COMP0035 2023/24 Coursework 1 specification

Version: Approved. Version 1.0 27/09/23

Contents

- Introduction
- Getting started
- Technologies to use
- Coursework content
- Submission
- Marking
- Appendix

Introduction

This document specifies coursework 1 which is worth 50% of the assessment marks available for the module.

The aim of this coursework is for you to select and apply relevant software development and data science techniques for aspects of a project lifecycle to define and design two data science web applications. You are also expected to evidence the use of appropriate software engineering tools and techniques in your work.

weixin: scs_ryan

This assessment includes:

- Data preparation and understanding
- Product and project definition
- Tools and techniques
- Methodology (groups only)

You will submit a portfolio of work with evidence for each section. The required evidence is detailed in section 5 Coursework content.

Coursework 2 will continue from coursework 1, covering further aspects of the lifecycle.

For students continuing to COMP0034 Application Programming for Data Science, in COMP0034 you will develop and test the applications that you design in COMP0035.

Getting started

There are 3 steps in this section:

1. Choose to work as an individual or group
2. Be allocated a data set
3. Accept a GitHub classroom assignment

Step 1. Choose to work as an individual or group

Please carefully consider the choice as you cannot change your decision during the course i.e. if you work as a group for coursework 1 you must also work with the same group for coursework 2 (likewise for individuals).

Groups should carefully consider the potential for conflicting timetables and competing time demands in your 3rd year which can make group work more challenging.

If you choose to work as a group, read thegroup work guidance in Moodle.

Use the selection task on Moodle to register your choice.

Step 2. Be allocated a data set

Data sets are allocated as soon as the group selection task is completed, typically at the end of the first week of the module.

Where multiple groups/individuals are allocated the same dataset, ensure that you do not collude with other students ; collusion is considered Academic Misconduct. It is expected that those using the same dataset will have very different coursework as you will each have your own product ideas, target audience, questions etc.

Step 3. Accept a GitHub classroom assignment

Create a new repository by accepting the GitHub classroom assignment. Instructions are also given in the first tutorial.

weixin: scs_ryan

Individuals

1. Login to GitHub.com.
2. Click on the GitHub classroom link for individuals
3. Accept the assignment.
4. If prompted, accept to join the comp0035-ucl organisation.

Groups

1. Login to GitHub.com.
2. Click on the GitHub classroom link for groups
3. **One person only (first person from the group)**
    1. Accept the assignment.
    2. If prompted, accept to join the comp0035-ucl organisation.
    3. Create the group (team) using the same name as your Moodle group name (e.g. Group1, Group2 etc.).
4. **All other group members**
    1. Accept the assignment.
    2. If prompted, accept to join the comp0035-ucl organisation.
    3. Select your group (team) name from the list.

Technologies to use

Technologies that must be used

Technology	***Notes***
GitHub for source code control	All elements of the coursework are to be maintained using source code control. Source code control should be in GitHub, with repositories created in GitHub classroom to allow tutors and PGTAs to gain access. If you can't use GitHub for some reason please contact the course tutor to agree alternative source code control.
Python coding environment	You need a python coding environment. Guidance can be found on Moodle.
PDF or Markdown for text [1]	Text created using any software such as Word, Powerpoint, Excel etc must be saved as `.pdf`. Markdown is a widely used format in software development so you may use this. GitHub provides a markdown guide though there are many freely available guides online.
pandas and matplotlib for data preparation	You must use these libraries for the data preparation and exploration. Do not use additional libraries unless absolutely necessary (i.e. if pandas or matplotlib do not have the necessary capability).

weixin: scs_ryan

[1] Students who prefer audio to text may audio record any textual submissions instead. Use .mp3 format. Do not create video as you can be more easily identified.

Use of AI

You **are permitted**, but not required, to use AI (e.g. GitHub Copilot, ChatGPT) **only** for the data preparation code.

You **are not permitted** to use AI for any other section (including the explanation of your data preparation code).

You must state and explain your use of AI in the 'tools and techniques' section of the course content.

Coursework content

Creating the content

The following subsections state what must be produced. You can choose to provide any additional relevant items that you believe are useful.

Variances for groups and individuals are noted within each section.

Use techniques included in the teaching materials, or other techniques that are typically used in software engineering and data science. Please provide a relevant reference if you use a technique not included in the course materials.

Add all the content to your GitHub repository.

Add all the written content is to be added to the same file in the order shown below for submission. Please use the section numbering shown below:

```
1. Data preparation and understanding
   1.4 Explanation of code preparation & understanding
2. Product and project definition
   2.1 Problem statement
   2.2 Product overview
   2.3 Persona
   2.4 Project goal & objectives  /  Questions
3. Tools & techniques
   3.1 Source code control
   3.2 Linting
   3.3 Project planning and tracking
   3.4 Use of AI
4. Methodology
   4.1 Methodology selection
5. References
```

weixin: scs_ryan

Do not change the names of the files given in the starter code and use the filenames given in the following sections. Automated scripts are used to separate files for marking purposes, if you change the filenames your coursework is likely to be missed from marking.

1. Data preparation and understanding

The purpose of this section is:

- to demonstrate that you understand the data. You should have a good understanding of the data set structure, attributes and their data types, statistics, distribution etc.
- to prepare the data for later use in developing the web apps (products). For example, ensure you have the data that you need, remove any unnecessary data, address any data quality issues you discover (e.g. missing values, inappropriate data types), etc.

For this section add evidence for the following to the indicated files:

Item	*Individual*	*Group*	*Filename*
1.1 Python code to prepare/understand the data	✓	✓	data.py
1.2 Original data set	✓	✓	dataset.(csv/xlsx)
1.3 Prepared data set	✓	✓	dataset_prepared. (csv/xlsx)
1.4 Explanation of code preparation & understanding	✓	✓	coursework1. (pdf/md/mp3)

Code and data set

Write python code that makes use of the pandas and matplotlib libraries to explore and prepare the data.

Include the original dataset and the dataset after the code has been applied to it.

You may create charts as part of your preparation and understanding e.g. to examine distributions, outliers etc. Do not spend significant effort formatting any charts you create as the visual aesthetics not assessed.

weixin: scs_ryan

You may have a data set that is already well prepared with few corrections or changes needed. You should still demonstrate in the code that you have examined the data set to show that this is the case, and you should still explain your findings.

Groups are expected to spend more time than individuals on the code. This is because all group members must understand the code that is created.

Code quality will be considered in the marking. This includes:

- code structure (e.g. use of functions and/or classes)
- adherence to appropriate python conventions (e.g., PEP8 style guide , PEP275 docstring conventions)
- documentation (comments, docstrings) and more generally whether your code is easy for others (in this case the marker) to read and understand.

Code quality will not consider file length. Typically, code that exceeds 300-400 lines of code would be split into modules. For this coursework, please ignore this aspect of code quality as for marking purposes all the code must be in a single python file.

Explanation

Describe and explain the results of your code to prepare and explore the data. Explain any decisions that you took.

For example, if you identify code quality issues then explain how you addressed these; or if you chose not to address them, then explain the reason for not addressing them.

There is no word limit for this as it depends on the format used and the number of decisions made.

2. Product and project definition

The purpose of this section is to provide those working on, or those interested in, the project with an understanding of what the project will deliver (i.e. the web apps products); why these apps are needed; and who they are being created for.

For this section add all the following to a single file named `coursework1.pdf/md`:

Item	***Individual***	***Group***	***Filename***
2.1 Problem statement	✗	✓	coursework1.pdf/md
2.2 Product overview	✓	✓	coursework1.pdf/md
2.3 Persona	✓	✓	coursework1.pdf/md
2.4 Project goal & objectives **or** questions	✗	✓	coursework1.pdf/md

weixin: scs_ryan

2.1 Problem statement

Write an appropriately structured problem statement, or a product vision, that summarises the problem or opportunity your web app product (App2) will attempt to provide a solution to.

2.2 Product overview

You will be designing two web app products in coursework 2. The product overview will be very short, likely a few sentences. It should make clear the type of web product (web app) that you plan to develop and the focus of the data. There is no formal technique or template for this. It provides a context for the marker.

The two web apps products are:

- **App1**: A REST API that will allow developers to programmatically access and use your dataset in their apps. The target audience for this is software developers.
- **App2**: A web app product that makes use of your dataset via the REST API (App1). You can choose from the following types of app:
  - a. A data visualisation / dashboard app
  - b. An app that deploys a machine learning model
  - c. A web app that uses the data in some other way e.g. to generate content that has interactive features

See Moodle for links to real world apps that are examples of 2a and 2b. It is expected most students will create an app of type 2a or 2b. 2c is intended for those who can imagine an app using the data that doesn't fit either a. or b.

2.3 Persona

Create a persona(s) that describes the target audience(s) for App2 only. A single persona is typical though there may be instances where you intend your web app to be used by multiple personas in which case you can create as many as needed. Creating more personas is not an indicator for higher marks.

For App1 the target audience will be software developers who need to use your REST API in their own web app. You do not need to create a persona for this target audience.

2.4 EITHER 'goals and objectives' OR 'questions'

Provide either option 1 or option 2. Providing both will not lead to more marks.

Option 1: Write project goal(s) and objectives. Objectives should be SMART. These can focus on just App2 or both apps if you prefer.

Option 2 (*for data visualisation apps only*): Write questions you will attempt to answer by creating visualisations (charts) using your dataset. The questions should relate to the target audience and the problem statement.

weixin: scs_ryan

3. Tools and techniques

The purpose of this section is to demonstrate appropriate and effective use of relevant software engineering tools and techniques.

For this section add evidence for all the following to `coursework1.pdf/md`:

Item	***Individual***	***Group***	***Filename***
3.1 Source code control	✓	✓	coursework1.pdf/md
3.2 Linting	✗	✓	coursework1.pdf/md
3.3 Project planning and tracking	✗	✓	coursework1.pdf/md
3.4 Use of AI	✓	✓	coursework1.pdf/md

3.1 Source code control

Add the URL for your repository to `coursework1.(pdf/md)`.

Make regular use of source code control throughout the project. Use it for all the coursework documents, not just code.

For **groups** all members are expected to show evidence of source code control. This can be seen in the commit history.

3.2 Linting

Make use of a linter to demonstrate how effectively your code meets Python style standards such as PEP8, PEP257.

In `coursework1.(pdf/md)`:

- provide evidence of the results of running the linter.
- explain how you used this to improve your code quality.
- state which Python linter you used.

3.3 Project planning and tracking (groups only)

Provide evidence of regular project planning and tracking in `coursework1.(pdf/md)`.

These should indicate what all members of the group did (use initials rather than full names). A typical format might be similar to the Scrum daily stand up:

- What did you do in the last reporting period?
- What do you plan to do in the next reporting period?
- State any issues that are blocking progress

weixin: scs_ryan

You may use any planning and tracking technique and tool that you think is relevant. Options include but not limited to:

- Add .txt or .md files to your GitHub with your weekly plan/progress/issues
- Submit reports using the optional weekly Moodle reporting task (see Further resources section for each week)
-

Use GitHub Projects to add and assign tasks

Past students have also used Kanban-style boards, Sprint plans/reviews, etc. When using external tools you must save the evidence (e.g. to PDF) and include it in your submission. A URL hyperlink to an external tool is **not acceptable** for marking.

The exact timing of the report is not graded; though reports that are submitted just before the assessment deadline or where there is no evidence of timing are not considered to as ‘regular’ reporting.

3.4 Use of AI

You are permitted to use code assistance AI (e.g. GitHub Copilot, ChatGPT)**only** for the data preparation code.

If you did not use AI, then please clearly state “AI not used” in this section.

If you did use AI then explain your use of AI in this section. For example, state which tool(s) you used then explain how you used them and to what extent the resulting code was used.

Note, if you do not explain your use of AI here, and it is found to have been used in the coursework, then it may be considered as plagiarism.

4. Methodology

The purpose of this section is to demonstrate that you can select an appropriate methodology for a given situation. This is one of the module learning outcomes.

For this section add the following to `coursework1.(pdf/md)`:

Item	***Individual***	***Group***	***Filename***
4.1 Methodology selection	✗	✓	coursework1.pdf/md

Select a methodology, or combination of methodologies, that is suitable for use in the project.

weixin: scs_ryan

Justify your choice. For example, you could define selection criteria and review how well your choice meets these. Use relevant literature to support your views and (if relevant) your own experience.

There is no ‘correct’ choice, what is important is that you demonstrate why what you have chosen is relevant to the particular nature and circumstances of your project.

You can consider methodologies from data science and software engineering.

The ‘project’ in this instance refers to all the activities that you will carry out across both coursework 1 and 2 in both COMP0035 and COMP0034. The activities will include project and product definition, data preparation, requirements analysis, application design, development, testing.

There is no word limit for this as it depends on the format used to present your response. For example, some use tables, others write textual paragraphs that tend to be longer. There are past examples on Moodle.

5. References

The methodology section is expected to have some references.

Other sections are a report of your work, rather than a dissertation. As a result you are unlikely to have many references.

If you do have any then please state them here.

5. Submission

Please refer to Moodle for the deadline date and time.

Submit your work on Moodle as a single .zip in the assignment submission (see Checklist in the next section). Moodle is used as the submission date/time and to authenticate students. GitHub is NOT an acceptable alternative for submission.

Check carefully that you have the correct files with the correct file names. Automated scripts will be used to separate elements for marking; if you change the file name your work is likely to be missed.

Files that are only provided as URLs that link to external locations will be excluded from marking consideration since they may be modified after the coursework is submitted.

GitHub provides an option to download repository contents as a zip.

You can include other files in the zip; though marking will only consider the files stated in the coursework content sections above.

DO NOT add your .venv folder to the zip file, this creates unnecessarily large zip files.

weixin: scs_ryan

Submission checklist

Check you have the following content in each file

Section	***Filename***	***Item***	***Individual***	***Group***
1. Data preparation & understanding	data.py	1.1 Python code to prepare/understand the data	✓	✓
	dataset.(csv/xlsx)	1.2 Original dataset	✓	✓
	dataset_prepared.(csv/xlsx)	1.3 Prepared dataset	✓	✓
	coursework1.(pdf/md/mp3)	1.4 Explanation of code preparation & understanding	✓	✓
2. Product/project definition	coursework1.(pdf/md/mp3)	2.1 Problem statement	✗	✓
		2.2 Product overview	✓	✓
		2.3 Persona	✓	✓
		2.4 Project goal & objectives **or** Questions	✗	[illegible]
3. Tools & techniques	coursework1.(pdf/md/mp3)	3.1 Source code control	✓	✓
		3.2 Linting	✗	✓
		3.3 Project planning and tracking	✗	✓
		3.4 Use of AI	✓	✓
4. Methodology	coursework1.(pdf/md/mp3)	4.1 Methodology selection	✗	✓

weixin: scs_ryan

6. Marking

Mark allocation

Marks will be allocated for the coursework as follows.

It is expected each person will spend 25 hours on the coursework during the first 5 weeks of the module and part of reading week.

Groups will have 100 hours in total, 76 hours to complete the deliverables and 24 hours co-ordination overhead. An overhead for co-ordination is expected for groups. This has been assumed at 1 hour per person per week for 6 weeks (total 24 hours). This allows for a 30-minute weekly meeting as well as review of work items.

While not exact, an indication of the hours of effort given the marking weighting is shown below.

Section	***Group***	***Individual***
Data preparation and understanding	30% (23 hours)	60% (15 hours)
Product/project definition	20% (15 hours)	20% (5 hours)
Tools and techniques	20% (15 hours)	20% (5 hours)
Methodology	30% (23 hours)	-

Mark calculation

Marks for individuals will be given for each section. Each section is then weighted using the % in the table above to calculate the mark.

weixin: scs_ryan

A group's raw mark is calculated using the same method as for individuals. All members of a group are required to complete an IPAC assessment that includes their own and peers performance. The calculated IPAC rating will then be applied to the group raw mark to calculate the specific mark for each person within the group, i.e.

```
individual group member mark = group raw mark * individual's IPAC rating
```

IPAC guidance is given in the appendix .

Grading criteria

The coursework will be assessed according to the standards set in the standard UCL Computer Science grading criteria (see copy on Moodle in the Assessment section).

The criteria most relevant to this assessment are:

- 1 Quality of the response to the task set: answer, structure and conclusions
- 2 Understanding of relevant issues
- 3 Engagement with related work, literature and earlier solutions
- 4 Analysis: reflection, discussion, limitations
- 5 Algorithms and/or technical solution (note: this applies to the code aspects only)
- 8 Writing, communication and documentation
- 9 Formatting aspects, visuals, clarity, references

The following tables gives indicators that are more specific to this coursework. Given the open-ended nature of the coursework it is not possible to describe every possible aspect that could be considered for each level. Please use the tables below as a guide and not an absolute.

Some aspects relate only to groups, individuals, or both. Check ‘section 5 coursework content’ for what each needs to include.

NOTE: Marks are not awarded for doing more. For example, individuals choosing to do items identified for groups will not achieve higher marks as this would be unfair since groups do not have the same opportunity.

Data preparation and understanding

Grade band	***Generic CS Descriptor***	***Coursework-specific indicators***
Distinction 90+	Exceptional response with a convincing, sophisticated argument with precise conclusions. Exceptional grasp of complexities and significance of issues. Exceptional thought and awareness of relevant issues. Sophisticated sense of conceptual framework in context.	
Distinction 70-89	A distinctive response that develops a clear argument and sensible conclusions, with evidence of nuance. Thorough grasp of issues; some sophisticated insights. Evidence of innovative analysis. Concepts deftly defined and used with some sense of theoretical context.	There is evidence of a thorough understanding of the data. Analysis and preparation is clearly explained and decisions justified. Code quality is high with effective use of functions or classes, and adheres to relevant Python standards (e.g. PEP8, PEP257).
Merit 60-69	A sound response with a reasonable argument and straightforward conclusions, logical conclusions. Sound understanding of issues, with insights into broader implications.	There is evidence of a good understanding of the data. Steps have been identified that allow the data set to be immediately useful in the development phase of the project. Code quality mostly adheres to Python standards.

weixin: scs_ryan

Grade band	*Generic CS Descriptor*	*Coursework-specific indicators*
High pass 50-59	A reasonable response with a limited sense of argument and partial conclusions. Reasonable grasp of the issues and their broader implications. Reasonable reproduction of ideas from taught materials. Rudimentary definition and use of concepts.	There may be limited evidence that the student has understood and/or prepared the data. Any decisions taken are not clearly explained. Reasonable code, using basic required concepts, several flaws in implementation.
Low pass 40-49	An indirect response to the task set, towards a relevant argument and conclusions. Rudimentary, intermittent grasp of issues with confusions. Analysis relying on the partial reproduction of ideas from taught materials. Some concepts absent or wrongly used.	Insufficient evidence that the data has been understood. Rudimentary preparation code, but mostly incomplete.

weixin: scs_ryan

Product and project overview

Grade band	*Generic CS Descriptor*	*Coursework-specific indicators*
Distinction 90+	Exceptional response with a convincing, sophisticated argument with precise conclusions. Exceptional grasp of complexities and significance of issues. Expert-level review and innovative synthesis (to a standard of academic publications). Exceptional thought and awareness of relevant issues. Sophisticated sense of conceptual framework in context. Reads as if professionally copy edited.	

Grade band	*Generic CS Descriptor*	*Coursework-specific indicators*
Distinction 70-89	A distinctive response that develops a clear argument and sensible conclusions, with evidence of nuance. Thorough grasp of issues; some sophisticated insights. Extensive reading and thorough understanding of literature consulted. Excellent critical analysis of literature. Evidence of innovative analysis. Concepts deftly defined and used with some sense of theoretical context. Style and word choice show fluency with ideas and excellent communication skills. Referencing is impeccable.	A range of relevant artefacts are presented. The problem is clearly defined with relevant information. The target audience is described using an appropriate format and with a level of detail that will support later activities concerned with the dashboard and app design. Questions to be addressed by analysis of the data set have been clearly formulated and articulated and demonstrate insight. They are clearly linked to both the problem and audience.
Merit 60-69	A sound response with a reasonable argument and straightforward conclusions, logical conclusions. Sound understanding of issues, with insights into broader implications. Evidence of plentiful relevant reading and sound understanding of literature consulted. Evidence of student's own analysis. Concepts defined and used systematically/ effectively. Style and word choice work well to convey most important ideas. Well documented. Consistent referencing.	The problem and target audience are clearly defined and are provided in a format that is relevant for the nature of the project. Questions to be answered by the analysis of the dataset are clearly given but may show minor inconsistency with the data set provided.
High pass 50-59	A reasonable response with a limited sense of argument and partial conclusions. Reasonable grasp of the issues and their broader implications. Evidence of relevant reading and some understanding of literature consulted. Reasonable reproduction of ideas from taught materials. Rudimentary definition and use of concepts.	The problem statement is stated but may lack some clarity. The description of the target audience may be superficial and/or of a format that is not specific to the nature of the project. The questions to be answered by the data analysis may be unclear or inconsistent with the data set.

weixin: scs_ryan

Grade band	***Generic CS Descriptor***	***Coursework-specific indicators***
Low pass 40-49	An indirect response to the task set, towards a relevant argument and conclusions. Rudimentary, intermittent grasp of issues with confusions. Significant omissions in reading with weak understanding of literature consulted. Analysis relying on the partial reproduction of ideas from taught materials. Some concepts absent or wrongly used. Style and word choice seriously detract from conveying of ideas.	

Methodology

weixin: scs_ryan

Grade band	***Generic CS Descriptor***	***Coursework-specific indicators***
Distinction 90+	Exceptional response with a convincing, sophisticated argument with precise conclusions. Exceptional grasp of complexities and significance of issues. Expert-level review and innovative synthesis (to a standard of academic publications). Exceptional thought and awareness of relevant issues. Sophisticated sense of conceptual framework in context. Reads as if professionally copy edited.	

Grade band	*Generic CS Descriptor*	*Coursework-specific indicators*
Distinction 70-89	A distinctive response that develops a clear argument and sensible conclusions, with evidence of nuance. Thorough grasp of issues; some sophisticated insights. Extensive reading and thorough understanding of literature consulted. Excellent critical analysis of literature. Evidence of innovative analysis. Concepts deftly defined and used with some sense of theoretical context. Style and word choice show fluency with ideas and excellent communication skills. Referencing is impeccable.	Choice of methodology is clearly explained and justified within the context of the project and shows evidence of nuance in understanding the issues involved in selecting and using an appropriate method. The discussion is supported by relevant references.
Merit 60-69	A sound response with a reasonable argument and straightforward conclusions, logical conclusions. Sound understanding of issues, with insights into broader implications. Evidence of plentiful relevant reading and sound understanding of literature consulted. Evidence of student's own analysis. Concepts defined and used systematically/ effectively. Style and word choice work well to convey most important ideas. Well documented. Consistent referencing.	A methodology has been selected that is appropriate for the project proposed.
High pass 50-59	A reasonable response with a limited sense of argument and partial conclusions. Reasonable grasp of the issues and their broader implications. Evidence of relevant reading and some understanding of literature consulted. Reasonable reproduction of ideas from taught materials. Rudimentary definition and use of concepts.	A methodology has been selected though it may not be clear why this is appropriate for the given project.

weixin: scs_ryan

Grade band	*Generic CS Descriptor*	*Coursework-specific indicators*
Low pass 40-49	An indirect response to the task set, towards a relevant argument and conclusions. Rudimentary, intermittent grasp of issues with confusions. Significant omissions in reading with weak understanding of literature consulted. Analysis relying on the partial reproduction of ideas from taught materials. Some concepts absent or wrongly used. Style and word choice seriously detract from conveying of ideas.	Methodology explanation is limited and/or the selected methodology is inappropriate for the nature of the project.

Software engineering tools and techniques

Source code control will be assessed using the commit history in the repository. You must include the URL in your submitted work.

There are no generic CS criteria relating to this aspect.

weixin: scs_ryan

Grade band	*Coursework-specific indicators*
Distinction 90+	Near flawless use of a range of appropriate tools and techniques. Likely to be evidence that tools not taught in the course have been carefully selected and used.
Distinction 70-89	Exceptional use of the expected tools and techniques.
Merit 60-69	Effective use of the expected tools and techniques.
High pass 50-59	Appropriate use of the expected tools and techniques, those the use of these may be less effective.
Low pass 40-49	Limited, or inappropriate, use of the expected tools and techniques.

The following questions are not exhaustive and are indicators only:

Source code control

- Appropriate:
  - has it been used regularly? i.e. not just for a day or two before the coursework

deadline
    - has it been used by all team members? Everyone needs to demonstrate competence
- Effective:
    - has it been used effectively? e.g. are the commit messages clear and specific?
    - are there typical files used appropriately e.g. README.md, .gitignore? Are appropriate files excluded e.g. virtual environments?
- Exceptional: is there any evidence of sophistication? e.g. branches, pull requests, other features of GitHub

Linting

- Limited: has a linter been used? e.g. is there evidence of the results of a linter?
- Appropriate: has linting been used to improve code quality? e.g. does a linter report show few issues, or is there are before/after linter report that shows a reduction
- Effective: is there an explanation of any issues that have not been addressed with appropriate reasons given for these? or an explanation of steps taken to address issues identified by the linter? Linting may be automated in GitHub Actions.
- Exceptional: is there evidence that goes beyond what was taught in the course?

Progress tracking

- Limited: some evidence but it is not clear that this was created weekly/periodically and may be of limited use in understanding the project progress.
- Appropriate: weekly reports submitted on Moodle.
- Effective: weekly reports submitted, and provide clear updates for all team members.
- Exceptional: goes beyond effective, for example, may include additional techniques e.g. evidence sprint planning/reviews; evidence of tasks/plans; etc.

weixin: scs_ryan

Appendix

The appendix contains supplementary guidance and is common to all assessments on this module.

Data sets

Only use the ethics approved data set

The data sets allocated to students on this course have been approved by the Computer Science Ethics Committee and signed by the Head of Department for use in this course. These data sources comply with GDPR and UCL ethics and data protection policies. You must not use any other data set for the coursework.

The approved data sets do not require data protection registration and since you will not be working with participants in your project there is no requirement to complete any further training relating to data protection and GDPR.

Data set selection was guided by the following principles:

- does not contain data related to, or collected from, humans. There are some very limited exceptions to this and these also demonstrate:
  - the data was gathered lawfully and in accordance with GDPR, and that this can be verified
  - consent was given by the human participants and this consent can be verified
  - the data is wholly anonymous
- is available in the public domain (and further does not requires userid/password to access it)
- is not sensitive
- grants permission for its use (i.e. there is an stated license permitting you to use it)

weixin: scs_ryan

Complete the mandatory ethics training

UCL computer science provides a series of videos aimed at introducing students to the ethical implications of using secondary data in computer science projects, and in particular the processes that must be followed at UCL.

You must complete the two mandatory ethics training videos as shown on Moodle.

Assessment

Examples of previous coursework

Partial extracts from previous submissions are given on Moodle as examples.

Full coursework examples are not provided since the coursework remains similar to the previous year, and it would then be too easy to simply copy and adapt the coursework from a previous student. Turnitin plagiarism detection is used.

You are encouraged to start work early on your coursework and use the tutorials to gain formative feedback.

Time estimates for assessment

An *average* module is expected to take around 150 learning hours.

To help you plan your time, the coursework is designed with the expectation you will spend the following time on it:

- 2.5 hours per week (all weeks except reading week)
- 7 hours during reading week
- 10 hours during Christmas/Easter periods

Each coursework should take 20-25 hours.

All assessments are based on projects that start in week 1 and run throughout the module. You are expected to make progress each week. There are weekly activities and checkpoints in Moodle to guide your progress.

Support and guidance

weixin: scs_ryan

Module staff will be available to provide support and guidance in weekly lab sessions.

The tutor will be available during a weekly module office hour (see Moodle).

You can ask questions at any time in the Moodle Q&A forum where anyone in the course is able to reply.

IPAC peer assessment for groups

The assessment for groups includes peer assessment using the UCL IPAC assessment software and process.

Please read the IPAC student presentation if you are not familiar with IPAC. All IEP students should be familiar with this as it was used in ENGF0001 and other IEP modules.

For each assessment all members of a group must complete an IPAC evaluation that includes their own and peers performance. This determines an individuals IPAC rating. The calculated IPAC rating will then be applied to the raw group mark.

For example, for a coursework graded at 60%:

- a student with an IPAC of 0.9 would receive a mark of 54%
- a student with an IPAC of 1.0 would receive a mark of 60%
- a student with an IPAC of 1.1 would receive a mark of 66%

The criteria used in the IPAC are:

Criteria	*Definition*
Attendance	Attending scheduled tutorial sessions, meetings with the TA and other meetings arranged outside of these
Effort	The student's attitude to work in and out of the class, reflects their attentiveness and contribution to discussion and work load
Quality of work	A reflection of how complete, accurate and informative the work the student produced was
Teamwork	Demonstrates a cooperative and supportive attitude

Note that:

- the self-promotion score, which is provided by the IPAC software, quantifies how a student has valued his/her contribution in regard to the contribution of the group (ratio of averages). If the self-promotion score is too high (over 1.25), then the software ignores the scores given by that student from the calculations.
- 0.5 is the minimum IPAC score a student needs to have before being allowed to take his/her peer marks into account. The reasoning is that those students that do not contribute to the group cannot accurately rate their peers.
- student comments are moderated both automatically by the software and by the course tutor
- different students will be more generous or harsher when assessing their team, the IPAC software applies bias correction

weixin: scs_ryan

Requests to change coursework submission dates

Coursework submission dates are set centrally in Computer Science and not by the course tutor.

These have been planned carefully to fit with the teaching schedule.

Requests to change these for all students must be made to thecourse tutor who will then forward the request to the relevant authority in the computer science department. Requests must be supported with appropriate evidence of the need for change.

SoRA and EC

SORA: alternative formats

If you need the assessment or any teaching materials in an alternative format than has been provided, please contact the course tutor directly sarah.sanders@ucl.ac.uk.

If you need/prefer to submit the assessment in a way that minimises use of the written word, audio recordings instead of markdown text are accepted. Please don't use video, the files are too large, and you are more likely to be recognised (marking should be anonymous).

If you choose to use markdown and are concerned with spelling and grammar, there are spell checkers available for VSCode and included in PyCharm. The mark scheme does not focus on spelling and grammar.

SORA and EC implications for groups

Where a group member has an approved statement of reasonable adjustment (SoRA) or extenuating circumstances (EC) that allows for 1-2 weeks delay to submission, then the approved revised submission date will apply to the whole group. This is because it is not possible to separate an individual's contribution from the group's submission.

In the rare situation where a group member has an approved EC that extends beyond 1-2 weeks, e.g. into LSA, then the impact on the group will need to be considered on a case by case basis. It is more likely in this event that the student concerned would need to submit a new individual assessment and the group would continue to the original submission date.

Late submission penalties

Late submission rules apply to all assessments. Any penalties for late submission are applied by the computer science teaching and learning team when marks are entered in portico. The mark you see in the Moodle gradebook, therefore, is the mark before moderation or any penalty has been applied.

weixin: scs_ryan

If you have an approved extension resulting from a SoRA or EC, these will be carefully checked by the teaching and learning team before marks are entered in portico. However, the deadlines in Moodle may not be modified for individuals, so it may appear on Moodle that your submission is late even though you have an approved extension to cover the period.

Referencing

Book, journal, web article etc.

For book, journal, and web article references, follow the referencing style you use in your own department (e.g. Harvard, APA). Different departments in UCL use different styles. Include these in a 'References' section in your markdown.

Code

When you copy code from an external source, whether you are copying a snippet of code or an entire module, you must credit the source. This includes where you copy and then adapt the code. External sources include:

- user forums e.g. stack overflow
- open source repositories e.g. GitHub repos where an explicit open source license is stated
- communities associated with a library e.g. plotly forum

- course teaching materials
- official library documentation and tutorials of the python libraries and frameworks

It is not appropriate to copy from other students on the course, past or present.

For COMP0034 and COMP0035 you do not need to cite code from the course teaching materials or boilerplate code from the official library documentation of the required python libraries and frameworks that are expected to be used in the coursework (used in the courses e.g. pandas, flask, dash, bootstrap).

UCL has no single method for referencing code. For the purposes of this coursework use the following.

- Give the author, URL and the date of retrieval. If you adapted the code, indicate "Adapted from" or "Based on".
- Include citations within your code using code comments close to the affected code.

An example:

```
def play_sound(button_text):
    # Adapted from code from 'remdog' on the Plotly Community Forum at
    # https://community.plotly.com/t/linking-scatter-plot-elements-to-audio-
files/11908/6
    # Accessed 01/02/21
    ...sound code here
```

weixin: scs_ryan

Academic terms

It is assumed that you are familiar with essay terms such as 'describe', 'explain', 'discuss' and 'justify'.

There are numerous sources that provide guidance on these and academic writing in general:

- Essay question words
- UCL Guidance for academic writing
- UCL students union academic writing guide