

# **MSc Practical Software Development**

## **Coursework Assessment Handout**

Steven Carlysle-Davies, Alistair Grant, Bianca Prodan

Academic Year 2022/23

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b><i>Introduction</i></b>                                  | <b>3</b>  |
| 1.1      | Your project  | 3         |
| 1.2      | Your client   | 3         |
| 1.3      | Your team   | 4         |
| 1.4      | Your assignments  | 4         |
| 1.4.1    | Submission format   | 4         |
| 1.4.2    | Marking   | 5         |
| 1.4.3    | Peer Assessment   | 6         |
| 1.5      | Your contacts   | 7         |
|          | Good Scholarly Practice                                     | 7         |
| <b>2</b> | <b><i>Project Description</i></b>                           | <b>8</b>  |
| <b>3</b> | <b><i>Resources</i></b>                                     | <b>10</b> |
| 3.1      | Gitlab Project  | 10        |
| 3.2      | Prototype Code and Dataset                                  | 10        |
| 3.3      | EIDF Virtual Machine  | 10        |
| 3.3.1    | Accessing EIDF  | 11        |
| <b>4</b> | <b><i>Assessments</i></b>                                   | <b>12</b> |
| 4.1      | Formative Group Assessment: Prototype review                | 12        |
| 4.1.1    | Marking   | 12        |
| 4.2      | Summative Group Assessment 1: Design and Planning           | 13        |
| 4.2.1    | Marking   | 13        |
| 4.3      | Summative Group Assessment 2: Risks and Tests               | 14        |
| 4.3.1    | Marking   | 14        |
| 4.4      | Summative Group Assessment 3: Implementation and Evaluation | 15        |
| 4.4.1    | Marking   | 15        |
| 4.5      | Summative Group Presentation                                | 16        |
| 4.5.1    | Submission Format   | 16        |
| 4.5.2    | Marking   | 16        |
| 4.6      | Summative Personal Coursework                               | 17        |
| 4.6.1    | Submission format   | 17        |
| 4.6.2    | Marking   | 18        |

# 1 Introduction

## 1.1 Your project

Software development encompasses more than just coding. To name just a few elements, it involves gathering requirements, design, analysis, planning, identifying risks and communication. In this course, you are given a small software product development project that is split into multiple individual and group assessments spread over the duration of both semesters.

Your group assessment goal is **not** to build a complete solution to all the issues in the problem description. There are only twenty weeks of the course to operate in and you will have other course workloads to manage.

The goal for the assessment as a whole course is to gather requirements, design a system and build a prototype for a subset of the design to prove that the concept of your design works. This will be broken into different milestones, each forming a submission to be marked by the course assessment team.

Your initial design must cover an end-to-end solution, but your prototype is only expected to focus on one or more key element(s) of this design. This could be the ingest pipeline, data visualisation, or something else you identify.

**Please note that the goal of this assessment is to implement a prototype of the concept you have prioritised, and that you will be expected to be able to explain why your chosen element (or elements) has been implemented.**

Your solution must include major elements of a software project, including (but not limited to):

- Design specifications.
- Different types and levels of testing.
- Interaction with your client(s).
- An assessment of the quality of your prototype and project team.

When undertaking this project, you are expected to:

- Follow good development practices.
- Justify and explain the decisions you make as part of the assessments.
- Show ability to work individually as well as part of a team (i.e., team coordination).
- Show ability to communicate at both a technical level (i.e., with other software developers), and non-technical level (i.e., with clients).

## 1.2 Your client

The course assessment team will be acting as the clients for the purposes of this project. We strongly encourage you to e-mail the course assessment team if you have any questions or areas of confusion, or want to discuss the details of the product requirements.

## 1.3 Your team

This course will be 80% group assessed and 20% individually assessed. Group assessment sections will include peer assessment modifiers to the group mark.

The groups will be randomly assigned by the course administration and assessment team. Groups will be between 3 and 4 people depending on the total size of the course cohort.

Groups will be assigned at the start of Week 1.

Groups are expected to determine how they will coordinate and communicate – there is no prescribed method set for this as it depends on the individuals and the dynamic between them. Note that on-campus students can utilise face to face meetings, but online collaboration and meetings are feasible to work with as well.

Any internal group problems should initially be handled within the group. If this proves to be insufficient, the problem should be escalated to the course and student support team. This may result in a meeting with members of the course team and the project team. This is an opportunity to have issues resolved.

**Please note that when peer assessment is being reviewed, any major issues referenced in feedback that have not previously been raised to the course team will result in the peer assessment being disregarded for that group.**

## 1.4 Your assignments

This course has 6 assignments:

- An initial formative group assessment
- Three summative group assessments acting as milestones for one unified implementation of a solution to the client problem
- A summative group presentation
- An individual summative reflection and project review

### 1.4.1 Submission format

Content for each assessment submission will be the GitLab project your group has been provided with. This project will be the canonical source marked in the assessment.

The submission content should be in the repository, issue log and/or wiki of the project. All submissions as part of the group work should be in wiki/markdown format, or in a formal programming language.

Each group assignment must be submitted by **each member** of the group to the Assignment Submission tool (Turnitin).

Each member of the group should submit a text file of the following format to Turnitin:

---START FILE---

Practical Software Development Assignment: ASSIGNMENT\_NUMBER

Submission Owner: YOUR\_NAME : YOUR\_STUDENT\_NUMBER

Group Members:

- GROUP\_MEMBER : NUMBER  
 - GROUP\_MEMBER : NUMBER  
 - GROUP\_MEMBER : NUMBER  
 Gitlab Project Link: GITLAB\_PROJECT\_LINK  
 Commit\_ID: GITLAB\_COMMIT\_ID  
 Virtual Machine: VM\_NAME : VM\_IP  
 ---END FILE---

Example:

Practical Software Development Assignment: 1  
 Submission Owner: Yaxkin Christopher Stevenson : s3403354  
 Group Members:  
 - Tirta Yuraq Villaverde : s3454434  
 - Bertie Jinan Rhodes : s3543128  
 - Abiola Yona Nicolson : s3516097  
 Gitlab Project Link: <https://git.ecdf.ed.ac.uk/psd2223/project/>  
 Commit\_ID: 8d32c0f0407f5944178340f4c65ac25f0a7a4b8c  
 Virtual Machine: eidf018-assessment : 10.24.0.1

**Please note that work outside this repository, unless confirmed with the course team, is not considered as contributory evidence.**

### 1.4.2 Marking

Each assignment will be marked out of 100 and then weighted as part of the final course mark out of 100. The assignment weightings are the following:

| Assignment                           | Weight out of the total course mark |
|--------------------------------------|-------------------------------------|
| Formative Group Assessment & WebPA   | NA                                  |
| Summative Group Assessment 1 & WebPA | 25%                                 |
| Summative Group Assessment 2 & WebPA | 20%                                 |
| Summative Group Assessment 3 & WebPA | 25%                                 |
| Summative Group Presentation         | 10%                                 |
| Summative Personal Coursework        | 20%                                 |

Marking and grading is carried out in accordance with the University of Edinburgh Extended Common Marking Scheme CMS4 (<https://www.ed.ac.uk/timetabling-examinations/exams/regulations/common-marking-scheme>).

| Mark (%) | Grade | Description  |
|----------|-------|--|
| 90-100   | A1    | An excellent performance, satisfactory for a distinction |
| 80-89    | A2    | An excellent performance, satisfactory for a distinction |
| 70-79    | A3    | An excellent performance, satisfactory for a distinction |
| 60-69    | B     | A very good performance                                  |
| 50-59    | C     | A good performance, satisfactory for a masters degree    |

|       |   |  |
|-------|---|--|
| 40-49 | D | A satisfactory performance for the diploma and certificate,<br>but inadequate for a masters degree |
| 30-39 | E | Marginal Fail  |
| 20-29 | F | Clear Fail   |
| 10-19 | G | Bad Fail   |
| 0-9   | H | Bad Fail   |

### 1.4.3 Peer Assessment

Part of working in a team is the ability to provide feedback for each member of the team.

For each of the group submissions, a peer assessment of all members of a group must be submitted by each individual in the group.

This peer assessment will involve rating each member of the group on four key questions. Importantly, you must rate yourself in this peer assessment.

The questions will all ask you to give a person a rating from 0 (zero) to 5 (five).

#### **Important – please read the following:**

A rating of 3 (three) is considered satisfactory. 3 (three) is the default rating for someone who carries out their tasks and cooperates within the bounds of expected behaviour.

Ratings above 3 indicate a team member contributed significantly more/significantly higher quality work/acted in a significantly more professional or collaborative way.

Ratings below 3 indicate a team member contributed significantly less/significantly lower quality work/acted in a significantly less professional or collaborative way.

Ratings of 1 or 0 will not be taken at face value unless problems within the group were raised with the course team at least 5 days before submission and can be demonstrated with evidence.

The peer assessment will influence the group mark for each individual.

**Unsubstantiated ratings or coordinated efforts to negatively affect or manipulate this process will be treated as academic misconduct.**

## **1.5 Your contacts**

The following points of contact are available:

| <b>Purpose</b>                               | <b>Contact</b>                     |
|--|------------------------------------|
| General questions about software development | Course Organisers, Course Tutors   |
| Client questions for assessment purposes     | Course Organisers                  |
| Team issues and escalation                   | Course Organisers, Student Support |
| Other  | Course Organisers, Course Tutors   |

## **Good Scholarly Practice**

Please remember the good scholarly practice requirements of the University regarding work for credit. You can find guidance at the School page:

<https://web.inf.ed.ac.uk/infweb/admin/policies/academic-misconduct>

This also has links to the relevant University pages.

## 2 Project Description

The client would like a system for the analysis of academic papers and other sources (e.g., news articles). The analysis options would encompass (but not be limited to):

- Authorship
- Sentiment
- Theme
- Topic

The solution should allow researchers and/or analysts to ingest academic and other source articles into a data store for running analysis queries targeting the information and metadata collated about the papers. Some examples of analysis queries could include (but not be limited to):

- Clusters of papers around topics
- Links between authors
- Links between topics
- Links between authors and topics
- Sentiment analysis
- Paper outcomes
- Retractions
- Paper reference networks
- Timeline

**Please note the analysis is expected to include identifying how papers and/or articles are linked beyond just identifying common authors or referencing.**

The solution should provide a method to enable different visualisations of the results, from textual to graphical, supporting interactive analysis and near-real-time query processing on the data store.

The solution components should support (but not limited to):

- Data ingest from different sources.
- Data storing in raw and analytics-ready form.
- Query layer.
- Visualisation system supporting multiple visualisation types.

Components which could be introduced to support additional analysis could be (but not limited to):

- Additional processing pipeline for the identification of common technologies introduced or employed.
- Additional processing pipeline for the identification of common models used across a field.
- Functionality allowing individual analysts to add curated metadata or data about papers in the data store for themselves and others to use.



The solution would need to be a facility to publicise analysis outputs, share common analysis pipelines, and to submit updates or new data.

Additionally, the solution should consider ways of monitoring data usage, including query patterns and pipeline workloads, to be used for optimisation of the data store and different query types.

## 3 Resources

### 3.1 Gitlab Project

Each group will have access to a Gitlab project hosted on the University of Edinburgh ECDF Gitlab service. Access to your group project will be provided to you by the course organisation team.

Project Access URL will be of the form:

<https://git.ecdf.ed.ac.uk/psd2223/<group>>

Note: The owner of the project will be a member of EPCC – do not try to restrict staff access to the project otherwise your work will not be able to be assessed.

### 3.2 Prototype Code and Dataset

A repository containing prototype code, sample dataset and basic instructions has been provided.

Each group should clone this repository and push any changes or updates they want to make to their own repository from 3.1.

The repository is located at:

[\*\*https://git.ecdf.ed.ac.uk/psd2223/assessment\\_source\*\*](https://git.ecdf.ed.ac.uk/psd2223/assessment_source)

This repository contains a basic README and dataset along with code to evaluate in the formative assessment.

### 3.3 EIDF Virtual Machine

Each group will be provided with a Virtual Machine accessible via SSH and VDI on the EIDF platform.

This virtual machine will be a development machine for your group to use and for hosting any prototype solutions that are created. These VMs will be left in place for the whole course and can be used to show running code.

Note: This is not a substitute for writing deployment instructions for your prototype. This is in addition – a facility to show a working prototype.

Note: Group VMs are not backed up – group code will be highly dependent on good use of version control in the event of a failure.

### 3.3.1 Accessing EIDF

All students should ensure that they have a SAFE account before attempting to access EIDF.

To access EIDF resources, follow the guide provided here:

<https://epcced.github.io/eidf-docs/access/project/>

You will need to request to join the project EIDF-018 (EPCC MSc Courses).

After you have had your access request granted, you will be given access to your groups virtual machine.

This can be accessed via SSH and VDI:

<https://epcced.github.io/eidf-docs/access/>

Note: Members of groups will have “sudo” permissions on their respective VMs. Groups must adhere to the University computing regulations agreed to when they matriculated as a student:

<https://www.ed.ac.uk/information-services/about/policies-and-regulations/computing-regulations>

## 4 Assessments

### 4.1 Formative Group Assessment: Prototype review

In this assessment you are given an early prototype of the wanted system and a collection of hand-picked papers acting as data source. Your group will need to investigate and redesign this prototype, keeping in mind that the prototype is not optimised, it has not been tested and is not guaranteed to process all elements correctly.

In this assessment your group are expected to:

- Analyse the provided code and produce a summary of its functionality, defects and dependencies.
- Carry out testing to analyse the performance of the code.
- From the analysis, suggest improvements to the code.

The prototype is implemented in Python. It is expected that you may not be familiar with Python. A valuable programming skill is being able to use, read and profile code that is written in a language you have not encountered before, learning about the language as you go. A selection of resources are as follows:

- Python's Python 3 tutorial, <https://docs.python.org/3/tutorial/index.html>
- w3schools Python tutorial, <https://www.w3schools.com/python/default.asp>
- DataCamp's learnpython.org, <https://www.learnpython.org/>
- Software Carpentry's Programming with Python, <https://swcarpentry.github.io/python-novice-inflammation/>

#### 4.1.1 Marking

**Please note that this assessment will NOT count towards your final grade numerically. This assessment is to allow the course team to give feedback and advice on how to approach your learning in the course and the subsequent summative assessments. This initial assessment will be to provide a starting point for knowledge that you will need to build upon for the rest of the course.**

You will receive formative feedback on the following:

- Performance experimental design and documentation.
- Results and analysis visualisation.
- Quality of analysis.
- Conceptual understanding of the problem and code analysis.
- Quality of presentation.

## 4.2 Summative Group Assessment 1: Design and Planning

For the first milestone assessment your team will need to investigate the problem and create a set of requirements or equivalent information for designing a solution. Two different level designs must be provided:

- A high-level design of the end-to-end solution.
- A detailed design of the component(s) you and your team prioritised for your prototype.

A key requirement is for design details to link back to their source requirements.

The design should be supplemented by language and technology choices, presented with supporting reasons and rejected alternatives (with reasons for rejection).

The design should include a data model based on the requirements for ingest and analysis. The submission should include a timeline for the project for the rest of the prototype development. This should include an initial set of tasks and estimates of effort and elapsed time for the project.

Please note that you can choose to build onto the prototype you reviewed in your Formative Group Assessment, or completely replace it with your own solution. For either approach, you can use the provided collection of hand-picked papers as a data source. If you wish to expand this, or add other data sources, please contact the Course Organisers to discuss potential legal and operational implications.

### 4.2.1 Marking

Each section is marked out of 100 and then weighted as part of the final mark out of 100. This assessment is 25% of the final course mark with modifications for peer assessment.

|   |     |
|---|-----|
| Requirements Engineering                | 25% |
| Design, Language and Technology Choices | 30% |
| Data Modelling                          | 20% |
| Timeline and Initial Task Planning      | 20% |
| Quality of Presentation                 | 5%  |

## 4.3 Summative Group Assessment 2: Risks and Tests

In the second milestone assessment your team is tasked with:

- Creation of a risk register and description of a risk management approach.
- Design and deployment of a CI/CD pipeline.
- Creation of a test plan including expected inputs/outputs (prototypes), failure management, failure communication, and result recording.
- Review, choice and justification of an off-the-shelf testing framework that can implement test plan.

Note that test plans may also include non-machine-based testing such as user acceptance testing and usability testing.

### 4.3.1 Marking

Each section is marked out of 100 and then weighted as part of the final mark out of 100. This assessment is 20% of the final course mark with modifications for peer assessment.

|                                      |     |
|--------------------------------------|-----|
| Risk Review                          | 25% |
| CI/CD Pipeline Design and Deployment | 20% |
| Test Plans and Prototypes            | 30% |
| Choice of Testing Framework          | 20% |
| Quality of Presentation              | 5%  |

## 4.4 Summative Group Assessment 3: Implementation and Evaluation

For the final milestone assessment your team must provide:

- A working prototype complete with setup scripts and documentation.
- Usability testing and analysis.
- Performance testing and analysis.
- Project and prototype evaluation.

While there is no enforced language or technology, the prototype should be submitted as a code base with setup scripts so that a tester can deploy the code, run a set of user tests and evaluate the product. The documentation should support the user through the first few actions after installation.

The usability testing could be done via pen/paper or mock-up interfaces or via an implementation. This should include how the test cohort was created, how it deviates from an ideal cohort and what the results of the testing mean.

The performance testing should examine elements of the prototype for how well it performs given different conditions and report back on this with limitations of the code base and suggestions of improvement.

The project and prototype evaluation should summarise and critique the whole group process (e.g., did you meet your plan, did you deviate, why?). This should include how the requirements were met, what effort was expended and how the team and prototype performed and what lessons can be learned.

### 4.4.1 Marking

Each section is marked out of 100 and then weighted as part of the final mark out of 100. This assessment is 25% of the final course mark with modifications for peer assessment.

|  |     |
|--|-----|
| Final Prototype (code, setup scripts, documentation) | 25% |
| Usability Testing and Analysis                       | 25% |
| Performance Testing and Analysis                     | 20% |
| Project and Prototype Evaluation                     | 25% |
| Quality of Presentation                              | 5%  |

## 4.5 Summative Group Presentation

In this assignment, your group will present your design and prototype in progress to the client in a presentation. The presentation should highlight the technical merits of your design, how the prototype is implementing the prioritised concept and how this matches the collected requirements. The presentation should conclude with a summary outline of future plans for the prototype product.

The presentation should include:

- Product status and demonstration.
- Technical analysis.
- Planning for future development.
- How the current prototype meets a subset of the gathered requirements.

Presentations will take place in front of the course assessment team and other audience members. This will be a verbal presentation, individuals unable to attend will be able to contribute in a hybrid online environment where necessary.

Presentations will be 15 minutes long, 10 minutes of presentation and 5 minutes of questions.

### 4.5.1 Submission Format

The slides (PDF or PPTX) and any videos or live URLs should be added to the repository.

The submission to Turnitin will follow the format in 1.4.1

### 4.5.2 Marking

Each section is marked out of 100 and then weighted as part of the final mark out of 100. This assessment is 10% of the final course mark with modifications for peer assessment.

|  |     |
|--|-----|
| Product Status and Demonstration                             | 15% |
| Technical Analysis   | 25% |
| Planning for Future Development                              | 15% |
| How the current prototype meets a subset of the requirements | 15% |
| Quality of Presentation                                      | 15% |
| Questions  | 15% |



## 4.6 Summative Personal Coursework

The personal coursework is split into two sections:

1. Personal reflection (50%)
2. Code review (50%)

### Personal reflection

A personal reflection on the group learning project.

This should include a summary in their own words of what the project intended to achieve, what it did achieve, what the individual learned during the process and a summary set of learning points moving forward.

The reflection piece should include:

- assessment of the group performance and dynamics
- assessment of the individual and their role in the assessment
- a measure of the project against its intended outcomes
- a summary of the concepts and learning that they have undertaken during the process
- a set of learning actions

### Code review

Review a code module from the project. The course assessment team will not provide –this module, and the group work must be the source of the code module to review. Advice can be sought from the course assessment team as to the suitability of a code module for this task.

This review should include the following:

- adherence to acceptable project code standards
- functionality review comparing to requirements
- readability
- testing
- performance
- potential improvements
- existing issues

The code review should include a summary of the tools and processes undertaken for the review, a summary of the overall state and purpose of the code module and a recommendation summary.

#### 4.6.1 Submission format

Since this is an individual assessment, each team member is expected to upload their own report directly to the assignment submission rather than via the repository. The report should be no more than 10 pages long (approx. 3500 words), Size 12 Text, 1.5 line spacing. - excluding title pages, contents, glossary, and appendices. All statements, quotations and evidence provided must be referenced correctly. You may wish to refer to information given during your Project Preparation course for more information.

The report will be a submission template provided by the course assessment team. The submission format will be a PDF document.

#### **4.6.2 Marking**

Each section is marked out of 100 and then weighted as part of the final mark out of 100. This assessment is 20% of the final course mark with modifications for peer assessment.

##### **Personal reflection (50% of the assignment total)**

|   |     |
|---|-----|
| Assessment of the group performance and dynamics                              | 30% |
| Assessment of your individual role in the assessment                          | 30% |
| A measure of the project against its intended outcomes                        | 20% |
| A summary of the concepts and learning you have undertaken during the process | 10% |
| A set of learning actions   | 10% |

##### **Code review (50% of the assignment total)**

|  |     |
|--|-----|
| Adherence to acceptable project code standards | 20% |
| Functionality review comparing to requirements | 25% |
| Readability                                    | 10% |
| Testing  | 15% |
| Performance                                    | 10% |
| Potential improvements                         | 10% |
| Existing Issues                                | 10% |