University of BRISTOL

Project Handbook

Department of Computer Science

October 2021

Contents

0 How to Use this Handbook **4**

1 Introduction **5**

1.1 Principles and ILOs . . . 5
1.2 Timeline . . . 5
1.3 Types of Project . . . 7
1.4 Structure and Length of your Thesis . . . 7
1.5 Deadlines, Extenuating Circumstances, and Mental Health . . . 8

2 TB1: Preparatory Work **11**

2.1 Choosing a Project Idea . . . 11
2.1.1 Enterprise Projects . . . 12
2.1.2 Research Projects . . . 13
2.1.3 Theory Projects . . . 14
2.1.4 Development Projects . . . 15
2.2 Finding a Supervisor . . . 15
2.2.1 Getting In Contact . . . 16
2.2.2 The Project Specification and Preparatory Work . . . 17
2.2.3 What to Expect from your Supervisor . . . 17

3 TB2: Completing your Project **19**

3.1 Project Management . . . 19
3.1.1 Planning your Schedule . . . 19
3.1.2 Normal Life . . . 20
3.1.3 Mental Health . . . 20
3.2 Ethics: You Probably Do Need To Think About It . . . 22
3.2.1 Projects Not Requiring Ethical Review . . . 22
3.2.2 Projects covered by Ethics Application 0026 . . . 22
3.2.3 Projects Requiring Ethical Review by the FREC . . . 24
3.3 Innovation Case (Fourth-Year MEng only) . . . 24
3.4 Project Poster Day and Second Marker Meeting . . . 25
3.5 Writing your Dissertation . . . 25
3.5.1 Structure . . . 25
3.5.2 Formatting . . . 27
3.5.3 Style and Length . . . 28
3.5.4 Group Projects . . . 29
3.6 Procedures . . . 29
3.6.1 Changing Your Project from your Specification . . . 29
3.6.2 Changing Your Supervisor . . . 29
3.6.3 Changing Your Second Marker . . . 30
3.6.4 Extenuating Circumstances . . . 30
3.6.5 Mitigations . . . 31
3.6.6 Intellectual property (IP) Rights . . . 32

4 Vivas and Marking **34**

4.1 The Marking Process . . . 34
4.2 Vivas . . . 35
4.3 Project Assessment Criteria . . . 36
4.3.1 Challenge and Achievement . . . 36
4.3.2 Technical Approach . . . 37

4.3.3 Critical Interpretation 39
4.3.4 Writing and Presentation 40
4.4 Innovation Case Assessment Criteria 41

5 Academic Integrity and Plagiarism 43
5.1 Turnitin 43
5.2 Correct Referencing: Quoting and Paraphrasing 44

0 How to Use this Handbook

This is a long handbook and you are not meant to read all of it in one go. You should **read all of Section 1 (Introduction)**, and then use the rest of the handbook as a reference manual, reading sections as and when you need them.

1 Introduction

Your final project is the pinnacle of your degree. It is a chance for you to demonstrate what you have learnt and what you are capable of, and makes up a significant part of your degree classification. Your final project will be challenging, and you will need to demonstrate that you can overcome this challenge.

1.1 Principles and ILOs

The three principles you should bear in mind for projects are:

1. You will be performing **self-directed, independent work**. Unlike taught units where the unit director sets the topic, the exercises and the assessment, you are responsible for every part of the project, and you must show that you can take ownership of the project and everything around it (e.g. planning, writing up, submitting on time). This makes the project both the most challenging and the most exciting part of the degree; it is the unit that gives you both the most responsibility and the most freedom in your whole degree. It is **your** project.

2. You will be working **in depth, on a significant challenge**. You will agree the project aims and an outline with a supervisor before you start, and they will make sure that the level of challenge is appropriate and the topic is relevant for the field of Computer Science. It may be the hardest piece of work you have done so far, and ideally it will be something you can be proud of at the end.

3. You will need to demonstrate a number of **Intended Learning Outcomes** (ILOs). All units have ILOs, and the ones for the project are:

- Demonstrate understanding of research-level material.
- Identify a well-motivated, scientifically interesting challenge.
- Engage in a suitable approach to solving said challenge
  (e.g., developed a proof, analysed an algorithm, implemented a system).
- Critically evaluate your solution in a suitable manner.
- Present your results in written and verbal form.
- Manage your own time, and decide your own objectives.

These learning outcomes are designed to match what employers are looking for in Computer Science graduates, which includes managing a large project as well as delivering the outcomes.

1.2 Timeline

Your project will run in three phases:

- In TB1, you **specify** your project, that is you agree a topic with a supervisor.
- In TB2, you **execute** your project and write it up as a **thesis**.
- In the summer assessment period, we **assess** your project, by reading your thesis and asking you questions in a viva (oral exam).

In more detail, the timeline for the 2021–22 academic year is as follows:

Week	Item
1	Watch video lecture on project choice by the unit director. Start thinking of ideas.
2	Q&A session with unit staff.
3	Q&A session with unit staff.
4	"Project specification" system for choosing a topic and supervisor opens.
8	Deadline for choosing a topic and supervisor (before the 3-week coursework period).
13	Watch video lecture on carrying out your project by the unit director. Start work on project.
14	Q&A session with unit staff.
19	MEng only: Innovation Case deadline.
22	Project poster day. Optional, but recommended.
24	Hard deadline to submit your thesis.
26–28	Vivas take place (exact dates TBD).

There may be additional workshops and videos, and if so these will be published on the unit page. We try our best to keep this handbook up to date every year, but there may be changes beyond our control after it is published. In case of differences between information in the handbook and the unit page, the information on the unit page is what counts.

In TB1, you need to choose a project topic and to find a supervisor who agrees to supervise you on this topic. Occasionally, a student has two co-supervisors; in particular, every project must have at least one supervisor from the computer science department, so if your main supervisor is from industry then you will need a co-supervisor. You then confirm your supervisor by submitting a project specification with the agreed-on information. The deadline for this is set just before the start of your 3-week assessments; generally, if you don't have a supervisor in place by then, you will have a lot of difficulty finding one.

You can do some preparatory work for your project in TB1, but overall your main focus this teaching block should be on your other units. Section 2 of this handbook contains information useful in TB1, including advice on choosing a project and details of the project specification process.

In TB2, your main objective will be to complete the project and write it up into a thesis (also known as a dissertation). Importantly, this dissertation will be the part of the project your markers focus on; this means that **a project cannot get a good mark without a good dissertation**, and you cannot afford to leave it to the last minute. If you are taking the fourth-year MEng unit, you will also complete an Innovation Case worth one-eighth of the total credit (5 CP of the 40 CP unit), in which you will explain your work to an external audience. Section 3 contains information useful in TB2, including advice on managing your time well and on writing your dissertation.

Your project will be marked independently by two academics from Computer Science. The first marker will be your supervisor; if you have two supervisors and they are both from Computer Science, then you should agree which of them will be first marker before submitting your project specification. The second marker will be another academic who is independent of your project. Where possible, we choose academics with a strong background in your project area to be your second marker.

In the summer assessment period, you will have a viva (i.e. an oral examination) lasting 20–25 minutes with your second marker, in which they ask you questions about your project. The exact time will be arranged by the university and you must be available at the time chosen for you.

Both markers will mark your project independently, then discuss your project until they agree on a final mark. Section 4 contains detailed information about vivas and the marking process, including the overall mark scheme. If you are a fourth-year MEng student, you will be marked to a higher standard than BSc students. If you are a third-year maths and computer science MEng student taking the 20cp "short project unit" COMS30044, then you will be expected to do about half as much work as a student taking the normal 40cp unit.

1.3 Types of Project

All projects must have something to do with Computer Science, but beyond that it is up to you and your supervisor to agree a topic. There are some standard types of projects that you might want to consider, but your project might have more than one of these types, or (rarely) not fit into this schema at all. More details on these project types are included in Section 2.1

An **enterprise** project creates something that could be turned into a product, company or service. An important part of your thesis in this kind of project is evidence that your project creates some kind of value, and the usual way to get such evidence is to evaluate your project with potential users (with proper procedures and ethical approval). This kind of project lends itself particularly well to working with external companies or other organisations.

A **research** project aims to make an original contribution to human knowledge. It will generally start with reading academic papers, and the ideal outcome is that your project becomes a peer-reviewed publication itself. An important part of your thesis will be placing your work in the context of the relevant research literature (for example, with a literature review and a critical evaluation of your own contribution).

A **theory** project is a specific type of research project that makes an intellectual contribution to a theoretical (often mathematical) subject rather than doing something tangible like testing an algorithm or training a classifier. This could include, for example, summarising the results from several different papers and so making them available to a wider audience, or re-proving a theorem from paper X in the framework of paper Y. As long as the level of intellectual challenge is right and the result has some relevance in its field, theory projects can take many different approaches.

Enterprise, research and theory projects are all equally valued. They pose different kinds of challenges that might be easier or harder for individual students, but academics do not view any of these three types of project as better or worse than any of the others. We do not give higher or lower marks based on the category of project that you chose from these three.

A **development** project produces a piece of software (and/or hardware) that shows off your development skills and becomes an item in your portfolio when you go looking for developer jobs. A lot of focus will be on demonstrating your mastery of career-relevant techniques from particular frameworks (e.g. making a web app in React) to general skills such as effective use of version control, unit testing etc.

A pure development project is **not** equally valued compared to an enterprise, research or theory project. One of the key criteria for a First is that your project should in some way improve on what is already out there, and pure development projects typically do not fulfil this requirement. (Enterprise, research and theory projects can all involve a lot of software development as well, but it is a means rather than an end.) This sort of project is also more appropriate for a BSc student than a fourth-year MEng student. That said, for a BSc student it is no harder to earn a 2.i with a pure development project compared to other project types, so you may want to consider a project of this form if you are not aiming for a First but would to add to your portfolio for a job interview.

1.4 Structure and Length of your Thesis

The version of your thesis that you submit to the university must be a single PDF file. For the 2022–23 academic year, no paper copy of your thesis is required (but you may print some for yourself and your family if you wish).

Your thesis must start with some boilerplate sections known as the **front matter**, such as the cover page and table of contents. Most of these are compulsory, they must appear in a specific order, they much each start on a separate page, and for some of them you must use the exact wording supplied by the university. Details can be found in Annex 14 of the University Regulations (here); alternatively, templates will be provided on the unit page with skeletons of these sections included. If you want to print your thesis, it is traditional to print double-sided but start each of these on a separate sheet.

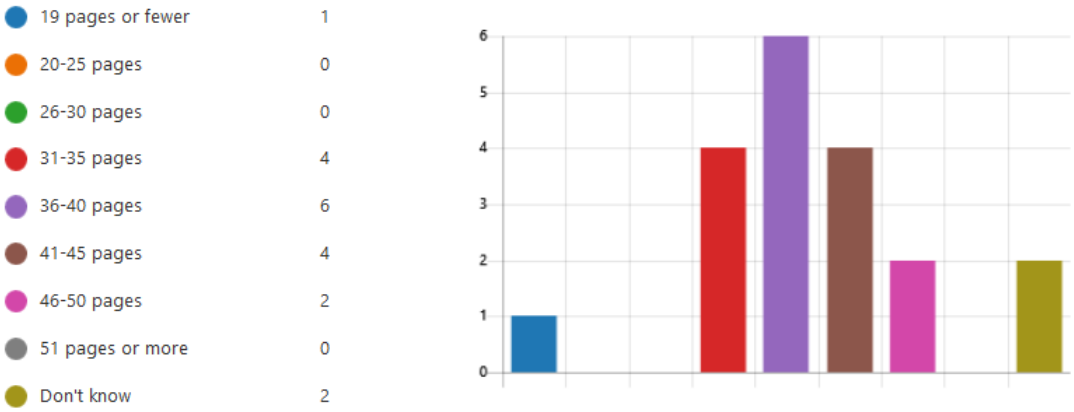

Figure 1: Supervisors' responses to the question: "Think of a project of the type you normally supervise which, ignoring the writing quality, would receive a mark of roughly 65 (i.e. a solid 2.i). If it is well-written and sensibly formatted, how many pages would you expect the *main body* of the project to be? (In other words, not including the title page, executive summary, ethics statement, references etc.)".

One particularly important piece of front matter is the ethics declaration (see Section 3.2). You will need to specify whether your project either a) did not require ethical review, b) was covered by the blanket ethics approval for the project unit, or c) was covered by its own dedicated ethics application. The unit includes mandatory training to help you make this distinction.

The structure of your thesis will vary depending on your choice of project, and the best person to ask about structuring it is your supervisor. That said, most theses start with an **Introduction** section, then have one or more **Background** sections such as contextual background, technical background, previous work etc. The **main body** of your thesis will be one or more sections describing your work, sometimes structured into methods, results and evaluation/interpretation. Most theses end with one or more **Evaluation** sections, and finish with a **Bibliography**.

You may include one or more **Appendices**, but your thesis must be understandable to a reader who chooses not to read them — they are meant for things like extra figures and tables that you have already summarised in the main body of the thesis.

There are no hard minimum or maximum lengths for your thesis. There is a soft maximum of 50 pages for your thesis, not including front matter, bibliography and appendices. However, you should not fall into the trap of thinking that a longer thesis will earn more marks! One of the things you will be marked on is the quality of your writing, and part of good writing is the ability to write concisely without sacrificing clarity. If you can reduce the length of your thesis from 50 pages to 40 pages without removing anything important, then your mark will go up rather than down. The ideal length varies widely by project and by discipline, so you should check with your supervisor, but most supervisors think 30–45 pages is a good length for a solid 2.i (see Figure 1.4).

More information on structure and formatting of your thesis can be found in Section 3.5. The one rule worth highlighting here is you should not typeset your thesis double-spaced (the space between lines of text is twice the height of the text itself). This is required at some institutions, but is discouraged in Bristol.

1.5 Deadlines, Extenuating Circumstances, and Mental Health

Your project demonstrates, to future employers and PhD supervisors as well as to us and to yourself, what kind of work you are capable of doing while dealing with the stress of doing a project and managing your life and health. We are assessing you on how well you can plan your project and whether you can deliver it on time whatever happens during

term, as well as the quality of what you deliver.

You should assume that **not everything will go to plan** and that it is your job to deal with this; you should further assume that **you will not get an extension on your thesis deadline**. There is an extensions policy, but it is for things like becoming a victim of violent crime and ending up in hospital, not for things like your data analysis taking longer than you expected — even if the delay to your project makes you feel extremely stressed and worried.

There are also logistical issues with long extensions on our end. The longer an extension you have, the less time we have to mark your thesis and the less time your second marker has to prepare for your viva. In the worst case, this may result in your graduation being delayed. A one-week extension is always fine. A two-week extension is *usually* OK, but not if your second marker falls seriously ill and your backup marker breaks their arm and they both had 500+ pages of projects to mark. (This actually happened.) A three-week extension is playing with fire, and a four-week extension is out of the question. At the time of writing, the university is considering new extension rules which would ban us from implementing this sort of "at-your-own-risk" long extension — **if this goes through, then we will be forced to defer graduation for all projects with an extension longer than one week**.

The way you get your project done on time despite unforeseen problems is to plan for them from the start, by following these guidelines:

1. Pick a project that is within your ability to complete, and be realistic about this. Within your ability means that the inherent stress and challenges from the difficulty of the project will not cause you mental health problems, even when (inevitably) not everything goes to plan. It might be better to aim for a comfortable 2.i than "First or bust", an attitude that has in the past led to students failing their project or having to drop out of university for health reasons. Your supervisor will help you with this.
2. Have a fallback option. A resilient project plan starts with a Minimal Viable Product (MVP) that is well within your ability and that would take significantly less time to deliver than you have for the whole project (one half to two thirds of the time allocated to the project is a good starting point), and which can then be gradually extended into your full project.
3. During TB2, you should regularly (at least weekly) review your progress and remove features or change the scope as necessary if you are falling behind schedule, or add extra items to your backlog if you are ahead of schedule. You should also meet regularly with your supervisor to discuss your progress.
4. You must leave enough time to write up your thesis, including any evaluation sections — it is not a valid excuse that the actual work took so long you didn't have time to write it up properly. You usually want to start thinking seriously about thesis-writing around one month before your deadline.
5. If your project is not going well or you are worried about your (physical or mental) health, you must speak to someone as soon as possible. This could be your personal tutor, your project supervisor, the wellbeing service, your GP, or friends and family members who you trust.
6. As a rule of thumb, you should plan your work on the assumption that you will be able to dedicate roughly **3.5 days a week, 8 hours a day, for 12 weeks of term** to the project (with the rest of the week going to your other TB2 unit). If you're taking COMS30044, halve this. You will very likely put in more time than this — most people work on their projects over Easter — but you will also very likely *need* more time than this, since almost every project suffers from unforeseen problems and delays. Also, if your plan requires that you work more than 40 hours per week for an extended period, chances are that at best you will fail to accomplish this and at worst you will burn yourself out completely.

More detail on this is in Section 3.1.3.

If you have had extenuating circumstances (EC) relating to your health in the past, you should consider before you start your project how you will manage any problems that might arise, and you may want to speak to a senior tutor or wellbeing adviser or to seek professional help. If you feel comfortable doing so, you could also talk to your project supervisor at the very beginning of the project.

If you have a registered disability that may affect your project, you should speak to Disability Services about this before you start your project. They may be able to offer you extra support, even if you have not needed it in previous years.

2 TB1: Preparatory Work

The project unit is designed to allow as wide a variety of projects as possible. Your project can be anything you can find a supervisor for, as long as it is clearly Computer Science in some way. Your project might be highly theoretical; it might be focused on running user studies; it might be focused on building a piece of software or hardware with an immediate application; it might be all of the above, or it might be none. To give you an idea of the breadth of possibilities, here are some examples of highly successful projects from the 2020–21 academic year:

- Analysing how COVID-19 has affected people's privacy practices online;
- Applying self-attention networks to the problem of automatically reading handwritten text;
- Applying qualitative coding to documents released by Google to analyse their stance on AI ethics and governance;
- Automatically detecting wildlife from camera trap footage;
- Creating a new domain-specific language for dataflow programming in Haskell;
- Designing and testing physical tools to teach programming to visually-impaired children;
- Implementing a newly-developed fair redistricting algorithm to test its performance on real-world data;
- Writing a detailed literature review of the CSI-FiSh signature scheme in cryptography and comparing it to other alternatives.

More examples are available on the unit site. This section will go into detail about how to choose a project idea (Section 2.1), how to find a supervisor (Section 2.2), and what to do for the rest of TB1 (Section 2.2.2).

2.1 Choosing a Project Idea

Most people choose a project idea in one of two ways. First, supervisors add project ideas to a spreadsheet which we will release around Week 4 and you can then contact anyone whose idea you like; secondly, you can come up with an idea of your own and contact anyone you think might be a good supervisor for it.

Both approaches can lead to outstanding projects, and neither one is inherently more or less likely to yield a high mark. The main advantage of coming up with your own idea is that you get to implement your own idea; if this doesn't sound like an advantage, then you should just choose something from the spreadsheet! The main disadvantages are that it will be a little harder to find a supervisor, and that you'll need to do more preparatory work in TB1 to refine your idea and make sure it's feasible (which your supervisor will help with).

If you are a single honours computer science student, then you also have the option of doing a group project. Unlike the group projects you have carried out in previous years, we do not provide support in finding group members or in mediating conflict inside the group; this option is intended for groups of students who already know each other, already work well together, and may be considering forming a start-up after graduation. Group projects are assessed slightly differently from individual projects, as detailed in Section 3.5.4. Naturally, if you work in an n-person group then we will expect you to produce n times as much!

If you decide to come up with a project idea of your own, here are some questions you should consider:

- Which units did you enjoy and do well in last year? Which units are you currently enjoying? Can you choose a project that makes use of those units?
- Are there any academics in particular you'd like to work with?
- Your project will be an important part of your CV after graduating. Will your project be relevant to your planned career path?
- If you find you can't get your project done in time, can you scale it down into something smaller but still reasonably complete?
- Do you have a backup plan for if you are denied access to university labs, e.g. due to COVID restrictions?

Most projects naturally fall into one or more of the following categories:

- An **enterprise** project aims to develop and/or evaluate a product or service that could lead to becoming a "real thing" with actual users.
- A **research** project aims to make an original scientific contribution to the body of knowledge in Computer Science.
- A **theory** project aims to make a contribution to a theoretical (usually mathematical) area of Computer Science.
- A **development** project aims to train and show the student's ability as a software developer, with a view to applying for such a job with the project forming part of their portfolio.

Enterprise, research and theory projects are equally valuable; for example, the Department does not weight research higher than enterprise. However, one of the key criteria for a First is that your project should in some way improve on what is already out there, and pure development projects typically do not fulfil this requirement unless they involve an extraordinary level of technical challenge. (Enterprise, research and theory projects can all involve a lot of software development as well, but as a means to an end rather than an end in and of itself.)

2.1.1 Enterprise Projects

An enterprise project might be for you if you want to start your own company, or take on a leading role in a start-up. You should be interested in design, development and innovation. An enterprise project may involve any combination of software and hardware or products and services as long as there is something relating to Computer Science involved.

An enterprise project's outcome may be commercial, non-profit, open source or another form that is appropriate to the project. We expect you as a student to choose, describe and justify a form for the outcome of your work.

A successful enterprise project ends up with real users, and provides real value for its users. In the words of Silicon Valley venture capitalist Paul Graham of Y Combinator[1]:

> "In a sense there's just one mistake that kills startups: not making something users want. If you make something users want, you'll probably be fine, whatever else you do or don't do. And if you don't make something users want, then you're dead, whatever else you do or don't do."

If you are doing an enterprise project, we expect you to have a clear idea of your target user group (or develop it as you go along), to have access to some potential target users, and to develop your project together with these potential users and evaluate your project with some form of user study. Even if you can't prove that your project is something that users definitely want, you should at least show in your thesis that you have tried something on real human users — even if it is only an early prototype that you are evaluating.

This implies that you should think of how you will get access to potential users and testers before you pick your topic, and whether there are any legal or ethical implications. For example, if your target users are A-level students then you will need at least an agreement from some school to support your project before you even start, and a full ethics application as your users are under-age. You will always need ethical approval for any kind of user study, but how much work this is depends on the kind of study and the kind of users, and in some cases you may be able to use a blanket ethical approval for the project unit; see Section 3.2 for details.

An enterprise project may involve ongoing research and technology development at the university, or an industry partner or sponsor. You might work with an external company or non-profit organisation, as long as everyone understands that you as the student are responsible for the project to the extent necessary to align it with the university's expectations and marking schemes.

[1] `http://paulgraham.com/startupmistakes.html`

The university does not own the intellectual property produced by your project, and you as a student can sign a non-disclosure agreement with a third party if you like, but your supervisor will not normally sign any agreements with third parties and you must be able to write up enough in your thesis that is sufficient to award you a mark without being commercially secret.

Key questions you want to answer in your thesis for an enterprise project include: Who is this for? What value does it add for them? How close is it to being a "real thing"? And what is the evidence for all of this? An ideal outcome for an enterprise project is something that is already being used productively, by real users, by the time you hand in your thesis (with evidence of this).

2.1.2 Research Projects

A research project could be for you if you are interested in discovering and presenting new ideas and contributing to the body of human knowledge that is Computer Science. It is an obvious choice if you want to stay in academia, for example to do a PhD, but a great deal of research takes place in industry as well and a research project can also be a great preparation for a graduate career outside of academia.

Research projects can take many forms including designing and evaluating theory, algorithms, models, software or hardware. Your work could include proving mathematical results, building software and/or hardware, fixing a gap or mistake in the literature, synthesising work from different publications, improving on previous results and many other approaches. Anything that makes an original scientific or technical contribution and fits within the field of Computer Science is a potential research project.

For a research project, it is important to find a supervisor who is familiar with the research area in which you will be working, and who knows the conventions and trends in the field (for example, which of several journals in the field is the best fit for your project). Some supervisors will have suggestions of their own for research projects that could lead to a co-authored publication. Different fields will also require different levels of background reading to transition into research work, and your supervisor will be able to help estimate how long it will take you to get up to speed and what sort of problem you'll be able to tackle in the remaining time. Taking on a project which requires a lot of background reading will not put you at a disadvantage relative to other students.

In your thesis, it is important to place your own work in the context of previous work and literature. This usually happens in two ways. First, typically in a background section near the start of your thesis, you survey previous work. Second, typically in an evaluation section near the end of your thesis, you compare your own work against the standards or metrics of the field you are working in and compare your results to previous ones. Being able to write this section accurately is almost as important as the quality of your results themselves — a project that didn't go well with an evaluation section that explains what went wrong and why will get a significantly higher mark than the same project with an evaluation section that tries to argue that actually everything went perfectly and that the flaws were all according to plan. (See Figure 2.)

For a practical example, if your work involves using machine learning to recognise individual penguins in Bristol Zoo, then your background section could survey existing work on animal biometrics, especially if there is already existing work on penguins[2]. Your evaluation would take a format appropriate to machine learning (size of the training set, relevant parameters, precision/recall etc.) and compare with the results of other work recognising the same or different species of animal.

The ideal outcome of a research project is something that could be submitted to a leading academic conference or journal, or in the best possible case, your work has already become accepted as a peer-reviewed publication in a reputable venue by the time you submit your thesis. Note that the time and skills needed to produce something publishable vary widely by field (some journals can take longer to review a paper than you have for the whole project for example, whereas

[2] Such as the work by our own Tilo Burghardt.

Figure 2: An example of how not to approach your project's evaluation section, even when things could have gone better.

conferences usually have a much shorter turn-around time) and your supervisor will able to give you advice on this, as well as take it into account when marking your work.

Not all strong research projects will lead to publishable work, and this is field-dependent. One of the key descriptors for a project with a mark of 80+ (i.e. a very strong First) is that it should be "consistent with the quality of a good early-career PhD student or an employee at a high-ranking institution". In some fields, PhD students are expected to publish early and often, and in these fields it might be expected that a project should be publishable in order to obtain a First. In other fields, particularly mathematical fields, new PhD students are expected to spend their first few months catching up with the methodology of the field and are expected to publish perhaps a single paper before the end of their first year, and in these fields a project would definitely not need to be publishable in order to obtain a First. These projects are very often theory projects (see below), and this is something you should discuss with your supervisor.

2.1.3 Theory Projects

A **theory** project is a specific type of research project which focuses on theoretical results rather than a tangible end-product. This could include, for example, summarising the results from several different papers and so making them available to a wider audience, or re-proving a theorem from paper X in the framework of paper Y. As long as the level of intellectual challenge is right and the result has some relevance in its field, theory projects can take many different approaches.

The main difference between a theory project and a research project is that a theory project is usually much harder to evaluate objectively: for a machine learning application you can measure precision/recall etc., but there is no objective measure for how interesting a theorem is. It is therefore especially important that you discuss with your supervisor how to present the value of your work; if your work would be of interest to other academics in the relevant area then that is a good project.

Theory projects normally have a mathematical flavour, in the widest sense of the term that includes for example research into programming languages. There are many good project opportunities in this area that are undoubtedly Computer Science, but do not involve writing any code at all!

Theory projects are often carried out in areas that require an unusually large amount of background reading before any research can be done, to the point where even a new PhD student would normally spend one or two months reading before starting work. In particular, reading and writing mathematical papers are difficult skills that take time to learn, and that most students have not been previously exposed to. This means the bar for an 'original scientific contribution' in a theory project is a little lower than for a pure research project, or at least the scope is wider — all of the following could count as original and would be capable of an outstanding mark, as long as the overall difficulty level is right:

- Express the results of paper X in the notation or framework used by paper Y.
- Put the proofs of results in papers A, B and C into a common framework, unifying notation and definitions.
- Simplify the proof of a theorem from paper W by using more modern techniques.
- Correct a broken proof in the literature, or complete a proof the original authors simply left out of their paper. (This happens more often than you might imagine![3])
- Present the work of several papers, or perhaps even a single paper, in a way that is accessible to undergraduates or non-experts in the field when the original papers are written in a way that is hard to understand. (This too happens quite often.)
- A literature review of one or more fields of theory, as long as it has sufficient scope and depth.

Many of these types of project come under the theme of 'Systematisation of Knowledge (SoK)' or similar terms (such as 'review papers') and some conferences and journals welcome such papers, or have dedicated conference tracks for them.

2.1.4 Development Projects

A development project is one where you develop some software to show off your skill as a developer and to use as a portfolio item when applying for developer jobs. While research and enterprise projects often involve a development component, the software is a means to an end and the outcome is often what counts: if you get an interesting scientific result using code that does not follow best practices in the field, the result does not count any less.

In contrast, in a development project you want to demonstrate your mastery of practices that could include version control, unit testing, refactoring, structuring code using object-oriented, functional or other design patterns, using standard components and libraries for sub-tasks rather than re-inventing the wheel, using frameworks such as React or Angular that are sought after by employers, paying attention to cross-cutting aspects such as security, privacy and legal requirements (e.g. GDPR) and anything else that could fall under the general heading of professionalism. It is not a mistake to apply all of these practices in a research or enterprise project too, but in a development project you will be evaluated on them in more detail.

The difference between a development and an enterprise project is that in a development project, the focus is more on the quality of your code and in an enterprise project, more on the usefulness, impact and evaluation of your final deliverable. Thus, a development project without an enterprise or research theme is considered slightly less valuable from a University point of view, is more appropriate for a BSc than an MEng, and may not be the right choice for a student aiming for a first-class degree. However, a development project is valuable if you are looking for a job as a software developer and want both some time to practice your development skills or learn new ones, and an extra item in your portfolio to demonstrate to future employers.

2.2 Finding a Supervisor

In roughly week 4 of TB1, we will release a spreadsheet containing a list of possible supervisors. For each supervisor, it will contain:

[3] Especially in Cryptography.

- A brief description of their supervision style;
- A list of sample projects;
- A list of keywords that tell you what sort of other projects they like to supervise;
- The number of additional project students they are able to take.[4]

You are free to approach individual supervisors before this point if you have a specific project idea, but we expect that most people will wait until the spreadsheet is available. It is then your responsibility to decide on a project idea (if you haven't already) and get in contact with potential supervisors. Student positions are normally **first-come first-served** — some supervisors may be willing to go above their nominal capacity for students, but this is entirely their decision. If you want to do a machine learning project, but none of the supervisors who can supervise machine learning projects have room for any more students, then you won't be able to do a machine learning project! Many sample projects are also only available to a single student. Generally supervisors are more likely to be willing to take extra students if they have a detailed project proposal and a good academic track record, but this is by no means a guarantee.

If you have an idea for a project that requires working with someone from outside the Department of Computer Science, such as a lecturer from elsewhere in the university or an external company, this is perfectly fine. However, you will still need a supervisor from within the department! They will act as a "caretaker co-supervisor", helping with questions specific to this project unit but otherwise leaving you free to work with your main contact. (For example, your co-supervisor will be able to advise you on writing up your dissertation.) If you have a project lined up but are having trouble finding a caretaker co-supervisor, then you should get in contact with the unit director and they'll help you find one.

2.2.1 Getting In Contact

The best way to get in touch with potential supervisors is by email. You should always feel free to ask questions — no-one wants to take on a project student and then discover a week into the project that actually they hate it! After you make contact, most supervisors will then set up a meeting to discuss the project in more detail. This might be a one-on-one meeting or a small group meeting with other interested students. If they agree to supervise you, you'll then fill out a short project specification (see Section 2.2.2) which they'll sign off on, and you'll spend the rest of TB1 and maybe part of the winter vacation doing a little preparatory work alongside your other units.

Unfortunately, your supervisors' inboxes are a thing of nightmares, and it is fairly common for academics to receive 50–100 emails per day. Sometimes an email gets lost in the deluge. For this reason, if an academic doesn't respond to your message within two working days[5], you should assume it has been lost and send a follow-up email rather than waiting for weeks for a response that will never come. (Their response might be an out-of-office message if they're ill or on leave, or it might be a holding email if they're unusually busy, but it should exist!) If they also don't respond to that follow-up mail within two more working days, then you should get in touch with the unit director and they will begin prodding more firmly in case something serious has happened.

Each academic can only supervise so many people, and people in popular areas will often have more students applying for projects than they have space for — in this case, you may well not get your first choice of supervisor. However, from the start of Week 7 onwards (i.e. two weeks before the specification deadline), if a potential supervisor has capacity free, then the default is that they will take on any interested students as long as it is in the students' academic best interests for them to do so. Some reasons it might *not* be in your best interests include:

- The project requires specific TB1 units (or year 3 units for MEng students) that you don't have.
- The project is unusually difficult and your academic track record is not good enough to support it, so you would be likely to get a third or failure on this project where you could get a 2.ii or 2.i on another project.
- The project is unusually easy and your academic track record is very good, so you would be likely to get a 2.i on this project where you could get a First on another project.

[4] This will be automatically updated as they accept students.

[5] That is, two days not including weekends and holidays.

- You're applying for a project they didn't list as a sample project, and they don't have the expertise necessary to supervise it well.
- You're applying for a project they didn't list as a sample project, they don't think it's feasible, and they can't see a way of reducing the scope to make it feasible. (In which case you should listen!)
- There is a pre-existing relationship between the two of you (e.g. tutor and tutee) which they feel would stop them from acting as an effective supervisor.

If you are unsure why a potential supervisor has refused you, or if you think it's not in your best interests, then you should get in touch with the unit director.

2.2.2 The Project Specification and Preparatory Work

Once you have decided on your project and found your supervisor, you will need to submit a project specification. This will be a short form that is intended to take no more than ten minutes to fill out; the most important pieces of information are your project title, a brief one- or two-sentence description, and a choice of subject keywords. The project specification is formative and does not count towards your unit mark, but it is still important, for three reasons:

1. Your project will have two markers. One of them will be your supervisor, and we will choose the other one based on your project specification, **especially the keywords**. If you have two supervisors, you should discuss which one will mark it with both of them before submitting the project specification. (See Section 4 for more details on the marking process.) While your second marker will not know as much about your project area as your supervisor, it's very still helpful for them to come from a related field. We need to assign markers very early in TB2, and if we don't have a project specification from you by then, we will be forced to guess at your area; this could impact your mark.
2. Submitting the project specification represents a commitment to working with your supervisor, which they must formally approve. This removes any possibility of misunderstandings, and allows us to keep track of which supervisors have the capacity to take on more students.
3. Historically speaking, people who don't find a project supervisor during TB1 tend to struggle very badly in TB2. Requiring students to submit a project specification early minimises the chance that this happens.

The deadline for the project specification is at the end of week 8, to avoid it interfering with your TB1 coursework. If you don't submit it by then, we will assume something has gone badly wrong and begin chasing you both directly and via the tutor system. In order to submit your project specification, you will first need to complete the mandatory ethics training for the unit (see Section 3.2).

When you have found a project and a supervisor, it is a very good idea to do some preliminary work in TB1, especially if you are coming up with your own project idea —- your supervisor should be able to guide you here. This doesn't have to take a lot of time; in fact, it's better not to spend too much time, since this could interfere with your other courses. But especially when it comes to design and preparation, a couple of days of thought spread out over most of TB1 can be worth two weeks of thought in TB2. Giving your ideas time to grow and change at their own pace is a very powerful technique, and you should take advantage of it.

2.2.3 What to Expect from your Supervisor

This section sets out what most students will be able to expect from their supervisors. Some supervisors might do more; this is their right, but not their obligation. Some supervisors in very high-demand areas might do less, in order to be able to take on more students than they're allocated. This is allowed, on the grounds that it's better for a student to have a part-time supervisor on a project they're interested in than a full-time supervisor on a project they hate. However, these supervisors should make this clear to you before you submit the project specification!

During TB1, your supervisor will meet you once or twice to agree on the project and to suggest some preliminary work, but no more than that — at this point you should be focusing on your other units.

During TB2, your supervisor will meet you regularly. On average, they should be meeting with you for roughly an hour every two weeks or half an hour every week. Some supervisors prefer to "back-load" their time with a student, providing more help towards the end as the dissertation gets close to the end. These meetings might be in-person, but only if both you and they feel safe doing so; otherwise, they will be online.

In general, your supervisor will not provide substantial help outside these meetings. Their role is not to solve your problems for you, but to help you solve your own problems by providing "high-level" guidance. Metaphorically, they'll point you down the right path and they'll warn you about the quicksand patches on the way, but they won't hold your hand as you go and they certainly won't carry you on a litter. To take a more concrete example, one algorithms project I was involved with required the student to implement their own variant of a balanced binary search tree without explicit keys (as part of a data structure used in a larger algorithm). I was happy to give a brief explanation of how this could be done and provide links to further reading material, but not happy to spend four hours preparing a bespoke 60-minute lecture on it complete with pseudocode and call it a project meeting.

The overall result is that your contact time with your supervisor will be limited, and you **will** be forced to work independently from them. You simply do not have enough time with them to be able to ask them for help every time you get a compile error or have trouble installing a software package. This is by design. As an advanced computer science student you are expected to be able to work independently, as you will need to in the workplace, and your ability to do so is an explicit part of the mark scheme. In general, you should treat the contact time your supervisor offers as a hard constraint — if you are struggling, then it is not their responsibility to spend more time helping you, it is your responsibility to scale down your project.

Your supervisor should reply to emails within 2–3 working days. This reply might well be "let's discuss this at the next meeting", but it should exist! If it doesn't, then you should assume they've missed the email and try again; if they still don't reply, get in touch with the unit director and they will check to make sure nothing serious has happened.

Your supervisor will typically not respond on Teams or similar services outside of prearranged meetings, even if their icon is green. A lot of academic work requires intense focus and no interruptions, so most people prefer to handle all non-urgent queries via email and check it no more than once per day.

Towards the end of TB2, you can and should spend some of your regular meetings talking about the dissertation, including discussing your current draft. However, your supervisor will not "pre-mark" your dissertation, and they will only read a limited amount outside of your regular meetings. Once during TB2, you can send them a single draft section (up to 5 pages long) for detailed and considered feedback. Preparing this feedback takes a while, so you will need to send this at least one or two days before the meeting itself, and you should discuss the timings with them beforehand.

3 TB2: Completing your Project

3.1 Project Management

Your project is a chance for you to show what you have learnt and what you can do with it –– to the university and future employers or research supervisors. This includes not only the work itself, and the write-up as a thesis, but everything around it including planning your time, your life and looking after your health so that you are able to do the work.

To graduate, you need to demonstrate the level of planning and endurance that you need to complete a 40-credit 12-week piece of work. This is as much a part of your final project as writing a thesis. You need to choose your project so that it gets done on time even when things go wrong, you need to keep going when things get tough, and you need to hit a fixed deadline. These skills are part of what you are being assessed on in the project.

Future employers and PhD supervisors have an obvious interest in how you can perform in the long term, not just on your best days. They will want to know what kind of work you are capable of doing while dealing with the ups and downs of normal life, which is never perfect. You need to demonstrate that you can manage your life as well as your work.

3.1.1 Planning your Schedule

You should plan for a schedule of 12 weeks of term, 5 working days a week at 8 hours/day. Since your project is worth 2/3 of your final term's credit points, this works out at 320 hours budget for your project and another 160 hours for your other 20CP mini-project unit. (If you are a third-year Maths with Computer Science MEng student taking COMS30044, then you should halve these figures.)

You might have heard from other students that many students go over this budget in practice and do work some combination of evenings, weekends and over Easter. You will probably do some of this too, but you should plan for a budget of 320 hours for the project anyway and then you will be doing overtime for the right reasons, not the wrong ones.

The wrong way to plan a project is to assume from the start that you will be working every day until the deadline, including over Easter, and to plan the amount of work you will do around that –– then, when inevitably something happens that disrupts your schedule, you fall behind and fail to complete the project on time and get a resit capped at the pass mark. The way you avoid this is to build enough slack into your schedule that you can use the spare time to make up for problems when they do happen.

The right way to plan and deliver a final project involves at least five steps:

1. You must choose a project that you are sure you will be able to complete on time. Be realistic about your own abilities, and about difficulties you have faced in the past. Do not pick a project that is too hard for you –– the university will not give you an extension to compensate for this.
2. You should set yourself a Minimum Viable Product (MVP) goal that is enough to pass, and that you are confident you can achieve in around 200 hours of work. If the project is much harder than expected, you can scale down to deliver only the MVP. You should discuss the MVP with potential supervisors before committing to a project. Everything beyond the MVP is 'extra features' and you can implement as many of these as time allows.
3. You must plan enough slack into your original project time plan, for example by sticking to the 320 hour limit. Very few projects go according to schedule, and your time plan has to be good enough that you can recover when — not if — you run into unexpected difficulties.
4. You must regularly review your progress, both for yourself and with your supervisor. If things are going well you can add more features to your backlog. If things are not going well you can agree to remove some extra features. If you think that achieving the MVP is at risk, you must tell your supervisor immediately.
5. You must budget enough time for evaluation and for writing up the thesis. These always take longer than you

think, and you want to start writing fragments of your thesis as early as possible. Whenever you complete a task, think about how it fits into the bigger picture and take some notes, or even write a section for your thesis.

The best kind of project is one where you can add features incrementally, using an "Agile" way of working (if you are feeling particularly enterprise-y). The most risky kind of project is the one which might have great value if you get everything done that you plan to, but does not provide much value at all if it is only 90% done; you should not attempt this kind of project unless you are certain that it is well within your abilities and your supervisor agrees. It is also very dangerous to leave a key feature until the very end of development — if the project doesn't work without it, then it should be included in the 200-hour MVP.

3.1.2 Normal Life

Here are some examples of the kind of disruption that counts as normal life, and that you need to be able to deal with. Part of what you are demonstrating in your final project is that you can deal with this and still complete the project on time:

- Some days you will have an important commitment, such as a job interview.
- You might fall unwell with a cold and need a couple of days off.
- Your computer might break, and you might need a few days to replace it.
- There might be an event among your friends or family such as a wedding, and you may need some time to travel to and from the event (e.g. overseas).
- You might break up with a partner and need a couple of days to recover.

These are all examples of disruptions that are not good reasons to extend your deadline. In your project and thesis, you are demonstrating the level of work you are capable of doing on time while dealing with everything that normal life throws at you. This is the minimum level of professionalism that will be expected of you in a job, and that is standard for people with much lower average salaries and worse life outcomes than someone with a STEM degree from a Russell Group university. If they can stick to deadlines, so can you.

The university regulations say that for full-term coursework and projects, anything that causes up to one full week of disruption is normally judged to have a minimal impact on the work and will not be extenuated.

3.1.3 Mental Health

Your mental health is important, and the university will support you with it wherever it can.

Your main contacts for mental health support are:

- The Wellbeing Service, which you can contact at `wellbeing-access@bristol.ac.uk`.
- Your GP, for example at the student health service.
- Other university services and resources, that you can find for example at `http://www.bristol.ac.uk/students/support/wellbeing`.

Your project supervisor and personal tutor can help you to some extent with the academic side of things, and you are strongly encouraged to keep in contact with both of them, but they are not trained mental health professionals and they are also not qualified or responsible for advising you how to manage your life. You may also want to talk to the Senior Tutor if you want to speak to an experienced academic who is not directly involved in your project.

However, it is your own responsibility to do some things to look after your health that no-one else can do for you.

The most important thing you can do is not overwork yourself. In your final project, you will often be working at the upper limit of your ability to demonstrate the best work you can do, and very few people if anyone can keep this up for 15 weeks without interruption. Professional athletes have rest days in between their training and competitions, and you must do the same.

For this reason, you should plan to work 5 days per week, to not go over this regularly, and to never work more than 6 days per week whatever happens —- Sundays are for rest and recovery (or Saturdays, if you prefer). You should not regularly exceed 8 hours of work in a single day.

One thing you can do from the start to support your health is not pick a project that is too hard for you. This could be both because of the topic (if you are struggling with the Machine Learning unit, for example, then do not pick a machine learning project) or because of the scope (trying to deliver 20 features when 15 would be a better target).

It is a serious error to think that you will get better work done just by working longer hours; there is a limit somewhere in the 35-–45 hour per week range that varies per person. Above this, total output actually drops (even simple things start taking longer to do), you make more mistakes, and you increase the risk of harming your mental health or burning out.

Related to this point, you must get enough sleep. Professional drivers and pilots have strict limits on the number of hours they can work per day or week and the minimum hours of sleep required because above its limits, the human brain loses concentration –– if you learn to drive a car, you might be told that going 24 hours without sleep decreases your cognitive ability as much as having more than the legal limit of alcohol in your blood.

Sleeping too little over a longer period of time is particularly dangerous as, according to a 2003 study from UPenn/NIH[6], sleeping only 4–6 hours for two weeks in a row causes a cognitive deficit as bad as no sleep at all for three days in a row, but you are not aware of it: "*subjects reported feeling only slightly sleepy and were unaware of how impaired they were*".

You must never, ever sacrifice sleep for extra work hours above your body's or mind's limits: this will not only lead to less work or worse work getting done in total, but it will also affect your physical and mental health negatively. The mental health effects of sleep deprivation are complicated because poor mental health can cause poor sleep as well as the other way round so a correlation does not tell you the direction of cause and effect, and the science is not quite as clear as it is on cognitive ability. However, it is definitely established that poor sleep can be a cause of poor mental health.

Final projects are always stressful in at least two ways: you are expected to do a large amount of high-quality work, and you know that it will have a significant impact on your degree classification. The abilities you need to demonstrate to get a good mark include the ability to deal with this stress and complete the work anyway.

You will have days when you feel less motivated or more stressed, and get less work done. This is part of normal life, and not a reason to ask for an extension. However, you need to do what you can to avoid making the project even more stressful than necessary – – such as being honest about your own abilities and your limits. If you can produce a project that gets a comfortable 2.i with a reasonable amount of stress, or push yourself above your limits to try for a First, then the correct choice to look after your health is to go for the 2.i.

If you notice your health (mental or physical) getting worse, or you are falling behind schedule, or you are worried that your thesis will not be complete on time, then you must speak to your project supervisor or Senior Tutor as soon as possible. If something has affected you before the Easter break, then we need to know before Easter; if something has been affecting you in previous years then you should mention it before you start your project.

[6] Announcement here, full paper here.

3.2 Ethics: You Probably Do Need To Think About It

All projects fall into one of three categories:

1. Projects that do not require ethical approval at all, because they do not collect or process any relevant data.
2. Projects that can be reviewed by your supervisor under "Ethics Application 0026", because they are judged to be low ethical risk.
3. Projects that require full ethical review by the Faculty Research Ethics Committee (FREC).

Before you can start your project, you will need to pass some brief ethics training to demonstrate that you've understood what is permitted under which set of rules. This is important to protect you from potential disciplinary or even legal consequences from e.g. accidentally violating the GDPR. It's also important to protect *us* from liability if someone breaks the rules — if that happens, we'll need to show we gave you appropriate training. This is the same dynamic underlying most training you'll get in the workplace.

Importantly, this training is mandatory for all projects. If you do not pass the training before starting work on your project, **you will not graduate**. However, we don't actually want this to happen to anyone — we just want everyone to do the training. As such, you can take the test at the end as many times as you like. You'll also need to input a code from passing the training when submitting your project specification, or the system will bounce it — this makes it impossible to miss the training by accident. More details are available on the unit page.

3.2.1 Projects Not Requiring Ethical Review

Your project is in this category only if it does not involve any gathering of data from human or animal participants. This means you **cannot** do any of the following:

- conduct a survey;
- ask anyone (including friends) to test and/or give feedback on your software;
- take any photos or video that contain people, **including yourself**;
- collect data from online sources such as Facebook or Reddit;
- process existing anonymous data in a way that could de-anonymise it.

Two examples of projects that would not require ethical review are as follows:

- Jean is doing a project in graph theory studying snarks. This will involve reviewing the literature and perhaps proving theoretical results or writing some code to generate snarks. The only experiments Jean will run involve testing and profiling her own code.
- Adam is doing a project in machine learning, and he is attempting to train a neural network to recognise different art styles and periods. In doing so, he will use a large data set of public domain images downloaded from the Internet, and perhaps one or two photos of his own pieces of art.

If you and your supervisor agree that your project does not require ethical review, then you don't need to do anything except add a sentence saying so in your front matter.

3.2.2 Projects covered by Ethics Application 0026

This category covers projects that involve gathering data only from humans (not from animals) and are judged to be low ethical risk. Ethical approval for projects in this category has been granted unit-wide by Ethics Application 0026, subject to oversight from your supervisor. You can use this option only if all of the following apply to your project:

1. The project does not specifically try to gather data from a vulnerable population, such as people affected by illness or economic disadvantage, primary or secondary school students (or any other under-age groups), victims of crime, disabled people, or people recruited from self-help groups.
2. The project does not gather data about protected characteristics including: racial or ethnic origin; religious or similar beliefs; membership of a trade union; physical or mental health; sexual orientation.
3. The project does not gather data about sensitive topics such as sexual activities; criminal history; drug use; or other obviously sensitive information.
4. All data is anonymous at collection, so that if the data were to be lost there would be no realistic prospect of the participants being identified. In particular, the project does not take photos or videos of people, and it does not ask about personally identifying information such as: names; addresses; postcodes; phone numbers; email addresses; physical features; or social media handles. This also forbids recording audio of participants in interviews. (As an alternative way of recording their answers to questions, we recommend using the automatic transcription feature of MS Teams.)
5. Before any data is gathered from a participant, they give full informed consent. That is, they understand what data will be collected and what the data will be used for, and they give an explicit verbal or written statement to this effect. If part of the data comes from observation, then participants are told what will be recorded before the start of the test. **In particular, projects which automatically scrape data from online sources such as Facebook or Reddit are not covered by this category and must make a full ethics application.**
6. No participants are tricked or deceived in any way, for example by being given false feedback about their performance at a task or being misled about the focus of the study.
7. If the participants are asked to perform a task, then this task involves no danger of physical or mental harm. For example, the project does not ask participants to sprint or to view distressing images. **This applies even to data gathered from yourself.**

Three examples of projects that would fall into this category are:

- Ayodya is writing a computer game for her project. As part of refining the core gameplay loop, she wants to get feedback from new players. She asks a few of her friends to play the game as she watches and take notes. Afterwards, she asks them a few questions about the game. She is careful to tell them in advance that she'll be observing them. Importantly, she writes down their responses to questions rather than recording audio, as an audio recording would be identifying information.
- Tan is writing a piece of software for research into grumkins for his project. Since Bristol researchers specialise mostly in snarks, he posts to a forum for grumkin researchers explaining his situation, asking what they would like out of a grumkin verifier and noting that any replies may be incorporated into his project. He then uses the requirements to decide what features to develop, and later releases the software as open source. When writing up his project, he removes all forum names from the feedback before including it.
- Mohammed is testing the effectiveness of a new user-interface element as part of an HCI project. To do so, he codes two versions of a user interface, one using the new element and one using a more traditional design. He divides participants into two groups, one for each version. He tells each group he is interested in testing the effectiveness of the user interface of the software and asks them to use it to carry out a task while he watches and takes notes.

This is generally the category for projects who want to create some software and do a small user study at the end. If you think your project falls into this category, then you should confirm this with your supervisor **before** you start gathering data, and you should not gather any data without your supervisor's approval. The two of you will then need to fill out a very short form available from the unit Blackboard page, and you will need to add a note to your front matter. Your project should also include an appendix with full detail on how you obtained consent from any participants, e.g. with the contents of any forum posts you made soliciting feedback. Sample participant information sheets and consent forms for in-person studies have been made available on the unit page.

3.2.3 Projects Requiring Ethical Review by the FREC

All other projects fall into this category. Six examples are:

- Chris is doing an HCI project to study effective survey design. They ask people to fill out one of two randomly chosen versions of an online survey about their favourite types of pizza. The two surveys are identical except that at the end, both versions ask for participants' email addresses in different ways; Chris is interested in which survey design gets more people to give their email addresses. The email addresses are discarded on collection to avoid gathering personal data. This would still require a full ethics application, since they are deceiving participants about the purpose of the study.
- Alan is doing a cybersecurity project on the information inadvertently leaked via Bluetooth from peoples' personal devices. He puts up a sign in the MVB atrium explaining that an experiment is happening, then sits there with a device gathering leaked data. He reads through it to remove anything potentially identifying, then includes it in his project. This would require a full ethics application, since it involves gathering data from participants who have not given full informed consent. The ethics committee would probably require significant revisions to experimental procedure.
- Reut suffers from clinical depression, and for her project she is writing a mobile app to help fellow sufferers track their moods. When she has finished coding the app, she posts a link to Big White Wall (an anonymous support forum) asking for feedback. This would require a full ethics application, since it involves asking about mental health and gathering data from a vulnerable population.
- Johannes is doing a computer vision project in which he attempts to train a computer to recognise numbers on signs. As part of his test data, he walks through the department and takes photos of all the office doors. This would require a full ethics application, since it involves gathering information (namely office numbers) from participants without their consent.
- Bill is doing a computational finance project on predicting stock market movement by analysing social media, and plans to use the contents of r/wallstreetbets as training data. This would require a full ethics application, since it involves gathering information from every poster on r/wallstreetbets without their consent.
- Sally is doing a computer vision project to evaluate users' form when exercising in the gym and offer suggestions for improvement. As training data, she uses videos of herself exercising. For the moment (as of September 2022) this still counts as recording identifying data from a participant, and still requires an ethics application. We do realise that this is silly, and changes to the faculty ethics rules are being discussed that would remove the need for it, but those changes aren't here yet.

If, after discussion with your supervisor, you decide that your project is in this category, don't panic!

The next step is to make a formal Stage 1 ethics application to the Faculty Research Ethics Committee at `https://orems.bristol.ac.uk/ActivityForm/Index`. The application process is not hard, involving only a short form, and typically if a project is "obviously harmless" but not covered by Application 0026 then it will be approved. That said, you must make your application well in advance, as delays of three weeks or more are relatively common, and you will not be granted an extension if this stops you from completing your project before the due date. If your application is approved, then you will need to add a sentence to your front matter quoting the application number.

If the FREC identifies significant causes for concern, then they may ask you to make a Stage 2 ethics application. This is a more difficult affair, and if it happens then you should think carefully about whether your planned experiments are crucial to the project.

3.3 Innovation Case (Fourth-Year MEng only)

As you approach the end of your degree it is important to consider how the advanced computer science skills and concepts that you have learned connect with the interests and needs of the outside world. Whether your future is as an entrepreneur, a researcher, a software developer, a data scientist, a roboticist, or in any career drawing on computer

science (or even in a career outside computer science), it will be crucial for you to be able to understand and express how your skills and knowledge can make a positive and valuable difference in the world. As part of your final year project, the Innovation Case assignment is an activity that is designed to help you achieve this connection in a way that is flexible and suits the nature of your project and your own interests.

As part of your project, you will develop a short Innovation Case document that lays out and argues for the relevance and significance of computer science work in your research project area for a specific external audience of your choice. This audience could be venture capitalists looking to invest in profitable new computer science start-up, or a national research funding agency looking to invest public money in supporting a novel, high-impact computer science research project, or an educational institution looking to improve and expand the teaching materials that they employ, or a governmental or non-governmental policy unit looking to improve regulations or legislation that pertains to an important area of computer science. Which audience you choose will be determined by a combination of (i) what topic your project is related to and (ii) what interests you most. The content and structure of your Innovation Case document will also be influenced by these considerations, but all Innovation Cases will have the same overall aim (to allow you to demonstrate that you understand how advanced computer science ideas can be relevant and impactful outside of a computer science degree) and will be assessed according to the same overarching marking criteria (see Section 4.4).

The Innovation Activity will be taught in the first six weeks of TB2 via a series of lectures and workshops. You will have a chance to develop and get feedback on your Innovation Case idea in parallel with the project activity that you are undertaking with your Supervisor(s). Your final Innovation Case will be submitted mid-way through TB2 allowing you to focus on the Project Activity for the remainder of the teaching block. Full details of the Innovation Case teaching and assessment will be provided via the unit web page.

3.4 Project Poster Day and Second Marker Meeting

Every year in Week 22, we put on a poster day of final-year projects open to anyone who wants to stop by and ask questions. We are currently planning to make this an in-person event with refreshments and industry participation; if this falls through, we'll do it online with Padlet and Teams like last year. Attendance is optional, but strongly encouraged — by this point the implementation of your project should be almost entirely done, so it will be a chance to show it off to the world! We may also be able to get attendance from industry.

At around the same time, you'll have a chance to meet your second marker and talk with them a little about how the project's going and what sort of thing they'd like to see in the dissertation. Very often this will overlap with the poster day, but not every second marker will be free that afternoon; they should get in contact with you to confirm. This discussion is not a viva, it doesn't carry any marks, and you don't have to do it if you don't want to. Most students do, though, since it's a good way to see where the second marker is coming from in advance of the actual viva in the summer assessment period.

More details will follow on the unit page in TB2.

3.5 Writing your Dissertation

3.5.1 Structure

The university sets out regulations for the structure and formatting of dissertations in Annex 14 of the regulations, available here. We strongly recommend that **you do not read this**. It's much easier to instead build your dissertation from the LaTeX template made available on the unit page, or failing that from the Word template. If you insist on not doing this, then your projects must include the following sections, in this order, separated by page breaks. The only exception is the acknowledgements section, which is optional.

- A single **title page**. The title of the dissertation should be at the top of the page. Your full name should be in the centre of the page. At the bottom centre should be the following words:

  > A dissertation submitted to the University of Bristol in accordance with the requirements of the degree of [Bachelor of Science/Master of Engineering] in the Faculty of Engineering.
  >
  > School of Computer Science, Electrical and Electronic Engineering, and Engineering Maths (SCEEM), [insert submission date here].

  Underneath, you should include the word count of the main body of the dissertation (i.e. not counting the frontmatter, references or appendices).
- A single-page **executive summary** (or **abstract**). This is a non-technical overview of the project in at most one side of A4 and at most 300 words. It should be single-spaced and in font size 10, 11 or 12.
- If your project is a resubmission, a **resubmission statement**. This is a one-side overview of all non-trivial changes made to the initial submission. Ideally this should also clearly highlight additional work completed, and responses to issues raised by the original feedback.
- Optionally, a single-page **dedication** or **acknowledgements** section thanking anyone you feel has particularly helped you (e.g. friends or family).
- The **author's declaration**, which should contain the following words:

  > I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Taught Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, this work is my own work. Work done in collaboration with, or with the assistance of others, is indicated as such. I have identified all material in this dissertation which is not my own work through appropriate referencing and acknowledgement. Where I have quoted or otherwise incorporated material which is the work of others, I have included the source in the references. Any views expressed in the dissertation, other than referenced material, are those of the author.
  >
  > SIGNED: ..... DATE: .....

  You must sign this, either by including a scanned image of your signature, or by using a digital signature feature as available e.g. in Adobe products[7].
- An **ethics statement**. In almost every project, this will be one of the following statements:
  - "This project did not require ethical review, as determined by my supervisor, [fill in name]"; or
  - "This project fits within the scope of ethics application 0026, as reviewed by my supervisor, [fill in name]"; or
  - "An ethics application for this project was reviewed and approved by the faculty research ethics committee as application [fill in number]".

  See Section 3.2 for more information. If something went wrong and none of those three statements apply, then you should instead explain what happened.
- A table of contents.
- The main body of the thesis, along with any appendices (see below).
- A **list of references** or **bibliography**. There is no need to use any particular citation style as long as you are consistent, although IEEE is common (and the LaTeX default).

The main body of the thesis is much more flexible and can vary to suit the topic. You should discuss how to structure it with your supervisor, but one very common structure is as follows:

- The **introduction** is a non-technical overview of your work; a hurried reader who reads only your introduction and conclusion should still get an idea of the big picture. The introduction is also the place where you can help your markers by claiming what your main achievements are; your mark will be weighted more heavily towards areas that you claim to be your main achievements and the introduction is a good place to do this.

[7] `https://acrobatusers.com/assets/uploads/public_downloads/2221/adobe-acrobat-xi-esign-pdf-file-tutorial-ue.pdf`

- The **background** can be variously described as: the state of the world before you started your project; the context into which your work fits in; anything the reader needs to know to understand your work that is not work you have done yourself for this project. The background may include a literature review, and it may be split into different subsections e.g. contextual and technical background.
- The **main body** of your work describes what you have done, how and why. This part will vary the most between different students' theses, e.g. a theory project will have a different body structure to an experimental one. Your supervisor is the person to ask about what makes sense for your project.
- The **evaluation and conclusion** is where you repeat what your main claims are, and then provide evidence for each one. This can include experiments, measurements, user studies or other forms of evidence. Make sure you distinguish between claims, facts/data and your own conclusions or interpretation — and make sure you do not forget to add your own interpretation for any results (particularly from experiments) that you state here.
- You may include one or more **Appendices**, but your thesis must be understandable to a reader who chooses not to read them — they are meant for things like extra figures and tables that you have already summarised in the main body of the thesis.

Not all of these sections are appropriate for all projects, and (with the exception of the compulsory sections discussed above) you should feel free to adapt or omit them as you see fit. For example, an expository research project intended to present a unified view of a suite of techniques would probably be structured more like a technical research paper, with an introduction followed by different aspects of the theory being developed in different sections.

3.5.2 Formatting

The following rules are mandatory, and are here to help markers be consistent when they are discussing and comparing different students' theses. As in the previous section, we strongly recommend that you use the LaTeX template provided on the unit page, in which case most of them will be applied automatically throughout the document.

- You must not use double-spacing. The maximum line spacing allowed is 1.5, but around 1.2 is recommended (with sensible adjustments depending on the font you are using).
- You must typeset your body text in a serif font (e.g. Times, Garamond, Baskerville; not Arial or Calibri).
- You must use at least a 10 point font size.
- Pages should be numbered consecutively at the bottom centre of the page, including appendices and taking the title page to be page 1.
- Pages must be DIN A4 sized and margins must be at least 2cm on all sides.
- Anything that you want to refer to (diagrams, figures, tables, algorithms, code listings) must be numbered relative to the chapter. For example, the fifth table in Chapter 3 would be labelled "Table 3.5".
- Figures, tables and diagrams must have a caption briefly describing what they show; if a figure, table, diagram or similar is included verbatim from another source then this must be indicated in the caption.
- Citations must be numeric. For example, "in [1], it is argued …" not "in (Smith 2015), it is argued …". You may include an author name, for example "Smith [1] argues …", but the name is not part of the citation itself. When you are referring to a section or figure/table/etc. in cited work, you may also include this in the citation, for example "in [Section 3.2, 1], it is argued …". This means Section 3.2 in the work listed as reference 1.
- You must include a complete bibliography including all cited work (and no other work that you don't cite in the text). Both LaTeX and Word include tools to e.g. keep the numbering updated automatically. Your references should be a consistent style; IEEE is common in computer science.
- When using direct quotes from other work, you must use quotation marks; you may use other formatting (such as italics) as well but this does not allow you to drop the quotation marks. An exception is figures, tables, images, diagrams etc. which you quote by making clear in the caption (usually directly below the item) that it is quoted from other work. In all cases where you quote from other work, you must also reference the source.
- You must submit your thesis as a single PDF file no more than 20MB in size. In particular, if you write your dissertation in Word, you must convert it to PDF before submitting. If your thesis is more than 20MB, then you have probably

included high-quality uncompressed images in the file; you should compress these.
- You will be able to submit other materials (e.g. code samples) separately, but be aware that your markers are not required to examine them.

3.5.3 Style and Length

Although many good technical writing guides exist, it is difficult to give generic advice because the challenges involved are personal to each student and specific to each topic. As such, most of what we can say consists of guidelines rather than rules – for more topic-specific advice, you should talk to your Supervisor(s) and any other proof-readers you can find. To provide some other points of reference, the unit web page houses a range of sample dissertations.

Your second marker is not necessarily an expert in your topic, although we will try to assign someone with an appropriate background. If they cannot understand your contribution, then they are likely to give you a poor mark no matter how good your thesis is! As a rule of thumb, if one of your peers taking the same degree programme can read and understand your dissertation, then your second marker can probably do the same.

In general, you should aim to be precise first and concise second. For example, "rain" and "precipitation" are not the same thing — the latter includes snow, and if you are describing data from a weather station then you need to pick the correct term of the two if it makes any difference to your project or the argument you are making. This is one point where technical writing such as a thesis differs from everyday English, where you might often talk about rain (or snow) but would rarely use the technical term "precipitation".

That said, being concise is a virtue and you should not sacrifice it without good reason. If you can make your thesis ten pages shorter without removing important content, then doing so will make your mark go up rather than down! There is a soft maximum of 50 pages for your thesis, not including front matter, bibliography and appendices. However, most projects will not need to write this much. The ideal length varies widely by project and by discipline, so you should check with your supervisor, but most supervisors think 30–45 pages is a good length for a solid 2.i (see Figure 1.4 in Section 1.4).

Leave any word or phrase out that is not doing any work. For example, you would only write "it is extremely important" if you had a reason that "it is important" would not be good enough. "The ability to make decisions with the assistance of data science is continually increasing in importance" could be replaced by "Data science is becoming more important to help us make decisions". If your project is about data science to help decision-making, then the second version also has the advantage that the important point (data science) comes first.

When I was writing the last paragraph, I first had "the second version has the additional advantage", then I changed it to "the second version also has the advantage". Look out for when you are using longer words than you need ("unnecessarily lengthy vocabulary") and change them for shorter ones if they say or mean the same thing. Technical writing is about being precise, not about showing off with long words. Long technical terms are absolutely fine — if you are doing a project on Convolutional Neural Networks then you will need to spell that out at least once — but keep the words that are not technical terms in your area short and your markers will be happy.

You should use the active voice and first person singular (I) except in a group project, where you should use plural (we). For example, "I performed an experiment" not "An experiment was performed" or "We performed an experiment". There are areas of scientific writing where passive voice is (or used to be) required; a Computer Science dissertation is not one of them. The one exception to using "I" instead of "we" is that "we" is valid English if it is intended to include both you and the reader. This is particularly common (although optional) in mathematical writing. As an example, "we now prove Lemma 47" followed by the proof is valid English, but "we spent two weeks proving Lemma 47" is not, since the reader didn't spend two weeks proving Lemma 47.

You should write formally, but not pompously. For example, write out contractions such as "I do not think that" rather than "I don't think that" (unless you are making a direct quote), but note that "John's book" for example is a possessive,

not a contraction, and does not need changing.

The absolute minimum standard of spelling and grammar is that which you would get by running your work through Microsoft Word's spelling and grammar checker and correcting any mistakes it finds (it will also tell you about some issues of style, but remember a thesis is technical writing).

There are times when you want to make a logical argument over several steps, for example: A proves P, and B proves P and not-Q implies not-R. C claims R, therefore if we accept C's claim then we can conclude Q. In this case it is fine to start a sentence with words such as "But", "Therefore", "Also" and so on (technically these words are called conjunctions), even if some style guides warn you against this in general; you need these words when making a logical argument to get the structure of the argument right. When you start a sentence with a conjunction, it is usual to put a comma immediately afterwards, for example "Therefore, we conclude Q".

Clear citations to relevant third-party work are a hallmark of good technical writing; they demonstrate your understanding of the topic, give credit where it is due, and allow the reader to find extra information where required. You will never diminish your own work by citing someone else's. You should, however, try to cite from either original sources or high-quality secondary sources. For example, if you are using a mathematical result, you should cite the original paper in which it is proved, or (if it is sufficiently well established) a textbook containing it. You should try very hard to avoid citing, for example, a relevant answer on Stack Exchange. You should **almost never** cite Wikipedia — not only does a well-written Wikipedia article link original sources which you should cite instead, but the article itself may be edited or deleted between you writing your thesis and your marker reading it.

3.5.4 Group Projects

In a group project, you will submit one dissertation for the entire group, comparable to a dissertation for an individual project. (Of course, you should expect that a dissertation for a three-person project will take a bit less than three times as long to write!) You will also submit individual reports which clarify and reflect on the contributions and achievements of individual group members. If you are taking the fourth-year MEng unit COMSM0052, then you will write individual innovation cases as normal; you must do so independently from each other. These projects are unusual, and if you're planning to do one then you should get in touch with the unit director early on.

3.6 Procedures

3.6.1 Changing Your Project from your Specification

It is normal for a project to change significantly as it progresses, and it is entirely possible that the title and description you gave as part of your specification in TB1 will no longer be accurate by the end of TB2. This is completely fine, and you don't need to ask for permission from anyone. However, if your project has changed so much from the specification that you think it will need a different second marker, then you should get in touch with the unit director ASAP; if you leave it too late, then changing the second marker may no longer be possible. For example, if your project has changed from automated recognition of giraffes from camera footage to automated recognition of penguins from postcards, then there is no need to contact the unit director. If your project has instead changed to developing a cooking simulator, then you should contact the unit director.

3.6.2 Changing Your Supervisor

As discussed in Section 2.2.2, submitting a project specification represents a commitment to working with that supervisor, and under normal circumstances we will hold you to that commitment. In TB1, it's not fair to other students if we allow

people to “hold an offer” from a supervisor while they go looking for other possibilities; in TB2, changing supervisor normally leaves too little time to do a good job with the project. As such, changing your supervisor needs the permission of both your personal tutor and the unit director. The one exception is if your supervisor has become unavailable for an extended period of time (e.g. due to illness), in which case you should contact the unit director and they will help you find a temporary or permanent replacement.

3.6.3 Changing Your Second Marker

You will find out who your second marker is in the middle of TB2, and under normal circumstances it is impossible to change them. Many students are worried about their second markers being harsh, but ultimately your final mark has to be agreed between them and your supervisor — they can’t unilaterally decide to mark you down. If the two markers absolutely can’t agree then the unit director can bring in a third marker to arbitrate, but this is very rare indeed. Despite our best efforts, some students also end up with a second marker without much background in your project’s area because there are no more appropriate choices available; in this case, in the event of a disagreement they will typically defer to your supervisor.

That said, if you have a strong reason to believe that your second marker cannot mark your dissertation objectively, then you should contact the unit director and they will decide whether a replacement is appropriate. A strong reason would include, for example, your second marker having turned you down as a supervisor in TB1 on the grounds that your project could not be made to work. It would not include, for example, having got a bad mark in one of their units in year 2. (In this case, you will probably remember them far better than they remember you!)

3.6.4 Extenuating Circumstances

In 2020–21, the university allowed every student a one-week extension on their final project. This was a special case as part of the Covid-19 pandemic mitigation, and does not apply this year. **You should assume that you will be handing in your project on time this year.**

You will have a meeting with your personal tutor shortly before the Easter break, and one of the purposes of this meeting is to check that you are on track to graduate. If anything has happened before Easter that could affect your ability to complete the project, you must tell us before the break. The reason for writing this paragraph is that in past years, students have sometimes told us in the week of the deadline about something that has been affecting them since the start of the year or longer, and by then it was too late to do anything that would help them to graduate on time.

The university has an extensions policy for Extenuating Circumstances, which is meant for **exceptional** and acute cases with a clear cause. Before the pandemic, the percentage of students asking for extensions was increasing every year and the reasons for which students were asking for extensions were not always in the spirit of the policy.

We are now taking a stricter line on extensions and we will not grant extensions except for circumstances that are exceptional, serious and could not reasonably have been foreseen or otherwise mitigated by the student. This way we hope to avoid the situation which arose in summer 2021, in which over 90% of students took an extension and for capacity reasons we had to defer graduation for all students with an extension longer than a week (however serious their case).

Examples of cases that would count as extenuating circumstances include:

- Victim of violent or sexual crime, e.g. rape, sexual assault, or stabbing.
- Death of a friend or family member, especially if unexpected (e.g. suicide).
- Serious accident or illness with a hospital stay (e.g. car crash, meningitis).

The following are all examples that are not extenuating circumstances:

- Poor time planning, including running out of time to write the thesis.
- Wanting more time to get a better mark.
- Minor illness, e.g. cold/flu, lasting up to a full week.
- Travel and technology problems.
- Normal levels of stress that are to be expected on a degree.
- Work taking longer than expected.
- Picking a project that turns out harder than expected.

The above examples in the second list are not a reason for an extension even if they lead to effects on your health. For example "the work is taking longer than expected and this is making me feel anxious and depressed" is not an extenuating circumstance; it is a reason for you to talk to your supervisor about reducing the scope of your project and deciding which features to drop, even if this may reduce your overall mark.

If you have a disability, or you have had extenuating circumstances in previous years, if you have struggled with the degree, or have other factors that may affect your ability to complete the project, then you must talk to a (senior) tutor and ideally also to your project supervisor before you start your project so that you can agree a plan that will let you graduate on time.

If you have a disability and have registered alternative exam arrangements (AEA) with the university, you should also ask for advice from the university's disability services. We will be more lenient in this case, but an AEA form does not mean an automatic extension on your project. Like all other students you will need to choose a project that is within your ability to complete, but you may be entitled to extra support such as a mentor organised by disability services.

3.6.5 Mitigations

If you do have an extenuating circumstance, the university may be able to offer you one of the following:

1. Let you defer your project to the autumn (usually early August). This will delay your graduation.
2. Let you suspend your studies and restart the project, or do a different project, in the following academic year.
3. Exceptionally, offer you an extension but still mark your project so you graduate on time.

Option 3 increases the workload for staff because they still have to mark your project and prepare your viva to the same deadline, but they now have less time to do it. Even before the pandemic, this was becoming unsustainable; during the pandemic, the combination of the free one-week extension and the extra workload for all staff but especially those with children made things even worse —- which is one of the reasons that we have reconsidered our approach to extensions.

There are also two issues of fairness that academic staff have become increasingly uncomfortable with over the years.

- On the staff side, imagine two students in a similar situation (maybe both have been unwell for a few days), but one of their supervisors is happy to mark their thesis one week late and the other supervisor is not because they already have other students with late work, childcare, a paper deadline or other commitments.
- On the student side, imagine that Alice and Bob both realise three weeks before the deadline that they are behind schedule. Alice follows the rules and drops a feature or two from her project plan, and writes up and hands in on time. Bob carries on with an extra feature, and asks for an extension because he is now short of time to write up the thesis. It would be unfair to give Bob extra time as this gives him an advantage over Alice — as a reward for breaking the rules, he can get one more feature done. This is true even if Bob's extension request comes with an explanation how being behind schedule is affecting his mental health and even if this reason is entirely genuine: firstly because he could have avoided the problem like Alice by better planning such as dropping a feature earlier on; secondly because arguably Alice has demonstrated the better project planning skills as she handed in on time.

This leads to the philosophical but very practical question, who do you give extensions to when you do not have the staff

capacity to give one to everyone who asks? Our answer is that we will only offer an extension that lets you graduate on time, if we could offer the same extension to every other student with an equally good reason and still have a reasonable time to mark all projects. This means that extensions will only be for cases that are both acute and exceptionally severe, such as the examples listed earlier.

However severe your case, it is the university's decision whether they offer you an extension or a deferral –– if you are not able to hand in by the original deadline then you must accept the possibility that this will delay your graduation. There is no automatic right to an extension.

Extensions and deferrals are designed mostly for acute problems, not chronic ones. If you have been generally poorly for most of the teaching block and never felt quite as productive as you wanted to be, your options are: you can adjust the scope of your project and hand in what you have by the original deadline, or you can suspend studies if you think that repeating the project in the following year would improve your situation. "I was not at full capacity so I need extra time to make up for this" is not a valid reason for an extension or a deferral –– either you are well enough to do a project, or you should suspend studies.

3.6.6 Intellectual property (IP) Rights

Since work on your project is work produced as part of your degree, the intellectual property rights are governed by Section 5 of the Student Agreement (here) and by the accompanying IP policy document (here). Here's the unit director's interpretation of these documents, subject to the important caveat that they are not a lawyer and this is not legal advice. For almost all projects, the situation is as follows.

- Anything created by university staff (e.g. pre-existing experimental data that you might be analysing) is the university's IP, and you should defer to your supervisor on what you can do with it. In particular, you must **not** make it public (e.g. by pushing it to a public GitHub repository) unless they have given permission.
- Anything you create during your project is your own IP, and you can do what you like with it.
- Despite this, the university has rights to **non-commercial** use of it. For example, we have the right to put an anonymised copy of your thesis on the unit website, or to write a paper using some of the ideas it contains (crediting you for those ideas).

There are a couple of rare exceptions to this:

- A few projects benefit from outside funding (e.g. from the UK government or from the EU), and this funding normally comes with restrictions on the IP rights, e.g. requirements that any research be made freely available to the public. Normally, this will only apply to projects that fit into an existing research grant — check with your supervisor if you're unsure.
- A few projects involve working with someone outside the university who has a claim on your IP, e.g. a supervisor from industry. In that case, you should point them to the original documents linked above (not the summary) and make sure their lawyers are happy with them. Normally this doesn't present issues, and collaborations with industry are relatively common.
- For a very few projects, your supervisor might require that you sign over IP rights to them or to the university. This is very rare, and would normally only happen if e.g. you're working on part of a much larger project of theirs that they're hoping to commercialise. If it does happen then your supervisor must make this clear to you and give you a form of assignment to sign **before** you start the project. You're also entitled to a share of any profits under the terms of the Revenue Sharing Scheme (here).

The unit director's personal stance is that having an interesting project go ahead is far more important than what happens to the IP afterwards. They don't have the authority to sign away the university's rights, but they do have authority over unit-level decisions, and will generally be happy to agree to ideas like not making the dissertation available to next year's students if this makes things easier. In particular, under normal circumstances you would need to submit copies of

auxiliary materials like source code and raw data alongside your project, but we're happy to waive this if there are IP concerns. (However, you will still need to show portions of your code and raw data in the viva, to demonstrate that you've done what your thesis says you have.)

4 Vivas and Marking

The BSc unit 'Individual Project' is worth 40 credit points and you will get a single mark for your thesis. The third-year MEng unit 'Individual Short Project', available only to Maths and Computer Science students, is worth 20 credit points and you will get a single mark for your thesis. The final-year MEng unit 'Individual Project with Innovation Case' is also worth 40 credit points, but your unit mark will be calculated as 88% thesis mark, 12% Innovation Case mark; so the project is worth the equivalent of 35 CP and the Innovation Case is worth 5 CP. (We wanted to make the split 87.5%/12.5%, but there were technical issues.) Most of what is written below applies to the thesis mark only; there is a separate marking scale for the Innovation Case later in this document.

4.1 The Marking Process

Once you have handed in your thesis, the following marking process takes place. This is managed by a piece of software called MarkerBot written and maintained by the project unit director.

1. Your first marker, who is also (one of) your supervisor(s), reads your thesis and comes up with a mark suggestion and some text justifying this mark. The mark and text are stored in a database. This happens without talking to the second marker.
2. Your second marker reads your thesis and holds a viva with you, typically online and recorded. After this, your second marker also enters a mark suggestion and some text into the database, without talking to the first marker.
3. Once both markers have submitted their mark and text, the system releases both marks to both markers. The markers then discuss and agree on a final mark and feedback. Often the two marks are either equal or very close, and in this case the markers will typically handle it by email. If there is a larger gap, or a gap which covers a grade boundary, then they will typically schedule a longer meeting in-person or over Teams.
4. Your first marker then enters your final mark and feedback (which may involve merging the two justifications) into the database, along with some metadata and a short explanation of how the mark was agreed on.
5. The system emails the second marker for final confirmation and signoff.
6. After the final signoff, the unit director moves the marks and feedback from the database into Blackboard at the end of the marking period, and you receive them as normal.
7. Your project also has a designated backup marker. If the markers cannot agree, or if they would like a third, independent opinion, or if one of them becomes unexpectedly unavailable (e.g. due to illness), then this backup marker may read your thesis and contribute to the discussion. This is very rarely necessary.

Both markers are equals in the discussions that lead to your final mark. Both of them will have read your thesis, but each one will have different information beyond that: your supervisor will know how you worked during the project, for example how independently you worked and what problems you hit along the way; your second marker will know how you performed in the viva. Both markers will consider all of the marking criteria, which are listed later on in this document.

A simple case of a marking discussion might be that one marker suggests a mark of 62 and the other suggests 64; in this case they might both agree to average it and submit 63. More complicated cases would be if the marks from first and second marker are further apart, or on different sides of a grade boundary such as 68 and 71. This kind of case would not be dealt with by a simple average: a decision on which side of a grade boundary to put a borderline mark is always a conscious decision that both markers must agree on. The markers are expected to report back to the unit director how they decided which side of the boundary to agree on.

As a student, you only get to see the final thesis mark and feedback, not the initial marks and justifications that your supervisor and second marker entered before starting the discussion.

There are two levels of quality control after marks have been returned by all markers. First, the project unit director will

look at the whole cohort marks again and check with individual markers if they spot any anomalies. Secondly, an external examiner (from a different university, but also a Computer Scientist) will check all marks again and they may read part or all of a sample of theses to make sure that our marking is consistent and that our degree outcomes match the ones being awarded in other comparable universities, such as other Russell Group universities in the UK. They will also check that we have procedures in place for recording how we decided on marks, especially in borderline cases (which is why markers report this back to the unit director).

4.2 Vivas

You will hold a viva with your second marker (that is, the marker who is not your supervisor) in the summer assessment period after you have submitted your thesis. Under Section 11 of the university regulations, these vivas must be run either in-person with two members of staff present or online and recorded. Typically it is impractical to run them in-person due to scheduling conflicts between markers, and so we run them online via MS Teams — this way the recording happens automatically, and can easily be shared with both markers and the external examiner. If your second marker runs your viva differently to this, you should get in contact with the unit director ASAP.

The purposes of the viva include:

- To get assurance that those parts of your work which can't be incorporated into your thesis function as described. For example, you might be asked to show raw data from any surveys you have carried out, or to demonstrate any software you have developed.
- To get assurance that you have done your project yourself. The aim here is not to trip you up with trick questions — if you have written the thesis and done the work yourself, these questions should be no problem to answer. For example, you might be asked to show your code for a given feature and explain briefly how it works.
- To discuss possible errors or omissions in the thesis — if you can explain something in the viva that is not clear from the thesis, that counts in your favour.
- To correct possible errors in your understanding of the material.
- Especially for top students, to sample the depth of your understanding of the material and to see how independently you've worked from your supervisor, for example by asking questions on the topic that go beyond what you have covered in the thesis.

The details of the viva are up to your second marker and will depend on what makes sense for your particular project, but a common format is the following:

1. You give an overview of your work in roughly 5 minutes (with no need to e.g. prepare slides in advance).
2. If appropriate, you spend roughly another 5 minutes demonstrating results (for example, running your software and sharing your screen).
3. For the rest of the time (roughly 10–15 minutes), you answer questions from your second marker.

Vivas are fairly short, and this is by design: a viva is more about high-level understanding and less about technical detail. These details belong in the thesis.

It is your responsibility to prepare for the viva by, for example, making sure the software runs correctly on the machine you will be using to demonstrate, that Microsoft Teams is installed, and that you have access to a microphone, camera and good internet connection. If you do not have access to a camera, this may be counted as an unauthorised absence — we need to be sure that the person giving the viva is really you.

The viva is not a separate component of the unit mark. Instead, your thesis mark is the result of a discussion between your two markers as explained earlier. The viva will be recorded and shared with your first marker, so both markers can take your performance into account. In terms of "how much should I worry?", though, it is good to treat the viva as though it were worth 5–10% of your mark: worth preparing for diligently, but not worth any sleepless nights. The second

marker will be aiming to see what you can do rather than what you can't do — for example, you don't have to worry about "losing marks" by making a silly mistake — and ultimately you're still the same person taking the viva as you were when you carried out the project. For most students, the viva clarifies and reinforces the second marker's initial impressions from reading the dissertation rather than changing it completely.

4.3 Project Assessment Criteria

How your project is assessed will depend on many specific factors, including the type of degree it is for (BSc, MEng), the type of project (enterprise, research, theory, development etc.), and what achievements you claim in your thesis. For example if you tell us in your thesis that the usability of the software you developed is your most important achievement, your markers will most likely weight usability higher than other criteria.

That said, here are some generic criteria against which we assess all projects. The four big themes are

- Challenge and Achievement: Results and Contribution
- Technical Approach: Design and Implementation
- Critical Interpretation: Evaluation and Understanding
- Report Presentation: Organisation and Communication

In each of these four themes, your markers may choose to give you one of the following descriptors:

- Outstanding (80–100 mark range)
- Excellent (70–79)
- Good (60–69)
- Satisfactory (50–59)
- Weak (40–49)
- Poor (0–39)

The descriptors reflect the level at which your markers judge that you have achieved most or all of the relevant criteria, as explained in the following sections. Your thesis mark is not a weighted average of the marks for each theme, and you will not get a mark for each one. Rather, the themes are inputs for you as a student to decide what to focus on in your work, and for your markers to use in the discussion where they agree a single mark for your work.

4.3.1 Challenge and Achievement

This theme looks at what results you have produced and what contribution these results make. The relevant aspects for this theme are

- The **ambition** of your project goals and the degree of challenge that it posed.
- The **extent** to which you achieved your project aims.
- The **utility** of the work that you produced.
- The **degree** to which the work was self-directed or relied on your supervisor's direction.
- The **distance** to which the work goes beyond the ideas and techniques taught in the curriculum.

For an **outstanding** rating, your project would achieve most of the following:

- An ambitious, challenging project that achieves all of its aims.
- Could form the basis of an academic research paper or a commercial product.
- Consistent with the quality of a good early-career PhD student or an employee at a high-ranking institution.
- Features a substantial volume of self-directed work.

- Goes significantly beyond the scope of the degree programme.
- Could be used as a clear example of a model project.

For an **excellent** rating, your project would achieve most of the following:

- An ambitious, challenging project that achieves all of its aims.
- Could be further developed to form the basis of an academic research paper or a commercial product.
- Features a significant volume of largely self-directed work with only limited guidance from the supervisor.
- Goes beyond the scope of the degree programme.
- Could be used as a clear example of a very successful project.

For a **good** rating, your project would achieve most of the following:

- A reasonably challenging project that achieved most of its aims.
- A reasonable volume of occasionally self-directed work but with significant supervisor input.
- Remains mostly within the scope of the degree programme.
- Could be used as an example of a solid project.

For a **satisfactory** rating, your project would achieve most of the following:

- A somewhat challenging project that made substantial progress but failed to achieve some of its aims.
- An adequate volume of work, but mostly supervisor-directed.
- The scope remains mostly or wholly within the degree programme.

For a **weak** rating, your project may meet most of the following:

- Little challenge.
- Made some progress but failed to achieve several significant aims.
- An inadequate volume of work, mostly or wholly supervisor-directed.
- Remaining entirely within scope of the degree programme, or aligning poorly with Computer Science.

A **poor** project may meet most of the following:

- Trivial challenge, little progress, achieving few if any aims.
- Inadequate volume of work and little or no self-direction.
- Scope entirely within the degree programme, incoherent, or aligning poorly with Computer Science.

4.3.2 Technical Approach

This theme looks at the design and implementation of your project. The main aspects are:

- Your **grasp** of the relevant technical material, theory and methodology.
- The **quality** of your design and implementation.
- Your **command** of the tools, techniques and methods that you demonstrate.
- The **degree** to which your work was systematic, considered, and well explained.
- The **value** of your project outputs, compared to the state of the art.

For an **outstanding** rating, your project would achieve most of the following:

- Your project's findings represent an advance on the state of the art.
- Your work could form the basis of an academic publication in a reputable venue or a new (commercial) product.
- The methodology, tools and techniques are appropriate and are motivated and employed correctly throughout the project.
- Approaches are motivated well and alternatives are considered where appropriate.
- Very strong command of the relevant tools and techniques, and clear evidence of this.
- Technical material is handled in a clear and convincing fashion throughout the thesis.

For an **excellent** rating, your project would achieve most of the following:

- The project findings are largely novel.
- Results, analysis or designs are sound and useful.
- Methodology, tools and techniques are appropriate with some motivation.
- Some consideration of alternatives, where appropriate.
- Solid command of relevant tools and techniques, with evidence.
- Technical material is handled in a clear and convincing fashion; where there are technical problems, fixing them would be only a minor effort.

For a **good** rating, your project would achieve most of the following:

- The project findings are somewhat novel.
- Results, analysis or designs are sound and to some extent useful.
- Methodology, tools and techniques are employed appropriately for the most part, even if they could have been used more effectively in places.
- Some motivation and consideration of alternative approaches.
- There is some evidence of appropriate use of tools and techniques.
- Technical material is mostly handled in a clear and convincing fashion; where technical problems exist they could be fixed with moderate effort and would not require a re-design.

For a **satisfactory** rating, your project would achieve most of the following:

- The project findings have some value, even if they mainly replicate existing work.
- Tools and techniques are reasonable but could have been used more effectively in places.
- Little consideration for motivating the choice of tools or alternative approaches.
- Methodology is somewhat ad-hoc or unsystematic in places.
- Some lack of command of the tools and techniques employed in places.
- Technical material is sometimes handled in an incorrect or unconvincing fashion, however resolving the problems would require moderate effort and little redesign.

For a **weak** rating, your project may meet most of the following:

- The project findings are of little value, for example they fail to replicate existing work.
- Tools and techniques are mis-applied or not always appropriate.
- Little consideration if any of the choice of tools or alternatives.
- Methodology is ad-hoc or unsystematic in places or more generally.
- Some lack of the command of tools and techniques employed, which may involve examples of clear mistakes in their use.
- Technical material is handled unconvincingly or incorrectly, and resolving the problems may require significant effort or substantial redesign.

A **poor** project may meet most of the following:

- The project findings have no value, or almost no value.
- Tools and techniques are mis-applied, or not appropriate.
- Little or no consideration of the choice of tools, or of alternatives.
- Methodology is unsystematic or absent throughout.
- Evidence of a profound lack of command of the tools and techniques employed.
- Technical material is handled incorrectly or unconvincingly throughout, and fixing the problems would require restarting the project.

4.3.3 Critical Interpretation

This theme is about your **evaluation** of your work, and the **understanding** you demonstrate in doing so. The main aspects here are:

- The quality and extend of critical or comparative **analysis and evaluation**.
- The use of appropriate and valid **metrics and techniques**.
- An appropriate balance of **qualitative, quantitative and analytic** considerations.
- The level of **understanding and interpretation** of the project findings.
- A honest appreciation of the **limits, flaws and drawbacks** of the work.
- Grasping the **relevance and implications** of the work.

For an **outstanding** rating, your project would achieve most of the following:

- Extremely well-designed and well-executed comprehensive evaluation.
- Metrics and techniques are appropriate and generate interesting conclusions.
- A strong reasoning behind the conclusions, supported by effective and comprehensive analysis.
- A sophisticated and deep critical appraisal that aligns very will with both the project aims and relevant research literature.
- An appreciation of the relevance of the results for future work.

For an **excellent** rating, your project would achieve most of the following:

- A suitably well designed and executed evaluation.
- Metrics and techniques are appropriate and generate sensible conclusions.
- Conclusions are supported by argument and evidence.
- Sophisticated critical appraisal that aligns well with the project aims.
- Identifies relevant future work and open problems.

For a **good** rating, your project would achieve most of the following:

- Some evaluation, that is mostly or wholly suitably designed and executed.
- Metrics and techniques are appropriate and conclusions generated are consistent.
- The evaluation presents a critical appraisal and is reasonably well aligned with the project aims, even if it is shallow or sparse in places or has some problems in its execution.
- The evaluation contains some consideration of future work or relevant open problems.

For a **satisfactory** rating, your project would achieve most of the following:

- Some evaluation, but it may be poorly designed or executed.
- Metrics and techniques may not be appropriate, or generate only limited conclusions.
- The critical appraisal may be shallow or not well aligned with the project aims.
- Little or no useful consideration of future work or relevant open problems.

For a **weak** rating, your project may meet most of the following:

- Very little evaluation, poorly designed or executed.
- Metrics and techniques may not be appropriate, with no or very limited useful conclusions.
- The appraisal may be shallow or only partially applicable to the project aims.
- Little or no useful consideration of future work or relevant open problems.

A **poor** project may meet most of the following:

- Evaluation is superficial, incoherent or completely absent.
- Little or no useful conclusions are reached.
- Essentially no critical appraisal.

4.3.4 Writing and Presentation

This theme is about how your thesis is **organised** and how it **communicates** to the reader. The main aspects here are:

- The quality of the **writing, grammar and spelling**.
- The clarity of the thesis **structure and formatting**.
- The use of **diagrams, figures and tables**.
- The extent and relevance of the **literature review**.
- Citing appropriate **prior work**.
- Your performance in the **viva**.

For an **outstanding** rating, your project would achieve most of the following:

- The thesis is clear, accurate and engaging.
- The quality of the presentation is extremely high and makes the thesis a pleasure to read.
- Very few, if any, problems with spelling and grammar.
- Visualisation is creative and effective and was developed by the student themselves. Visual consistency is kept throughout the work.
- The literature review is strong, relevant and extensive.
- Appropriate prior work is properly cited and treated in a way that demonstrates a strong understanding of the wider context of the work.
- The student can answer questions on their topic without significant prompts.

For an **excellent** rating, your project would achieve most of the following:

- The thesis is mostly clear, accurate and engaging.
- The quality of presentation is high with few mistakes of spelling and grammar.
- Visualisation and illustration is effective throughout the thesis.
- The literature review is strong and cites relevant work.
- Appropriate prior work is properly cited in a way that demonstrates a good understanding of the wider context of the work.
- The student is usually able to answer questions on their topic without significant prompts, and is able to extend

their answers to a more advanced level with some guidance.

For a **good** rating, your project would achieve most of the following:

- The report is easy to understand.
- The presentation quality is good, with only minor problems of spelling and grammar.
- Visualisation and illustration of reasonable quality and mostly used in an effective way.
- The literature review is solid and cites relevant work, demonstrating an understanding of the wider context.
- The student is usually able to answer basic questions on their topic without significant prompts, but not to extend them to a more advanced level without help.

For a **satisfactory** rating, your project would achieve most of the following:

- The report is understandable but may be unclear in places.
- The quality of presentation is satisfactory but may have substantive problems in places, for example unclear definitions.
- Visualisation and illustration may not be used effectively.
- The literature review is superficial or otherwise flawed, for example clearly relevant and important work is missing.
- There is no demonstration of a solid understanding of the wider context of the work.
- The student may be unable to answer basic questions on their topic without at least some prompts.

For a **weak** rating, your project may meet most of the following:

- The report is hard to read.
- Presentation is weak, there may be substantial problems that hinder the reader from understanding key concepts.
- Visualisation is absent, ineffective or visually inconsistent.
- The literature review is either absent or inadequate in scope and depth, misses important citations of direct relevance to the topic, and demonstrates a limited understanding if any of the wider context of the work.
- The student is generally unable to answer even basic questions on their topic without significant prompts, or at all.

A **poor** project may meet most of the following:

- The report is frequently very difficult to understand.
- Quality of presentation is poor to the extent that it prevents the reader from understanding the points that the thesis is trying to make.
- Visualisation is either absent completely or confusing to the point that it does more harm than good.
- Literature review is completely absent or inadequate in both scope and depth, has few or no correct citations, may attribute key ideas or techniques to the wrong sources, and shows a very limited understanding or even a clear misunderstanding of the wider context of the work.
- Even with significant prompts, the student is unable to answer the most basic questions on their topic, possibly to the point that it is unclear if the student has done any work on the topic at all.

4.4 Innovation Case Assessment Criteria

If you are on the MEng degree, you will submit an Innovation Case during term that is worth 5 CP (or 12%) of the 40 CP project unit. The thesis and innovation case are marked by different people and they do not discuss among each other, so your thesis mark does not affect your Innovation Case mark and vice versa.

Your innovation case will be marked on four themes. In each theme, you will be given a rating that contributes to your overall mark. The ratings are: outstanding (80–100), excellent (70–79), good (60–69), satisfactory (50–59), weak (40–49), poor (0–39). The numbers indicate the innovation case mark range you would get if your rating for all four themes falls in this category, for example an innovation case rated 'good' on all themes would get a mark in the 60–69 range.

To get a particular rating for a particular theme, you would normally have to achieve most but not necessarily all of the criteria listed below at the given level. For example, if you are judged good on 2 of three points for one theme and satisfactory on the third, your rating for this theme would normally be 'good'.

The interpretation of each criterion below will depend on the nature of your project and the case you make, so the criteria and ratings are deliberately written in a very general tone.

The themes for the innovation case are:

Research and Methodology: the quality of the research process demonstrated in your innovation case, particularly

- The extent and relevance of primary and secondary sources used.
- The quality and range of your evidence and argument.
- The evidence of careful reflection and thought in your submission.

Planning: the quality of the planned activity in your case, in particular

- The relevance and potential value of the planned activities for the target audience.
- The clarity and substance of the planned activities.
- The creativity of the planned activities.

Analysis and Understanding: the level of analysis and insight demonstrated, in particular

- The precision and appropriateness of the defined target audience.
- The salience of the identified need or value.
- The depth of understanding evidenced.

Presentation and Communication:

- The quality of your writing, grammar and spelling.
- The clarity provided by the structure and formatting.
- The efficacy of compelling and persuasive language, which can include diagrams, figures, tables etc.

5 Academic Integrity and Plagiarism

We expect you, as a final-year student, to be familiar with and to correctly follow the university rules on academic integrity and plagiarism. Plagiarism is an offence and can lead to a punishment even if you were unfamiliar with the rules or made an honest mistake.

There are at least three ways that we might discover if some of the work you submit is not your own:

1. At least one of your markers will normally be an expert in your subject area. If they notice text or other material in your work that they have already seen elsewhere, and that is not correctly referenced, you are likely to be investigated for plagiarism. For example, if you are doing a project on Neural Networks and you include the image from the Wikipedia page on Neural Networks without attribution, it is likely that one of your markers will recognise this.
2. Your work will be run through a piece of software called Turnitin. More on this below.
3. If it appears in your viva that you cannot explain even basic parts of your thesis, then you will most likely be investigated further.

A lot of plagiarism happens not because a student is trying to be evil, but because they are behind schedule and in a panic, so they copy-paste something and hope no-one will notice. If you are ever in a panic about not being able to finish on time, please get help from your project supervisor, tutor or another university service — based on our experience as academics, we do catch and penalise a lot of plagiarism that is done in a hurry.

For further reading on plagiarism, apart from the rest of this section you may want to consult the official university guidance here, Section 3 of the university regulations here, or the Faculty of Arts guide to proper referencing here; note, though, that in computer science we are far more relaxed about the exact style of reference used. (IEEE is default, but not required — see Section 3.5.2.)

5.1 Turnitin

Turnitin is a piece of software containing a database of published research and student work that is used by many universities. When your work is submitted to Turnitin, it gets added to the database, so if a student next year tries to copy from your thesis then this will get flagged up. The same will happen if you copy from a student or a research paper, even from another university. Turnitin also searches Internet resources (think of this like pasting paragraphs of your work into Google to see if there is an exact match).

Due to how Blackboard and Turnitin interact, you will only be able to submit your final version of your thesis once, and you won't have access to the "Score/Report" that Turnitin generates. This is because on a re-submission Turnitin deletes all submissions everywhere—even after the deadline has passed. If we allowed re-submissions then we can no longer reference which was the original piece of work submitted on time, when it was submitted or which is the correct version to mark.

Behind the scenes, Turnitin produces two pieces of output: a similarity score (a percentage from 0 to 100, higher means more similar to previous work) and a report where text and other materials that match other work are highlighted.

Both are only guidelines for us to decide whether we believe Academic Integrity has been broken. For example, if you include a paragraph of text from another paper that you correctly identify as a quote by putting it in quotation marks and adding a reference, then this is not plagiarism; if you copy the same paragraph without putting quote marks or a reference, then it is plagiarism. Both of these will produce almost exactly the same Turnitin score however, which is why there is no acceptable range that we can give you for this score: a thesis with a large literature review component and 50% similarity is absolutely fine if everything is correctly referenced and quoted, and a thesis with 5% similarity can get

you accused of plagiarism if that 5% is half a page directly copied from another source without referencing.

You will never be subject to a plagiarism investigation on the basis of a Turnitin similarity score alone. If you are investigated due to Turnitin, this is because a member of unit staff has read your whole report, checked whether the Turnitin-marked areas are correctly quoted and referenced or not, consulted your markers, and then made an academic decision to make a formal accusation.

5.2 Correct Referencing: Quoting and Paraphrasing

The basic rules are:

1. Plagiarism is using other people's work and claiming it as your own.
2. Your thesis must start with a declaration that all the work it contains is your own unless indicated otherwise.
3. It follows that, wherever you use work that is not your own in your thesis, you must indicate this, for example with referencing and/or quoting.

Some work is genuinely your own. This is good, and you do not have to reference anything.

Most work you do will be based on existing work, such as researching an idea from the literature and summarising it in your own words, or implementing something based on a description in a textbook or online. This is absolutely fine. You are actively to expected to make use of the work of others rather than reinventing the wheel, and doing so will give you more credit rather than less — as long as you reference properly. This is especially true for parts of your thesis where you are discussing background or reviewing the literature.

Occasionally you will want to use some source material directly, without rewriting it in your own words. This is fine too, but you must indicate that the material is a direct quote as well as referencing it. A normal citation gives the original author credit for the idea, but in this case you are also using their exact phrasing, and you must credit them for this as well. To do so:

- Either introduce the material with a reference, e.g. **Smith says:**, or put a reference e.g. **(Smith, 2015)** directly after the material that you've quoted.
- Put quote marks around the material that you've quoted. This is mandatory.
- You can also set the material in a different font or style (e.g. italic, indented).

The one exception to the above requirements is if you are citing a published statement of a theorem, lemma, or mathematical proposition. In this case, the original idea and the original phrasing are bound together so tightly that the normal assumption is that you are using both, and so a normal citation before the result statement is fine.

If you use or refer to others' work, but paraphrase it (also known as rewriting it in your own words) then you still need to reference, but you do not put quote marks. Paraphrasing does not mean copy-pasting and then changing a few words! Good paraphrasing is much more about restructuring than rewriting: it means taking one or more sources, parsing the structure of the arguments they are making (drawing a diagram might help you here), making your own decision which parts are important for your own work, reordering the points made if it suits your work better, maybe adding examples of your own, and then 'compiling' that into your own version.

This takes effort, and a good paraphrase requires you to understand the points that the original authors are making and even to apply critical thinking — for example, to see which of these points are irrelevant to what you're trying to say. In programming terms, good paraphrasing is less like changing variable names, and more like porting code from one language to another.

Especially in technical material where phrasing is restricted, the line between good paraphrasing, bad paraphrasing, and

plagiarism can become blurry. Here is a useful technique for paraphrasing that will always keep you on the right side of it. First, read the sources you are trying to include. Then, *without looking at those sources*, write down what you would like to say. If you do this, you will naturally end up phrasing things in your own voice.

In summary, there are three levels of referencing:

1. Genuinely your own work: no references required.
2. Others' work, paraphrased or summarised in your own words: you must reference, but not put quote marks.
3. Direct quotes from others' work: you must reference with quote marks.

If you have dyslexia, then paraphrasing might be particularly hard for you. In this case you should contact Disability Services to get the specific help that you need. You are expected to meet the same standards of academic integrity as everyone else, but you may be offered extra specialist help to meet them.

If you are not a native speaker of English, you might also find paraphrasing harder, but remember that it is about understanding the structure of an argument (which requires domain-specific technical knowledge) much more than about English language skills or replacing words with synonyms. If you don't feel confident in paraphrasing something, then you can always quote it, as long as you add the quote marks and reference.