2.3 Coursework II: The Prototype

Once an Engineering Design Review document has been reviewed by all interested parties, improved by the author based on feedback received, and management has decided to allocate engineering time for the proposed work, the engineer(s) assigned to the work can get started.

2.3.1 Task

Form a group with 3-4 members, ensuring that everyone in the group wrote an EDR in the context of the same organisation for the first coursework. Then choose one of your team's EDRs from the first coursework and implement a prototype for it. The emphasis here is solely on demonstrating that the design proposed in the EDR is viable and you should only build what is necessary for that purpose.

You may use any programming language, libraries, tools, etc. that you want for this. However, you should make use of software development techniques that have been covered in the module. This should include version control, appropriate workflows for a team, testing, build tools, dependencies on existing libraries where useful, continuous integration, etc. You can read the assessment criteria below for more information on what is expected.

The design proposed in your EDR forms part of a larger system, which does not really exist, and you are not expected to build all the systems your design depends on. Instead, you should "simulate" other parts of the overall system to the extent that your design depends on in whatever way is most convenient. This may be accomplished by, for example, hard-coding values received from other systems or parts of the same system, building extremely simple versions of other systems that just return test data, randomly generating data, replacing networking with direct user input, terminal output instead of network responses, and so on. If you are not sure how to simulate a part of the overall system that you need, feel free to ask for suggestions!

2.3.2 Working as a group

Every group of engineers works differently and different engineers bring different qualities. In a real workplace, some engineers will be more experienced and work faster than others. Regardless, everyone has their unique strengths and weaknesses. In your group, make the most of each others' strengths and work around your weaknesses. If you are in a group with someone who you feel is not contributing as much as others, try to identify what strengths they could tap into. Alternatively, it is common for more experienced engineers to mentor less experiences ones. Figuring out how to make your group work effectively and help each other is as much part of the coursework as the technical aspects.

2.3.3 Submission

Your submission should include an archive of your code repository, including its full history. It should also include a short report that briefly summarises (no more than 200 words) what was accomplished and discusses in detail how it was accomplished (no more than 800 words). This report may include any supporting material that demonstrates the software engineering techniques you employed that is not already included as part of the archive of your code repository. For example, screenshots of key issues demonstrating how you planned and provided updates on work.

Furthermore, each member of the group should *individually* write around half a page of *reflections* on every other group member. The reflections should be *constructive* in nature and you should be happy to share them with the individual they are about. Assume that, rather than being students at a university, you are colleagues at a workplace and that the end of the module or group project is not the end of your working relationship. What would you suggest to your "colleague" so that working with them will continue to be productive in the future or become productive? You can read more about what reflections are in the specification for the third coursework.