

Software Engineering Group Project

(COMP2043.GRP)

Student Handbook 2024–2025

Dave Towey

February 24, 2025

Contents

1	Introduction	3
2	Ethics	3
3	Important Dates	4
4	Teams	5
5	Projects	6
5.1	Project Allocation Key Points	6
5.2	Project Briefs	7
5.3	Project Preferences	7
5.4	Project Bidding	7
5.5	Project Awarding	9
6	Legal, Societal, Ethical and Professional Issues (LSEPI)	9
6.1	“Big picture” Reflection Checklist	10
6.1.1	Intellectual Property	10
6.1.2	Research Ethics	10
6.1.3	Data protection	11
6.1.4	Other Legislation	11
6.1.5	Broader Ethical and Social Considerations	12
7	Supervisors	12
8	Class Sessions	13
9	Meetings	13
10	Equipment	14
11	Assessment	15
11.1	Assessment Guidelines	16

11.1.1 Exceptional (90–100 %)	16
11.1.2 Outstanding (80–89 %)	17
11.1.3 Excellent (70–79 %)	17
11.1.4 Good (60–69 %)	18
11.1.5 Average (50–59 %)	18
11.1.6 Adequate (40–49 %)	18
11.1.7 Poor (below 40 %)	19
11.2 Peer Assessment	19
11.2.1 No Contributions?	20
11.3 Interim Group Report	21
11.4 Final Group Report	22
11.5 Software & Documentation	23
11.6 Individual Report	25
11.7 Marking of Other Components	26
11.7.1 Team Project Website	26
11.7.2 Demonstration	27
11.7.3 Presentation	27
11.7.4 Questions & Answers (Q&A)	27
11.7.5 Promotional Digital Aretfact	27
11.8 Submission of Deliverables	27
12 Open Day	28
12.1 Open Day 2024–2025	28
12.2 Presentations	29
13 Live Q&A	31
14 Golden Rules	31

1 Introduction

During your first/second year of study you learned some of the basic principles of computer programming. In industry, however, you won't usually be programming on your own, but will be working as part of a group of people developing a larger application. As an introduction to group working, part of your Part I Year is made up of a software engineering group project, in which you are divided up into small teams (groups) to design, program, and document a computer application.

The aim of the group project module is for you to gain some software engineering experience, and experience in the various different aspects of group working. This will include engineering requirements, system architecture and design, user interface design, implementing a medium sized, multi-component system, systematic testing and debugging, programming in a team, and use of software engineering tools. The project will also include running meetings, making collective decisions, time and people management, writing reports, giving presentations, interpersonal skills, and resolving conflicts. During the project, you will also learn that being a member of a programming team can be great fun!

The group project module runs for the entire academic year and is worth 20 credits.

2 Ethics

An essential step before undertaking any research, at university and beyond, is to address any ethical concerns. This may be the first time for most of you to consider this, but it will certainly not be the last!

There will be a class session (Section 8) about ethics, but you should also begin investigating the UNNC policies as soon as you can. A good starting point is here: <https://www.nottingham.edu.cn/en/research-and-business/ethics.aspx>

Obviously, each project will have different ethical dimensions: Teams will need to discuss these amongst themselves, and with their supervisor. No work on your project can begin until the relevant ethics forms have been correctly completed. This process carries marks (Section 11) and has a deadline (Section 3). Please note: Marks for correct submission on time will be reduced if it is later discovered that the team did not correctly address the ethical issues over the course of the project.

From 2024 onwards, GRP teams need to consider the broader “LSEPI” context of your projects: Please refer to Section 6 for details.

3 Important Dates

The group project is assessed through a number of tasks to be completed at specific times during the project. There are also other important deadlines that must be met. The key dates and deadlines for the 2024–2025 academic year are given in Table 1.

Table 1: Key Dates

Task	Date or Deadline
Project bids submission	Wednesday, 9 October 2024 (3 pm)
Equipment requests can begin	From Wednesday, 16 October 2024
Completed Ethics forms due	Thursday, 31 October 2024 (3 pm)
Group project site up and running	Thursday, 31 October 2024 (3 pm)
Interim reports due	Thursday, 5 December 2024 (3 pm)
Team final reports and software due	Wednesday, 2 April 2025 (3 pm)
Recording of Software Demonstration due	Wednesday, 9 April 2025 (3 pm)
Recording of Team Presentation due	Wednesday, 9 April 2025 (3 pm)
Team promotional digital artefact due	Wednesday, 9 April 2025 (3 pm)
Open Day	Wednesday, 16 April, 2025 (9 am – 6 pm)
1-minute introduction & live Q&A	Wednesday, 16 April, 2025 (9 am – 6 pm)
Individual final reports due	Tuesday, 22 April 2025 (3 pm)

4 Teams

The terms “group” and “team” may be used fairly interchangeably throughout this document and the entire module.

At the start of the group project module you will be divided into teams of around four to seven students. The team division is done so as to ensure diversity and, to the extent possible, a well-balanced set of skills within the groups. Note that forming your own teams is not permitted, and that you are not allowed to swap places with someone from another team. Working with people from different backgrounds, many of whom you may not know initially, is a central aspect of this module, just as it is going to be in your future professional lives.

The team allocations can be found in a separate document on Moodle. Note that it can take a couple of weeks from the start of the module before the teams are completely stable. **Important: If you are not on the list of students taking this module but should be, or if you have not been assigned to a team but know you should be in one, you need to get in touch with the module convenor as soon as possible.** In particular, if you are an incoming exchange student, you are unlikely to be in a team from the start. Similarly, if you find that you cannot make contact with certain team members, perhaps because they have elected to do a different module, it is important that you let the convenor know.

Each team should appoint one member as the Team Leader. In addition to their normal work as a member of the team, the Team Leader will play a management role. For example, the Team Leader should ensure that everyone is involved, that work is divided fairly and according to individual strengths, that tasks are completed on time, and that nothing is forgotten in the project. This does not mean that the Team Leader decides everything. On the contrary, the project is “owned” by the team as a whole, and the team may also find it useful to (formally or informally) appoint other leaders for specific aspects of the project, like Chief System Architect, Repository Maintainer, Lead Programmer,

or Lead GUI Designer. However, the Team Leader has a specific responsibility to ensure that the team works effectively as a whole, that everyone pulls in the same direction, that the project progresses according to plan, and, ultimately, that the project goals are achieved and the prescribed deadlines met.

Choosing a leader can be difficult, particularly if the team members do not initially know each other. However, it is very important for the success of the project that a leader is chosen. Not doing this is one of the most common regrets found in the final reports from group projects of previous years. In the past, some teams have changed leader during the project. If you have problems deciding on a leader, your supervisor can help in making the choice.

5 Projects

A number of different projects will be available for the teams to bid for. Each team will be allocated one project that they will engage with over the year.

5.1 Project Allocation Key Points

Similar to what happens in real life, there are a number of important things to be aware of regarding the projects:

- Each team must work on a different project to every other team — no two teams may work on the same (or even similar) projects.
- The projects may not be very clearly described in the project briefs initially made available to the teams.
- Initially, there are no supervisors for each project. This means that there is nobody to approach for additional information (at this stage).
- Projects that are available initially, may change, or even disappear.
- Each team should go through all the projects and identify up to three that they are most interested in doing. Each team may then submit up to three “bids,” one for each of the identified preferred projects.

- There is no guarantee that a team will get its preferred project. There is no guarantee that a team will get a project that they even want at all.
- Each team may compete with other teams to be awarded their preferred project(s), initially through the bids.
- There may (initially) be more teams than projects. In this case, those teams not assigned one of the initially offered projects will be assigned a new project at the end of the bidding process.

5.2 Project Briefs

A catalogue of “project briefs” will be made available through Moodle at the start of the year. Each brief will contain a basic overview of a potential project. At this stage, there is no further information about any of the projects, and no person to approach to ask for more information.

5.3 Project Preferences

Soon after you have been put into your teams, you should discuss which projects you would most like to do. You should decide on the top three choices. (Note that it is possible that you may end up with a different project to these choices.)

When you have identified your top three projects, your team should prepare your “bids” for each of your choices.

5.4 Project Bidding

The “bidding” phase involves each team preparing a package (called a bid) that they use to try to convince the project proposer to award the project to them. Each team may submit up to three bids (ranked in order of preference). The submission link and detailed explanation will be in Moodle. All bids must be submitted by the deadline: no late submissions will be accepted.

Each bid file should be a single archive file, in .zip format, containing all relevant materials. The bid file should be named as follows:

<TeamID>.<PreferenceNumber>.<ProjectID>.zip

where TeamID and ProjectID are the team and project identifiers (e.g. Team.2024.07 and P11), and PreferenceNumber is 1, 2, or 3 (indicating 1st, 2nd, or 3rd preference — where 1 is the most preferred). For example, if Team.2024.07 wanted to do project P11 as their first preference, their submission file would be named: Team.2024.07.1.P11.zip

The individual bid should contain all the files and materials that the team will use to attempt to convince the project proposer to award the project to them. There are many possible things to include in the bid, for example:

- A letter outlining why the team is a good candidate for completing the project
- Details of relevant experience or expertise. Some teams may like to include CVs or resumés for their team members
- Some proposed ideas related to the project
- An estimated timeline or schedule for how the team would complete the project on time

Not all bids would need to include all of the above items, and obviously there are many other things that could also be included. Your team should spend time considering how best to convince the project proposer to award you the project.

Note that the file upload size limits may mean that you should take steps to keep your bid file as small as possible. You should also take steps to make sure that you do not encounter last-minute network issues: It is the responsibility of each team to ensure that their bid is submitted. It has been noticed in the past that network traffic immediately before the deadline can cause last-minute submissions to not be accepted: Aim to submit with sufficient time before the deadline.

5.5 Project Awarding

Once the deadline for bidding has past, all the bids will be examined, and some projects will be awarded.

If any project has only one bid submitted, and that bid was as a first preference, then the bidding team will be awarded that project.

If any project has more than one bid as first preference, then the relevant bids will be examined and one chosen.

Once all the first preference bids have been processed, the second preferences will be examined in a similar manner, followed by the third preferences.

The projects not yet awarded at the end of the preferences processing stage will be then examined and matched to the remaining teams. Details of if and how this will be done will be explained in the class sessions (Section 8).

At the end of this phase, each team will have a project.

6 Legal, Societal, Ethical and Professional Issues (LSEPI)

Your project doesn't happen in a vacuum. Your work is affected by the world around you. This influence includes laws and University policies, but also the concerns and priorities of society and your peers. On the other hand, your work can affect the world around you. In fact, it might also change the world (for better or worse)!

As part of your "Contributions and reflections" in your final reports, you should explain how the team has considered the "bigger picture" within which your work is situated. There is a checklist below (Section 6.1) that identifies several possible issues. Depending on the nature of your project, some of the checklist items may be more relevant than others — but everyone should have something to reflect on (even if it is only the last question).

In the final report you should present your reflections on each issue, including

why they are (or are not) significant for your project.

6.1 “Big picture” Reflection Checklist

(You should search for, and read about, any relevant acts or bills listed below.)

6.1.1 Intellectual Property

You should consider the legal issues (in China and elsewhere) related to intellectual property (IP).

Has the work created any IP (e.g. source code, documents, photos or recordings, patentable technologies, trademarks, etc.)? If so:

- What IP has been created, and who owns it?
- Do you have a plan to commercially exploit the work? How?
- Do you plan to distribute or promote the work in other ways, for example using open source/open data/creative commons licenses? How?

Please also see the “Provision and Processing of Intellectual Property Rights for Students and Graduates at The University of Nottingham” (<https://www.nottingham.ac.uk/fabs/documents/studentipr.pdf>).

6.1.2 Research Ethics

Does the work involve human participants or data subjects? If so:

- Have you followed the University of Nottingham research ethics requirements?
- How does the work safeguard participants and data subjects?
- How does the work reflect principles of informed consent and voluntary participation?
- Were there any ethically tricky aspects to the work, and how did you deal with them?

- How did you ensure the ethical status of your project throughout its duration?

Please also see the Ethics section (Section 2) of this handbook.

6.1.3 Data protection

- Is the work collecting or using personal data?
- If so, how is it complying with the requirements of data protection legislation?

You may also like to look at what UNUK's CS have said about what constitutes Personal Data: https://moodle.nottingham.ac.uk/pluginfile.php/10771251/mod_label/intro/Personal_Data_Guidance.pdf.

Note that this should also have been covered by your research ethics approval process (Section 2).

6.1.4 Other Legislation

Is the work affected by other legislation? If so, how is the legislation relevant and how has the work responded to that legislation? Here are some examples from the UK (you may like to explore anything similar in China):

- The UK Equality Act (2010) has implications for accessibility.
- The UK Computer Misuse Act (1990) and Counter-Terrorism and Security Act (2015) have implication for (potentially) criminal use.
- The UK Online Safety Bill (if/when it comes into force) will have particular implications for systems accessed by children and other vulnerable groups.
- The Privacy and Electronic Communications (EC Directive) Regulations (2003) have implications (together with the Data Protection Act) for communication systems.

6.1.5 Broader Ethical and Social Considerations

How have you considered the broader ethical and social aspects of the work?

For example:

- Who will benefit from the work? How?
- Could the work have unintended consequences? What?
- Is there any way that your work could be used to cause harm to others?
If so, in what way, and what measures could be taken to prevent it?
- Who might be affected by the work, directly or indirectly? How?
- Have people who might be affected been able to influence the work?
- Is the work accessible or beneficial to all?
- Is the work likely to reduce or increase societal inequalities?
- What impact might the work have on the natural environment?
- Is the work worth doing? Why?

The Final Year module on Professional Ethics may look at “Responsible Research and Innovation” (RRI). Some resources related to this are:

- The RRI prompts and practice cards (<https://experience.horizon.ac.uk/2022/09/30/rri-prompt-practice-cards/>).
- The Moral-IT cards
(<https://lachlansresearch.com/the-moral-it-legal-it-decks/>).

These resources may be very helpful for you and your team to reflect on such aspects of your projects.

You may also like to explore the code of conduct for the British Computer Society (BCS): <https://www.bcs.org/media/2211/bcs-code-of-conduct.pdf>

7 Supervisors

Soon after the projects are assigned, each team will then be assigned a member of staff as their supervisor. The supervisor is not in charge of the project, and he or she is not there to make decisions for you. Rather, the role of the supervisor is to monitor the team, to ensure that it remains focused, to ensure that steady progress is being made, to promote discussion, and to offer advice on matters such as report writing and presentation. In some cases, the supervisor may offer technical advice. The supervisor also plays a significant role in marking the project. For some projects, the supervisor may also play the part of a client for whom the application is being developed. Teams meet with their supervisor once a week (see Section 9).

8 Class Sessions

The module is supported by a series of weekly class sessions given throughout the year. The exact number and content may vary from year to year. Various practical aspects of the module are covered, including specific support sessions on presentation and writing. There may also be guest lectures (perhaps from industry representatives) covering a wide range of topics relevant to software engineering, including professional and quality issues.

See the Moodle page for details.

9 Meetings

Each team should arrange regular times for two kinds of weekly meetings: one formal meeting with their supervisor present, and one (or more) without.

The weekly meetings with the supervisor may be held in his or her office, and should last about 30 minutes. The purpose of these meetings is for the team to report on progress since the last meeting, discuss any problems that have arisen, and plan the work that will be carried out by each member before the

next meeting. These meetings may be quite formal in nature, with a chairperson (who runs the meeting) and a secretary (who records the progress of the meeting and produces “minutes”). These tasks should rotate among the team members, so that everyone gains experience with both tasks. Most students will not have attended formal meetings before, so towards the start of the group project module you may be given a special class session on this topic. This session will give some suggestions for how to structure your meetings, and will explain the roles of the chairperson and the secretary.

The weekly meetings without the supervisor will normally be held in a mutually convenient location in the school, or elsewhere on campus, and have no prescribed time limit. What goes on at these meetings is completely up to the team, but they are usually more informal in nature. There is no requirement for a chairperson or minutes, but some groups may find it useful to carry on with these practices here.

10 Equipment

In the first few weeks of the group project module, each team needs to think about whether they will need any non-standard hardware or software (a list of standard software will be provided by the IS technical support group). If so, you need to make the equipment request, detailing the hardware and software requirements for your project, via the Computer Science technical support staff, Jane Zhao (jane.zhao@nottingham.edu.cn). Under no circumstances may students themselves install hardware or software on school machines. Note that you only can make one request related to your group project.

What kind of requests can you make? No guarantees are made that all requests can be accommodated, but Technical Support will try to help with any reasonable request. The most common would be installation of (various kinds of) “free” software that normally is not available on School machines, or standard hardware (like a PC) configured in some specific way for your project.

Table 2: Team Overall Mark Composition

Group Task	Weight
Team Project Website	2 %
Completion of Ethical Approval	3 %
Interim Group Report	15 %
Final Group Report	30 %
Software & Documentation	15 %
Demonstration	10 %
Presentation	10 %
Q&A	10 %
Promotional Digital Artefact	5 %

Beyond that, it may be possible to provide access to hardware and software that the School happens to have anyway. Your supervisor may also have access to special hardware, or he or she might be willing to purchase something to support the project.

For further details, please discuss with your supervisor.

11 Assessment

The group project is assessed through a number of tasks to be completed at specific times during the project. There are no written examinations. Most tasks are carried out as a team, but there is one individual task: an Individual Report (Section 11.6) to be handed in at the end of the project. Consequently there are two parts to the final mark that each student will get: one part for the team tasks and each individual's contribution to those, one part for the Individual Report. This is explained in more detail in the following.

For the group tasks, each Team is awarded a Collective Team Mark on the standard university scale, with the different aspects of the project contributing according to Table 2.

Peer assessment is then used to derive an Individual Mark for Group Work, on the university scale, for each team member from the Collective Group Mark. This is done by distributing the collective mark among the team members according to merit as perceived by the group members themselves (but vetted by

Table 3: Individual Overall Mark Composition

Component	Weight
Individual Mark for Group Work	80 %
Individual Report	20 %

the supervisor), such that the average of the individual marks for the team work equals the assigned Collective Team Mark.

The Individual Report is also assessed on the standard university scale. The Individual Mark for Group Work and the mark for the Individual Report are then weighted together into an Overall Individual Mark according to Table 3.

Important! Late submissions will be penalised. In most cases, this will be at the standard university guidelines of five percentage points/marks per working day (e.g., an original mark of 57% would go to 52%, 47%, ... etc.). If a different penalty is to be applied, this will be communicated through Moodle.

Most tasks are marked by the supervisor and at least one second marker. The presentations and Q&A are usually marked by a third member of staff in addition to the supervisor and second marker.

11.1 Assessment Guidelines

The following guidelines are applied when judging the quality of the Group Reports and Software. It should be noted that the descriptors used here (“exceptional”, etc.) may differ from those in any relevant rubrics (which will be made available in Moodle in due course):

11.1.1 Exceptional (90–100 %)

The reports and software should exhibit all the characteristics of an Excellent grade. Additionally, the project should have been carried out in an utmost systematic and professional manner, as evidenced by a problem analysis and subsequent requirements specification of stunning clarity and insight, a system design of highest possible quality that manifestly meets all requirements and given at a level of precision and detail that directly could be translated into an

implementation, an implementation of highest possible quality and completeness whose conformance to the original specification has been verified rigorously, and impeccable project management in terms of planning, workload management, meeting deadlines, making the best possible use of each team-member's skills and strengths, and with absolutely minimal (technical or otherwise) guidance from the supervisor.

11.1.2 Outstanding (80–89 %)

The reports and software should exhibit all the characteristics of an Excellent grade. Additionally, the project should have been carried out in a very systematic and professional manner, as evidenced by a problem analysis and subsequent requirements specification of significant clarity and insight, a system design of very high quality that manifestly meets all requirements and given at a level of precision and detail that more or less directly could be translated into an implementation, an implementation of highest possible quality and completeness whose conformance to the original specification has been verified thoroughly, and very skilled project management in terms of planning, workload management, meeting deadlines, making vary good use of each team-member's skills and strengths, and with very little (technical or otherwise) guidance from the supervisor.

11.1.3 Excellent (70–79 %)

The reports and software should display a complete and thorough understanding of the conceptual and practical issues surrounding the project topic, including related work. The project should have been carried out in a systematic and professional manner, as evidenced by a clear and insightful problem analysis and subsequent requirements specification, a system design of high quality that meets all requirements and substantially provides enough detail for implementation. Software should be completed in all respects and exhibit very high quality. There should be evidence of a high degree of testing. Supporting documentation

should be complete and approaching the standard of high quality professional documentation.

11.1.4 Good (60–69 %)

The reports and software should show a good understanding of the conceptual and practical issues surrounding the project topic, including an adequate grasp of related work. The quality of the analysis, requirements specification, and design should be good, and the writing of the reports should be good in general. Software should be competently written. Evidence of testing should be presented. The software should be a complete and usable package which not only illustrates the principles of the work but also exhibits good levels of quality. Supporting documentation should be excellent for all purposes; it should be complete, well written, well presented and generally exhibit high quality.

11.1.5 Average (50–59 %)

The reports and software would be expected to display an adequate understanding of the key conceptual and practical issues, although weakness may be present in some areas. Some account taken of related work. The quality of the analysis, requirements specification, and design would exhibit significant weaknesses; the design would not as it stands constitute an adequate basis for implementation. The writing would exhibit some flaws. Software should be adequate to illustrate principles; it may display weakness in areas not central to the work and lack comprehensive testing. Supporting documentation would be well presented yet lack completeness; the quality of the documentation should be very good.

11.1.6 Adequate (40–49 %)

The reports and software would display an incomplete understanding of the central issues relating to the project topic. The reports would lack a clear structure and strong argument, and the quality of analysis, requirements specification, and design would be below average. The writing would be mediocre.

Software would be incomplete, poorly commented and difficult to understand; it would exhibit poor levels of quality. Supporting documentation would be adequate.

11.1.7 Poor (below 40 %)

The reports would display a very poor understanding of the project area; there would be no clear structure and the analysis may be weak or incomplete. The reports would be poorly written and presented. Software would be limited in capability, and difficult to use. Supporting documentation would be inadequate for most purposes.

11.2 Peer Assessment

As explained above, peer assessment is a key aspect of the overall assessment. It is used to distribute the collective group mark to each individual according to merit as perceived by the peers, i.e. the other team members. Each team member is asked to rate each of his or her peers according to a number of aspects on a purpose-designed form. A written justification supporting the ratings is also required for each member. The peer assessments are submitted in complete confidence as part of the Individual Report, and will not be seen by the other students. See below.

The COMP2043.GRP Peer Assessment Form is available on Moodle.

Be sure to read the instructions on the forms carefully, as well as the instructions below, before filling out the forms. The completed forms are to be included as an appendix in your Individual Report. You may also be required to submit the details electronically, through Moodle and/or Microsoft Forms, for example. Please follow the instructions in Moodle.

Each group member is assessed according to the following aspects: Research and information gathering; Creative input; Co-operation within team; Communication within team; Quality and quantity of concrete contributions to team deliverables; and Attendance at meetings

Thus, it is not just the amount of contributed work that is being evaluated, say in terms of words in the reports or lines of source code, but a wide spectrum of contributions all of which are important for a successful end result. This also means that everyone gets a chance to contribute according to their specific skills and get proper recognition for this. Clearly, someone who excels in designing the system architecture and implementing it has made a very important contribution. Likewise, someone who is a great writer and took the lead on getting the team reports together has also made a big contribution. But someone who was instrumental in mediating between conflicting views, say, and thus helped holding the team together has clearly also made a very valuable contribution to the team as a whole.

Each aspect is rated on a five-point scale ranging from “None” via “Adequate” to “Excellent,” where “Adequate” means that the assessed person has done what is expected for that aspect, no less, no more. Note that it is the relative performance of group members that matters as the peer marking only serves to redistribute the assigned collective group mark. For example, if everyone rates everyone else as “Excellent,” then the end result is that everyone gets the same mark for their contribution to the group work which is going to be equal to the collective team mark.

Assessing your peers is a privilege and big responsibility. Be fair and objective in your evaluations. In the past, the members of the School faculty have been very impressed with the quality and honesty of the peer assessment, and there have almost never been any problems. However, each supervisor is charged with carrying out a sanity check on the peer marks based on what they have learnt during the interaction with the team members throughout the year, as well as from the team and individual reports. Should there be obvious problems with some of the peer marks, then the supervisor together with the module convenor will override any or all of those marks where necessary.

11.2.1 No Contributions?

As we'll discuss in the Class Sessions, there will be tension, and even conflict, amongst team members. One of the reasons why you are involved in this module is to experience team work, including the negative aspects. Obviously, if there are serious conflicts within the team, you should consult your supervisor or the convenor for advice.

If a team mate really does not contribute to the project, then it is likely that the other team members will grade that student's contribution as "None." In such situations, if the supervisor agrees, this may lead to that student receiving a failing grade for the module.

11.3 Interim Group Report

The Interim Group Report should be 4000–5000 words (around 12–15 pages; excluding any appendices). It is due around the half-way stage of the project. Each team must submit one such report, written as a group. The main content of the report should be:

- Updated and expanded description of the problem to be solved.
- Background information and research such as
 - survey of any existing systems that address similar problems
 - results of any market research conducted
 - results of technical research into suitable platforms, tools, technologies, algorithms, data structures, etc.
- Requirements specification for the system to be built (agreed between the team and supervisor).
- Initial design of the proposed system and its user interface.
- Record of key implementation decisions, such as programming languages, operating systems, computers, and any additional software and hardware to be used, along with reasons for those decisions.

- Results of any initial implementation steps/prototyping.
- Discussion of any problems encountered so far, including both technical issues and management issues, like group working etc.
- Time plan for the project.
- Minutes from the meetings held to date, included as an appendix.

The front page of the report should include:

- The text “COMP2043.GRP Interim Group Report”
- Project title
- Date
- Team name/identifier
- Names and School of CS usernames of all group members
- Name of supervisor

11.4 Final Group Report

The Final Group Report should be 7000–8000 words (around 20–25 pages; excluding any appendices). It is due at the end of the project. Each team must submit one such report, written as a group. The report should be a self-contained, updated, and expanded version of the Interim Group Report less parts that are no longer relevant.

Important! The report must be self-contained. In particular, the report must not assume that the reader has read the Interim Group Report prior to reading the final report.

The Final Report should include:

- Updated design of the system and its user interface.

- Discussion on the implementation and testing of the system. This must include a list of all major system components, which of these were written by the group and where the others come from, and an overview of the developed source code hierarchy.
- Summary of what was achieved, referring to the stated requirements.
- Reflective comments on the success of the project, both from a technical and a project management perspective, including group working issues etc.
- An appendix giving a description of how the developed system was tested (test cases, example outcomes, etc.)
- Minutes: Minutes from all formal meetings should be submitted as an appendix.
- A user-manual (if appropriate) should be included as an appendix. Excerpts of the developed code can be included in the report for illustrative purposes, but any lengthy excerpts should go into the appendices.

The front page of the report should include:

- The text “COMP2043.GRP Final Group Report”
- Project title
- Date
- Team name/identifier
- Names and School of CS usernames of all group members
- Name of supervisor

11.5 Software & Documentation

An electronic copy of all developed source code should be submitted at the same time as the group report at the end of the project. The code should be

submitted either in the form of a zip archive or a gzipped tar archive of the entire source code hierarchy. Auxiliary components, such as code libraries developed by others, databases, or other lengthy pieces of data may not need to be included (check with your supervisor). Consequently, it may not be necessary for what is submitted to be complete enough to run “out of the box.” But, if feasible, it is good if what is submitted is complete and if instructions for how to run the system are included.

Important! All source code should be properly attributed. Code written by the team should be clearly marked as such (e.g. by a comment at the top of each file). If code has been adapted by the team for the project then the sources must be clearly identified. Any system components not written by the team (e.g. libraries beyond standard libraries) must be properly identified. Any software licenses that apply to third-party components must be followed and be compatible with the project.

Assessed aspects of the submitted software include, but are not limited to:

- Functionality and features
- How well the developed software actually works
- The size and difficulty level of the addressed problem(s)
- The technical sophistication of the developed software
- How well the software is written (architecture, proper modularisation, proper abstractions, naming conventions, layout, documentation, etc.)
- Quality of the testing

Evidence taken into account includes the submitted software itself, information from appropriate parts of the submitted reports (including the testing appendix of the final report), the recordings, Open Day demonstrations, testing carried out by the first and second markers themselves (e.g., during the Open Day), insight gained from the team presentations and ensuing discussions and Q&A.

The submitted software should be easy to read and understand. Appropriate documentation for a (potential) maintenance team should be submitted, which may include:

- Summary of quality assurance
- Environment requirements (operating system, hardware, etc.)
- Installation instructions
- User manual

11.6 Individual Report

Individual reflection on the project is important for getting the most out of it. This is the primary role of the Individual Report. Additionally, this report contains each student's assessment of his or her peers. Each student must submit one Individual Report at the end of the project, written by his/herself. It should be 2000–2500 words (around 6–8 pages), excluding the peer-marking part and any appendices.

The main parts of the Individual Report are:

- Summary of own individual contribution to the project.
- Reflection on the project, the running of it, and own role within it, including honest and insightful self-assessment.
- Peer assessment.

The reflection should cover aspects such as what was achieved, what was not achieved and why, critical discussion on the running of the project, what you have learnt from the project, and what you would do differently if you were starting over again.

The peer assessment is carried out on a dedicated form. The form, in various formats, is linked from this document. Each student must evaluate all members of the group except themselves. Additionally, each evaluation must be justified

by a written statement on the performance of each peer. See Peer Assessment for further details, including links to the evaluation form in various formats to facilitate the inclusion of the completed forms with the Individual Report. (If none of the supplied formats is suitable, printed, completed forms can be scanned for inclusion.)

Note that Individual Report is marked primarily on the quality of the reflection and on the qualities as a report, not on the extent of contribution as such to the project: see Assessment Guidelines above.

The front page of the report should include:

- The text “COMP2043.GRP Individual Report”
- Project title
- Date
- Team name/identifier
- Your own name and School of CS username
- A list, clearly labelled “other group members” of the other members of the group
- Name of supervisor

11.7 Marking of Other Components

11.7.1 Team Project Website

The Team Project Website is used to allow the supervisor and module team to monitor the team. It is marked very simply based on having the necessary information available by the first deadline, and by having updated relevant information throughout the duration of the project. The necessary website information includes:

- Team name

- Project title
- Supervisor name
- Any other relevant information (such as links or explanation or technical details).

11.7.2 Demonstration

The Demonstration (live and recording) is marked primarily on how well the implemented system is presented running live, and how the team responds to requests/questions about the implemented system.

11.7.3 Presentation

The Presentation (live and/or recording) is marked primarily on the quality of the presentation, and how the team demonstrates their final project (including any implementation).

11.7.4 Questions & Answers (Q&A)

The Q&A is marked primarily on how the team handles questions and answers.

11.7.5 Promotional Digital Artefact

The Promotional Digital Artefact is a digital object (like a short video) that will promote the final delivered system. It will serve as an advertisement, and may be used by the School or Campus as part of promotional materials. The Promotional Digital Artefact is marked primarily on how professional and attractive (and accurate) the content is.

11.8 Submission of Deliverables

Please check the deadlines for each deliverable carefully.

Recall that the individual report must include the peer assessment as an appendix.

Unless specified otherwise, each deliverable should be handed in no later than 3 pm on the due date. Electronic copies are submitted through Moodle, with some secondary submission points (e.g. Microsoft Forms, Panopto, etc) — all submission point details will be in Moodle.

You are also strongly advised to avoid handing in at the very last minute, as any delays (for example, due to long queues to the printers, or heavy loads on the servers) are entirely your own responsibility, as such problems are not unlikely and thus should be taken into account when planning.

In some situations, the submission mechanisms may have limitations on the file sizes or types. If your team's intended submission is larger than the maximum file size, you will need to prepare an alternative submission, and get the agreement of the convenor in advance — this may mean submitting a DVD or USB device (that will not be returned) in person, at a specific time/location agreed upon with the convenor.

12 Open Day

The group project Open Day is a chance for the groups to show off their finished applications to the other groups, to students from other years, and to members of staff. We take over the labs for a day, and each team is given a small area in which to set up a “trade stall” with posters, leaflets, and a live demonstration of their application. Note that it is the overall quality and professionalism of the stalls that is assessed during the Open Day, not just the application itself. To make the day more fun, most people dress formally for the Open Day.

To give you a flavour, there are some photographs from previous Group Project Open Days in Figure 1.

12.1 Open Day 2024–2025

The Open Day date for this year has not yet been confirmed, but may be on 16 April, 2025 (9 am – 6 pm.) — you will be updated when the arrangements are



Figure 1: Open Day Photos

confirmed. Teams will have access to the room and poster boards to set up the Open Day from 6 pm on the day before.

12.2 Presentations

Each team must prepare a presentation, which will be submitted as a recording before the Open Day. The recorded presentation should be ten minutes long. The main content of the presentation should be a description of the problem addressed by the project, an overview of the application developed, and some reflective remarks on the success of the project. Note that it is the quality of the presentation itself that is assessed, not the quality of the project or the application.

Many students may not have given presentations before, so prior to this part of the group project there is a support session on this topic. This session gives some suggestions for what to put into your presentation, what to put onto your slides, and how to speak well.

To give you a flavour, there are some photographs from previous Group Project Presentations in Figure 2.

13 Live Q&A

During the Open Day, usually after the stalls component, all teams will be grouped into sessions for live questions and answers (Q&A) about their project. The live Q&A lasts up to 10 minutes, and can be started by the team giving a 1-minute introduction to themselves and project. This live Q&A session mimics the kind of thing that happens in real conferences.

Each of these sessions will have a member of staff acting as a *session chair* who will give warnings about time-keeping. Just like in real conferences, teams must very strictly adhere to the time limitations.

14 Golden Rules

Below are some tips based upon the reflective comments in the final reports from previous years group projects. Read them carefully; they are the golden



Figure 2: Presentation Day Photos

rules for a successful project.

- Work consistently throughout the project. Start work on your project quickly, and don't leave all the real work until just before the deadlines.

The group project module is worth 20 credits, each of which corresponds to 10 hours of time according to the official University Guidelines, so you should expect to spend around 200 hours in total on your project, which averages out to around 9 hours per week during term time. If you are not spending this amount of time per week on your project, then you are not working hard enough, and will have problems later on.

- Make sure that you have enough to do. Don't wait to be assigned tasks

— volunteer for things that you find interesting. Keep in mind that your peers' assessment of you will have a major impact on your final mark, so make sure that you are seen as a valuable group member by taking an active role in the project, striving to be co-operative and keeping the interest of the group and the project foremost, and working hard.

- Don't leave the programming until too late. It is very easy to spend too long designing your application, and then find that you don't have enough time to complete the programming. In particular, there are often problems linking the different parts of applications together. Make sure that the interaction between parts is well documented before programming begins in earnest, and don't leave linking the parts together until the last minute. Aim to have a prototype up and running as soon as possible, and certainly by late November. If you do not have any code running by then, you should probably be worried. Prototyping is also a great way to learn about what you are doing, and can thus have a significant impact on the design of the project. Don't be afraid to throw away a prototype once it has served its purpose and you understand how you should have approached the problem.
- Do use proper tools to avoid wasting time and effort. In particular, do use a version control system.
- Don't think that you can do the write up in just a few days. On average, 60% of the mark is for the written reports.
- Keep track of everyone's progress. Make sure that everyone in your group is working consistently on the tasks they have been assigned. Your supervisor can provide some guidance with managing your project. Ultimately, however, it is the responsibility of the group themselves to monitor the progress of each member, to ensure that tasks are being carried out as required, and to take remedial action as and when appropriate.

- Take the meetings seriously. Without proper formal and informal meetings each week, your project has little chance of being successful. Spend some time preparing for each meeting, and make your own notes during the meeting in addition to the minutes. Note, however, that most of the real work on the project will be done outside of the meetings. Students whose only work on the project is to contribute to the meetings will find that they have little to write about in their reports, resulting in poor marks for the module.
- Keep good records of your meetings. Collecting your minutes together on a web page is a good way to ensure that they do not get lost, and that everyone can read them. Use the minutes to ensure that the group doesn't discuss the same points for weeks on end, that decisions made are not forgotten about, and that real progress is being made.