MSc Project Briefing
The Report

Prof George Magoulas
g.magoulas@bbk.ac.uk

Part 3

Outline

- Working on your project
- Report key sections and content
- Ethics, risks and mitigation strategies
- Using and referring to existing material/sources and assessment offences
- Marking criteria
- Submission, late submission, deferral
- What to avoid

Working on your project

Start writing up as soon as possible.

- Start with the structure of the report.
- Proceed with the structure of each chapter, section etc.
- At the paragraph level, write what you plan to write in the paragraph.
- Collect your references, adopt a referencing style.

General rules for writing report

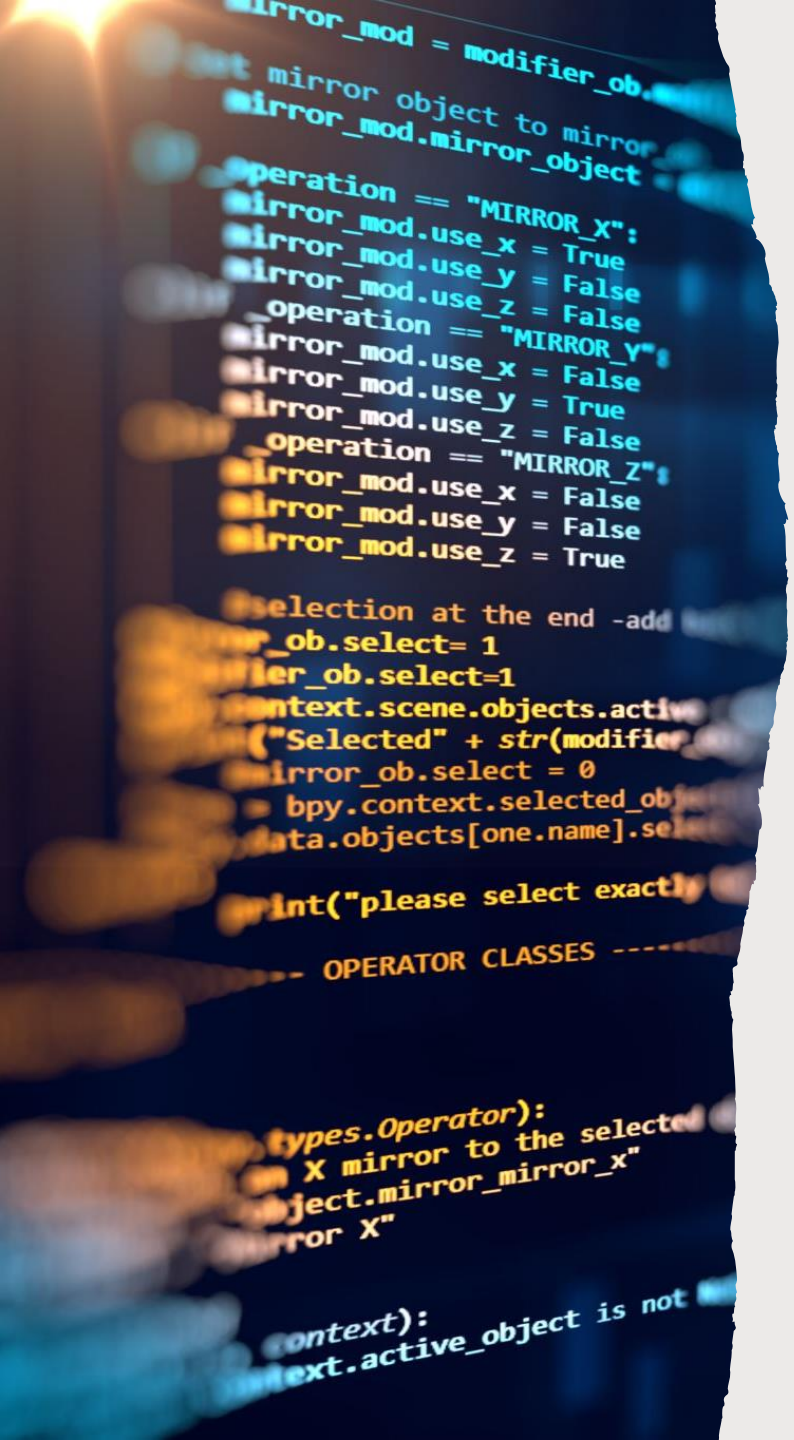
- Plan
  - Have a list of tasks describing what you plan to write
  - Make sure that you understand the meaning
  - Make sure that you can explain why you put everything in its current location
- Avoid using buzzwords
  - Avoid generic text that sounds good but does not add extra information (this gives a bad impression, far worse than being slightly less than the word count)
- Cohesion
  - Try to have a well organised work, easy to follow and logically coherent

General rules for writing report

- A good report is well proportioned:
  - None of the parts is too long
- All figures must be properly explained
- Do not have code examples unless they are very special
  - Function/procedure calls and some other basic code are not special
  - If you do have code examples:
    - You must explain them in a context that someone that is not familiar with your code can understand.

Advice for report writing

- Write the report, not for your supervisor, but as if you are addressing the Marker/External Examiner:
  - someone you do not know and who knows nothing about your project.
- Do not assume that the reader already knows about your project.
- Write in a detached style and avoid excessive use of personal pronouns such as I, you, me, my, etc.



Working on your project

Start Coding as soon as possible.

- It is suggested to use an incremental approach:
  - Start with the minimal implementation that uses all the required technologies. For example,
    - A GUI with one button and one text box. When the button is pressed something happens.
    - Proceed by involving a database
  - Once you have all the technologies working together, start adding features and refactoring.

Coding

- Advice:
  - If you start to work on the report, you can choose which features to add according to what makes your report look better.
  - It is better to implement some features two weeks before the deadline than to have a million features and two weeks to write the project report!

Report key sections and content

- There is no such thing as a typical report structure.
  - It will vary depending on the project.
  - However, the following structure is provided as a general guide.
  - If it does not seem relevant to your particular project, then adapt the suggested chapter headings.

Suggested content for report

(aligns with marking criteria presented later)

Title Page

Abstract

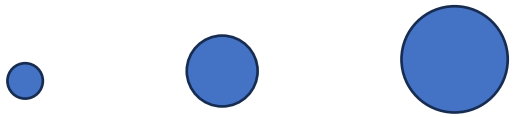
Table of contents

1. Introduction
2. Aims and Objectives
3. Background/Literature review
4. Methodology and Methods
5. Requirements specification and Design
6. Implementation
7. Testing and Evaluation
8. Results/Findings and Discussion
9. Conclusions/Recommendations/Future Work

References

Appendices

- User Manual, Relevant code, Technical information
- AI tools and prompts used



In your project you may need less than nine chapters, i.e. combine content of two or even three of these chapters into one.

Title Page

- Title of the project
- Name of the author
- Date
- …

"*This report is the result of my own work except where explicitly stated in the text. The report may be freely copied and distributed provided the source is explicitly acknowledged* "

Abstract

- One page that summarises the project (refer to aims, problem, approach to solution) and the main findings or results.
- Unless necessary should not contain technical details or refer to other sources from the literature

Introduction

- Brief description of the topic of the project
- In short what you hoped to achieve, why the topic is relevant or of interest to you… (not more than 2-3 paragraphs)
- Brief background the setting of the project in the context of other relevant work or theories or results. How this setting influenced the project. Refer to the
  - Context – a description of the area of the project
  - Key relevant sources about the project
  - Short description of 2-3 similar applications
- Short description of challenges and where your project fits in the area
- Finish by introducing the structure of report (what you will cover in each chapter).

The Introduction chapter is typically a couple of pages long (I think at least 1 page is needed to cover the above issues)

Aims and Objectives

- A high-level description of the project and its main goals
- What you plan to do in order to achieve the aims of this project
  - Consider Objectives of the project as necessary steps to achieve the aims/goal
  - For many projects, it helps to consider Objectives as Workpackages, which produce measurable outcomes, that markers can easily identify and assess whether they were achieved or not

Objectives example-1

- If you are creating a piece of software about a potential client, then this part might be straightforward since your aim will largely be influenced by the user requirements (but do not just list a set of requirements and consider those as the Aim)
- Then your project's objectives may be:
  - Gather Requirements and translate these requirements into functionalities/features of your system.
  - Design and implement software
  - Testing the software using software testing methods
  - Evaluate the software using user testing methods.

Objectives example-2

- If your project is about predictive analytics, then perhaps your aim is to investigate the potential of neural network architectures to model data and produce useful outputs to inform an investment strategy.
- Normally project objectives should include:
  - Critically review existing approaches;
  - Formulate the problem and identify data and features to be used, as well as an appropriate evaluation framework;
  - Identify appropriate neural architectures and train these to produce outputs that could be used to inform an investment strategy
  - Test proposed models in real-world conditions and against appropriate benchmarks;
  - Optimise the model by carrying out a study of the importance of model parameters and features used, and test again to validate the approach.

Background/literature review

- What the reader needs to know to understand the rest of the report. Although the background/review can build on the material included in the proposal, in the report you can present a more extensive coverage of the relevant areas and offer a critical analysis.
- Take into account any markers' comments/feedback on Moodle!

Methodology and Methods

Describe the overall project methodology and the methods that will be used to meet the objectives of the project. A separate chapter on methodology and methods may not be relevant for some projects, in such case information/material about methodology and methods is included in other chapters of the report.

- For software projects Methodology & Methods include: software engineering methods, design methods, programming languages, tools and technologies, testing methods
- For data science projects M&Ms include: relevant theoretical methods, data analytics/machine learning workflows/pipelines, data models, machine learning algorithms, packages (Keras, etc),

Methodology and Methods

Desc[illegible] the overall proje[illegible] methodol[illegible]nd the methods that will be used to [illegible] objecti[illegible] the [illegible] A separat[illegible] chapter on methodology [illegible] some projects, in such case infe[illegible] methods is incl[illegible]

Always explain what methods/tools are used and why -- justify their adoption in the context of your project!

- For soft[illegible] [illegible]ude: software engineering met[illegible] [illegible]ng languages, tools and techn[illegible]s, [illegible] m[illegible]
- For data science projec[illegible]&Ms inc[illegible]e: relevant theoretical methods, data analytics/[illegible]achine learn[illegible]ng workflows/pipelines, data models, machine learnin[illegible] algorithms, packages (Keras, etc),

M&M example

For the software project example, assuming that you have four objectives, i.e. Requirements specification, System/Software Design, Implementation, and Testing/Evaluation

1) Start by discuss software development methodologies and present which approach you will use; e.g agile, iterative, waterfall?

2) Organise the Methods section per objective.

-Requirements: explain what methods you will use to gather requirements, critically evaluate existing approaches etc.; prioritise requirements

- Design: how will this be done? What methods, use cases, UML? Will you use any tools? How will you make sure that your design meet the requirements?

- Implementation: how it will be done; schematic of system/software architecture, components etc What tools, platforms, libraries will you use/reuse?

- Testing: what methods, data/test cases etc. How will you evaluate- any metrics?

- Fine tuning or optimisation: sometimes this is optional but there are projects which is necessary to include it.

Requirements specification & Design

- Present requirements analysis and design of a system, software, or data processing/AI/machine-learning pipeline, explaining how your analysis and design meet the aims and objectives of the project.
- Comparison of different choices for methods, algorithms and data structures will add value to your report. Describe overall system structure, Dataflow, entity- relationship and object diagrams, as appropriate, help readers appreciate your work more and demonstrate deep understanding.
- Tailor/customise methods, algorithms, data structures, or settings to your project. You can refer to your Literature review/Background to support your arguments and the choices you have made for your project, but your design should refer to your project, i.e. avoid generic material.

Implementation

- Focus on important points that you want the examiners to pay attention to rather than attempting a detailed description of the whole program/system.
- Program documentation is not a substitute for a project report. Documentation is meant to be consulted as and when necessary and can be included in an Appendix

Testing & Evaluation

- Include a description of the test and/or evaluation plan, i.e. how the program/system/model was verified (e.g. software testing, simulations, experiments, user testing)
- How the overall solution was evaluated (e.g., depending on the project, this may involve user study, questionnaires, use cases, test cases, simulations). You can highlight some of the test results in the main text and put the actual test results in an appendix and refer to those in the discussion. This section may also include a user evaluation, where appropriate.
- For software projects, a long table of features (of which **only a small sample is** given in the report) with tick boxes that they have been tested. Ideally, this should be done automatically with a dedicated tool such as Selenium (these tools provide automation suites and tests). Unit testing, in the report demonstrate one of the tests and refer the reader to the relevant directory in your submission.
- For data science project, systematic testing of your model should be demonstrated, i.e no some random experimental results. Describe your experimental design and its rationale, organise your experiments and present their results.

Results/Findings & Discussion

- The project outcome. This might be:
  - Data collected
  - Demonstration of the working project, including screenshots
  - Tables and figures that demonstrate system/model performance according to different metrics
  - If the project includes experiments or simulations: why were certain experiments carried out but not others? What were the important parameters in the simulation and how did they affect the results? If there are many graphs and tables associated with this chapter they may be put in an appendix, but it is generally better to keep these close to the text they illustrate, as this is easier for the reader.
  - Errors and their analysis/explanation

Remarks:

This is where you want to show why your project is good, so make it interesting:

- Put only interesting things from the project
- Identify limitations
- Refer the reader to a user manual, details results/figures/tables in the appendix for the long list of features or tests.

Results/Findings & Discussion

- The project outcome. This might be:
  - Data collected
  - Demonstration of the working project, including screenshots
  - Tables and figures that demonstrate the system performance according to different metrics
  - If the project included certain experiments carried out, graphs and simulations and tables associated with the appendix, but it is generally better this is easier for the reader.
  - Errors and

critical evaluation of the results and reflection on the approach and the project, including strong and weak points, lessons learnt, design decisions which, looking back, would be made differently, ways in which the project could be improved or extended.

Remarks:

This is where you want to show why your project is good and make it interesting

- Put only interesting things from the project
- Identify limitations
- Refer the reader to a user manual, details, results/figures/tables in the appendix for the long list of features or tests.

Conclusions/Recommendations

- If you were trying to prove some hypothesis, did you succeed? (the project does not need to be a success)
- Did you achieve all your objectives?
- Challenges
- Future directions
- Experience gained, things you learned to do better

References

- All references should be cited in the body of the report. A typical reference in the report might take the form, “Donar and Kebab (1996) suggest that high cholesterol levels do not lead to heart disease....” or “empirical eating studies show that.... (Donar and Kebab, 1996)”.

- You can add Bibliography, a list any relevant literature that has not been cited in the report.

- You should use the Harvard Referencing system

Appendices

- **Link to your demo** (5minute video recording- optional but strongly encouraged)
- User Manual
- Relevant code
- Technical information
- Transcripts of interviews (these are the raw data used in the project)
- Detailed data
- **Extensive presentation of test or evaluation results** (main text should point to appendix/intergrate these tests/results in the discussion)
- **AI tools and prompts used** (even if the only tool used is Grammarly!)

Ethics, risks and mitigation strategies

With respect to **Ethics**, discuss with your supervisor whether your project has ethical implications.

- Most of the time, this is routine.
- Students have to fill in an Ethical Review Form (available on the project Moodle page- KEY INFORMATION tile) and include it in the Project proposal.

What is **risk assessment** for your project?

- For example, if you have open research questions, or you are not sure the data are available or in a format that can be directly used for ML, “fallback” plans should be laid out; e.g. you can provide a risk analysis plan and risk mitigation strategies.
- Explain risk impact score for each of the risks you have identified and order them to identify the critical risks to your project.

Using and referring to existing material/sources and assessment offences

- Nobody expects you to reinvent the wheel
- You are allowed to use existing work as a basis for your project, provided that:
  - you reference this work properly
  - there is enough of your own work in the project
  - you don’t have a collage of other people’s work
- You should not quote large chunks of text from another source! Excessive quotation indicates that you're not thinking enough for yourself, and markers may consider it a poor substitute for critical analysis. Doing so may reduce the marks awarded in certain criteria.
  - The simplest thing would be to summarize it in your own words and reference the source.
  - The ideal thing is of course to provide a critical analysis of other people’s work, identify gaps and provide insights that can inform the work in your project or can help you justifying decisions taken about your own project.

References

- References include the full bibliographic information about the source, such as the author(s)'s name(s), date of publication, title of work, place of publication, and publisher. This information is usually given in the section called References or Bibliography at the end of the text. The key principle is that you should give enough information to allow another person to find the source for themselves.

- Here are some examples using the Harvard referencing system:
- [when you are referring to a book]

Lewin, K., 1951. Field Theory in Social Science. New York: Harper and Row.

- [when you are referring to a chapter in a book, where 'ed.' means editor, and 'edn.' means 'edition']

Piaget, J., 1970. Piaget's theory. In: P. Smith, ed., Handbook of child psychology. 3rd edn. New York: Wiley, 1970, pp. 34-76.

References

- [when you are referring to a journal article]

Holmqvist, M., 2003. A Dynamic Model of Intra- and Interorganizational Learning. Organization Studies, 24(1), 95-123.

- [when you are referring to a webpage]

W3C, Web Accessibility Guidelines and Techniques, available online at http://www.w3.org/WAI/guid-tech.html. Last accessed 12/02/2023.

- Independent of their type (e.g. book, article, webpage), all references are included at the end of your document in alphabetical order starting from the author's name as in the example above.

- Harvard guide to citing references. Available to online at: https://www.slideshare.net/leshere/harvard-citation-hlp

References

- All references should be cited in the body of the text. A typical reference might take the form, “Donar and Kebab (1996) suggest that high cholesterol levels do not lead to heart disease....” or “empirical eating studies show that.... (Donar and Kebab, 1996)”.

- You can add Bibliography, a list any relevant literature that has not been cited in the report and does not appear in the References.

- **You should use the Harvard Referencing system**

Using and referring to existing material/sources and assessment offences

The following are all examples of assessment offences:

- Plagiarism - the presentation of another person's thoughts or words or artefacts or other output in such a way that they could be assumed to be your own.
- Collusion - producing a piece of work for formal assessment with the assistance of another person, or persons, when the assignment was to have been undertaken and completed by you working individually.
- Contract Cheating - where a student submits work for assessment in part or in entirety where they have used one or more of a range of services provided by a third party, such as an essay mill, which is not permitted. Please be aware that services like Chegg are willing to cooperate with the School on contact cheating – in examinations they have provided us details of the question submitters and full answers published.
- Use of text generated from AI tools, such as ChatGPT– any use of AI should be acknowledged in your text by providing in-text citation, and the prompts that were used in an appendix. Copying AI generated text, especially verbatim, indicates poor assimilation of the material and may lead to lower marks.

Plagiarism

- Unfortunately, there have been a few cases of plagiarism recently here at the School
- This is an important topic, as the penalty for plagiarising work can be very severe
- Plagiarism is using words and ideas from another text without proper acknowledgement
- The College's Policy on Assessment Offences lists examples (http://www.bbk.ac.uk/reg/regs/) and penalties
- ***The Avoiding Plagiarism module on Moodle:*** you can self-enroll https://moodle.bbk.ac.uk/enrol/index.php?id=45

Marking criteria

- ***Specification and design- 20%***: the specification and design of the system/software/data analysis framework, AI or machine learning pipeline clearly demonstrate how to meet the aims and objectives of the project and the requirements. The report explains how all components fit together in a coherent way, underpinned by relevant literature.

- ***Implementation, or execution of research- 30%***: The key stages of the implementation or research are clearly explained and justified based on relevant literature and/or research conducted in the context of the project. The implementation or research for a challenging problem is appropriate to the project objectives and requirements and is carried out to a high standard as demonstrated in the report.

- ***Testing and evaluation, results, analysis, and reflection- 30%***: The report uses appropriate testing and evaluation methods and justifies their adoption based on relevant literature and/or research conducted in the context of the project. It describes systematic testing and evaluation activities, and a solution (e.g. system, software, or a data analysis/AI/machine learning approach) that demonstrates deep insight into the problem/research question/aims and objectives. The report attempts to provide a critical and justified reflection upon the project's contribution and its limitations. Key results are accurately and critically analysed. Relevant conclusions are drawn.

- ***Presentation of the proposal/report and documentation- 20%***: The report is self-contained, providing background and context. It is well-organised and demonstrates the concepts presented. Relevant literature is exploited and cited to support the analysis and the discussions, and in general the report analyses and explains complex issues clearly. The overall approach and solution are well-justified. The report cites relevant resources using consistent style and is of professional quality.

Submission, late submission, deferral

- Follow the guidelines in the submission page:
  - Submit the report and your code (GitHub repository; Students must obtain their allocated repository before August 15 – see Project Guidance and tile “Report Submission” for Oct cohort, or tile “January start” for Jan cohort)

Check the deadlines on Moodle:

https://moodle.bbk.ac.uk/course/view.php?id=42460

- There is a normal deadline and a cut-off deadline (but penalty applies)
- If you miss the cut-off deadline, you won’t be able to submit your report.
- To avoid the penalty, you must claim mitigating circumstances and provide evidence to support your claims. Please contact admin directly for advice.

What happens if I submit late?

- If you submit your report/project after the deadline and before the cut-off date, your grade will be capped at a pass
- If you are late even by one second, you will have to submit mitigating circumstances.
- **Please contact admin directly for advice.** The PD and the project coordinator cannot help in this case and should not be contacted for this purpose.

What to avoid

- Writing the report in the last two weeks
- Sending the report to the supervisor getting a fast reply fixing, extending and sending it to the supervisor again expecting a quick reply.
- Disappearing for a long time and then writing long texts at the last moment and sending it to supervisor

Questions?