIT: Poet,R
- Quality Officer: Miller,A
- SE/ESE Coordinator: Storer,T
- TC: Vanderbauwhede,W
- Turnitin: Norman,G

Click projects on the left of the screen to add them to your bidding list. You will only see projects you are not supervising. You can re-rank the projects as required. Typically you should bid for (at least) twice as many projects as you expect to be marking, to have a solid chance of being allocated projects from your bidding list.

4.2 Notes

If you do not rank projects, or a match cannot be made from the list that you have chosen, then you will be assigned random projects to read up to the marking capacity required. UoG staff should ensure they select projects to read from UGS and UoG sites, as these are allocated separately. If you choose only UoG projects, you will receive random Singapore projects.

5 How to mark projects

Projects are uploaded to a file server in the School by students. Projects will be available for markers to read shortly after the submission deadline. The source code for the project will also be uploaded to the same server.

*Note: if the project is sensitive in nature (security concerns, NDA, etc.) the students affected must be instructed **not to upload** the project to the server. Instead, the student should provide the project on an **encrypted** USB disk or similar to both of the markers. The project supervisor will need to coordinate the distribution of the submission to the reader and verify that the submission deadline has been met in these special cases.*

5.1 Accessing projects

Projects are stored on a file server in the School. They can be accessed at:

`/local/lev4proj/<year>` [Linux systems]

or

`\\unix6.dcs.gla.ac.uk\lev4proj\<year>` [Windows systems]

Where `<year>` is the later year of the current session. For example session 2017/2018 will be `<year>=2018`

These will only be available on the School network (e.g. physically in the School, VPN'd or SSH'd in).

[illegible]

The marking deadline for Level 4 projects is typically around the end of April, about 4 weeks after dissertation submission. All marks are entered electronically. The marks will be entered in the online system at <https://gaua.dcs.gla.ac.uk/studentprojects/index.cfm>. Supervisors must enter a **context** for the project, as described above, before readers can mark projects.

Each of the mark categories in the marking scheme will be entered in a box in the online system, which looks like this:

View/Edit Context Edit Project Details			
Assessment Criteria (PDF)			
Student Name:		Matric No.:	
Supervisor Award:	H	Cohort:	CS4H
Project Title:			
Project Type:	Supervisor Defined	Credits:	40
Supervisor:		Reader:	
Marking			
Category	Comments (Please justify the bands awarded - provide comments indicating both merits and shortcomings.)	Band	Weight (1.00)
Analysis		H	0.15
Product (Research or Software)		H	0.40
Evaluation		H	0.10
Dissertation		H	0.20
Professional Conduct		H	0.10
Presentation Skills		H	0.05
Summary (Please indicate if you believe that the project should be considered for a prize.)			
Save Changes			

For each category, markers enter a justification (a few sentences), and select a band, and then write an overall review of the whole project. The overall grade is computed automatically.

The presentation mark should be recorded and entered in the online form. For presentations, there are several sub-categories on the marking scheme and it is up to supervisors how to reach an overall presentation grade from these components. The form will take the overall presentation grade.

The one page summary submitted in December is not explicitly marked, though a poor summary/plan might reflect badly on professional conduct.

Marks may be saved while being entered. Once saved, an option to commit the final marks will become available. Only press this when the grade is finalised; it will not be possible to edit the form after this point.

When both reader and supervisor marks have been committed, both sets of marks and justifications will be visible to each marker to aid in negotiating a grade.

6 Student Good Practice

6.1 Records

It is strongly recommended that students are asked to keep:

- a) Minutes of all meetings with supervisors and partners (if applicable).
- b) A log book indicating all time spent on the project which is reviewed regularly by the supervisor
- c) Source code and dissertations under version control, with sensible use of commit logs. For some projects, it is useful to have supervisor access to these repositories, and have the supervisor pull the repository and compile the project!
- d) References under control with a reference manager (Zotero, Mendeley, EndNote, etc.).

This is not assessed and not formally required, but is good practice. Students will be separately instructed to follow these practices. Note that these records are the responsibility of students.

6.2 Ownership, originality and independence

Independence and originality

Students must be able to show they have completed substantial *independent work*.

The work does **not** have to be original (it could result in developing an application that has similar functionality that already exists, for example), but it must be developed independently. It is perfectly acceptable for projects to build on existing work, but the separation of the project work from what went before must be made clear.

IP

In UoG, IP belongs to *students*, who have full rights to everything they develop (including commercialisation, etc.). Students can be asked to transfer IP, but are not under obligations to do so. IP agreements with commercial partners should be made in advance of the project starting.

Some projects require NDAs with partners. This is sometimes unavoidable, but it is discouraged as it can adversely interfere with the educational aspects of the project (e.g. presentations) and can hamper student involvement in awards or creating a portfolio of work for future employers.

Commercial partners

A project with a commercial partner cannot simply work as if the student was an employee of that partner. The student must be demonstrating independent development of the work, and not following specific weekly directions from the partner. The overall topic and direction can of course be given by the partner.

Particular care should be taken if the student is actually a current employee of the commercial partner; it must be crystal clear that the project is directed by the academic supervisor for the

academic benefit of the student, and **not** for the (sole) benefit of the employer (e.g. it doesn't matter that feature X is needed for a release next week -- if it's not directly relevant to producing a good project dissertation). Students will need to seek special permission from the project coordinator to undertake projects with their current employers.

Plagiarism

Students should be reminded to check the plagiarism policy very carefully, and rigorously adhere to the rules. Uncited text and figures have cost students dearly in Academic Conduct proceedings in recent years. *The University has no tolerance for plagiarism and the rules are enforced vigorously.*

7 Some example project titles + descriptions

[since last year's proposals are not available to view on LTC easily, I've collected a few representative samples from last year here].

7.1 Predict the Answer: Predictive Learning from Skyscanner Queries (Main)

This project is about predictive analytics in Big Data systems. Its novelty is that analytics tasks are answered through learning from queries issued to the Skyscanner[*]. The concept is to exploit past queries to create a predictor, that predicts the output (answer) of a new, unseen query. For instance, can we predict the cheapest air-ticket fare flying from Glasgow to London, without accessing the airlines' data?

The proposed system will be developed in Java over Spark. The goal is to develop a system that provides the data analyst with a predictor component using only information from historical queries and their answers.

Note: Because of the involvement of a commercial company, a student will need to sign an NDA with Skyscanner if they wish to take this project.

7.2 Smartwatch based multimodal reminding (Main)

Reminders can be an effective way to prompt people to perform a future intention. They are a particularly useful tool for people with memory impairments. However it is not always easy for reminders to be set in the first place, especially if users have limited experience with technology or have memory difficulties that mean they forget to set the reminder in the first place. Additionally different types of reminders may be more appropriate at different times. For example people may want a more subtle reminder for medication and a louder reminder for a social event. Smartwatches can be used to enter reminders quickly and to receive reminders subtly.

This project will investigate multimodal input and output for a reminder app linked to a smartwatch. The aim of this project will be to understand how different input gestures and output modalities or combinations of modalities of reminders are evaluated when people create and receive smartwatch notifications. For this project the student will develop a study investigating multi-modal reminders from different types of notification from a smartwatch. The student could choose to evaluate the system with participants from the general population and / or older users. The student will have access to state-of-the-art smartwatch hardware (LG watch urbane 2nd edition).

7.3 Live programming with error-driven code completion (Main)

(The second of two projects extending our experimental live programming language/IDE with new features.)

Live programming systems abandon the traditional edit-compile-debug cycle, providing a tighter feedback loop that makes the consequences of changes to the program immediately apparent to the programmer. Our experimental language (implemented in a TypeScript, a typed JavaScript) explores the use of liveness for testing and verification: as the user writes code, executions of the program which get "stuck" because of a runtime error are presented to the user as source-code errors. We collect the errors by running the program "behind the scenes", recording any states that go wrong. (See the video here for a demo: <https://vimeo.com/163716766>.)

The goal of this project is to use these errors to provide a potentially powerful form of "code completion". For example, if objects A and B can get stuck in particular state because B is missing a method that A wants to call, we could report this as a source error which the user can click to have the required method skeleton automatically added to that state of B. (In our language objects have explicit states, with different methods available in each state.) Each fix can expose new errors, allowing the programmer to drive the implementation forward by chasing away the runtime errors.

Special requirements: some familiarity with JavaScript or similar language and HTML/CSS required.

7.4 Electoral violent events clustering based on social media (Main)

Social media platforms such as Twitter are popular as a medium for reporting and discussing current news and events from around the world, including political events.

Both politicians and citizens are embracing social media to disseminate information and comment on various topics, particularly during significant political events and campaigns. The aim of this project is to build a web-based interface, which will display clusters of events involving violence as a timeline. Only significant tweets about the event should be displayed and supporting evidence in the form of links to newswire articles should be provided.

In this project, you will mine a large collection of election data from Twitter, apply clustering algorithms to group the dataset into different events and visualise the events in a suitable web-based timeline interface.

Technology: Java, Javascript, Python, CSS, HTML

7.5 Range type attributes and number representations for FPGAs (Main)

By adding a range/precision attribute in Fortran or a e.g. using the `__attribute__` format in GNU C and analysing the dependencies, it is possible to determine the range and precision for every variable using a few simple heuristics. The final aim is to use this for FPGA code generation. For this project, we propose to extend the Fortran or C datatypes so that they allow to express the number of bits required. The dependency analysis is challenging because it requires need to work out the dynamic range of a variable based on all expressions that are used to compute its value.

7.6 Visualisation for Cyber-Situation Awareness in SCADA systems (Main)

In critical infrastructures, it is important to present information about suspicious behaviour on a network in a way that the human operators are easily aware of an incident but not overwhelmed by too much information. A good situation awareness should lead to good decision making and ensure that the system assets are properly utilized to mitigate the occurrence of future incidents. This research will look at implementing situation awareness techniques in SCADA systems, specifically an experiment on visualising cyber-attacks on the network.

It will first review existing research on situation awareness and SCADA systems, investigate possible cyber-attack scenarios suitable in SCADA networks and then classify these attacks. Next, it would examine how the three levels of situation awareness theory can be applied to a SCADA network and then specify and justify which situation awareness measurement techniques are suitable for experimenting, finally the results will be discussed and conclusions extracted.