PRCO304: The Project Proposal

Information about the Proposal and PID processes are to be found in the ProposalPID folder on the PRCO304 DLE site:

https://dle.plymouth.ac.uk/mod/folder/view.php?id=269659

The first thing you need to do for PRCO304 is to identify an idea for your project and write it down in a *proposal* [1]. This proposal forces you to start to think through the details of your project, and is used to help the module leader allocate a suitable supervisor: you will not be allocated a supervisor until you have submitted an acceptable proposal.

Please note that your PRCO304 is to be an individual project:

- group projects are not allowed
- multiple students are not allowed to work on the same project
- projects that are dependent upon the work of other students are not allowed

Please also note that dissertations (i.e., extended essays) are not allowed.

Please note that everything that appears in your project proposal will no doubt get re-used in part of your Project Initiation Document (PID) and Final Report --- so none of the words that you write here are wasted effort.

Submitting your proposal

- Proposals will eventually be submitted via the Student Project Management System (SPMS) by copying and pasting into a web form, so please don't spend a large amount of time formatting your proposal.
- You will not be able to access the SPMS until your PRCO304 registration is activated … this should happen at the start of semester.
- Project supervisor allocation starts 9 November, and project proposals will not be processed until then. So although we wish you to make progress in developing your proposal, it is important to do this work properly rather than quickly.

[1] As a rough guide, a proposal might constitute 400 words

Developing your proposal

Your idea may be based on study, placement experience, career aspirations, a desire to learn new ideas or technologies, personal interest and/or a suggestion from staff, but must be relevant to your programme of study. **If you are in doubt about a potential project idea, discuss with your programme manager.** Given the very large number of students taking this module, please do not email your proposal to the module leader.

Information relating to previous topic choices are to be found in various folders on the PRCO304 DLE site namely: PreviousYearsShowcases; SomeRecentProjectReports; PreviousProjects.

As you will see, the vast majority of students undertake a software development project; we do allow projects that deviate from this model, but dissertations (extended essays) are not allowed. The key things to identify in your project proposal are:

- 2 or 3 key-word phrases[2] which encapsulate the nature of your project
- The intended client (if you have one)
- The main product (final deliverable) to be produced
    - and why your client needs it (if you have a client)
- Initial brief description of the final deliverable (e.g., web-based system providing the client with the ability to do XYZ)
- Method of approach
    - E.g., Likely software development process (e.g., waterfall, …)
    - E.g., Likely development platform (e.g., PHP & MySQL)
- Requirements for hardware, software (please note anything that is not currently available and for which you would require financial support) or other resources (e.g., external company support)

Please note that the information that you provide in this proposal is **not** to be seen as set in stone … certainly these features of your project may well subsequently change and evolve.

You may also wish to consider

- what learning you will have to undertake to carry out the project
- risk factors: what could go wrong and how likely is it?

[2] For example: "Web application development", "Software development", "Computer security", "Computer networking", "Systems analysis and design", … etc.

Project scope

After you've had your proposal approved, you'll be asked to submit your Project Initiation Document (PID) – in which you'll need to detail your project scope. So although scope does not need documenting in the proposal, you do need to give it some thought now.

It is important that your proposed project is not too thin – otherwise you are destined to fail. Examples that (probably) fall into this category are static web-sites or stand-alone databases (particularly MS Access). Obviously – also – we are expecting a step-up from anything that you may have done in Stages 1 or 2.

It is also important that your proposed project does not rely on a single, challenging & indivisible deliverable … problems can then leave you with absolutely nothing. So the advice is:

- aim to be relatively ambitious
- ensure that the project can be divided into
    - a core (which you are confident of delivering, and which is in some sense a self-contained whole; for example if you are undertaking software development, then the core deliverable should yield a software system that is worth deploying[3])
    - additional desirable elements that you'll undertake if you have time

These issues of scope will be considered by your supervisor in deciding whether (or not) to approve your project/PID.

[3] Consider for example a hotel booking system: some features (e.g., the ability to make a reservation) are utterly core … it simply would not be worthwhile trying to deploy a hotel booking system that did not have this feature.

Further comments on software development projects and scope

In a software development project there is a variety of aspects that you could gain credit for:

- Interaction with a real client
    - A real client can be a real benefit to a student project … it gives you an opportunity to do many things (e.g., requirements elicitation) "for real", and the client will generally drive up quality. That said, don't become overly dependent upon a real client – their priorities can change, so consider what would happen if they dropped out midway through.
- Feasibility
- Requirements elicitation, prototyping
- Requirements analysis, specification & validation (including non-functional)
- Design (architecture, DB, functionality, interface) … commonly considered too thinly
- Coding; controlled re-use
- Testing, V&V: essential
- User evaluation, HCI/usability
- Deployment to the client
- Development process; novel approaches and technologies
- Quality management
- Security
- Project management: essential
- Consideration of legal, social, ethical and professional issues: essential
- Project reporting and post-mortem: essential

In most software development projects the central aim is to deliver a deployable software system. In these cases, it is advisable to make the project as broad as possible, bringing in the various elements (as above) that would typically be found in a real software development project.

Some projects do however differ, and (for whatever reason) focus much more strongly on *some* of the elements of the software lifecycle. For example a project might focus on requirements and design, but only involve the production of a minimal prototype.

The balance between the above elements is up to you to decide: use your common sense, be clear where you will gain credit, and make sure that this balance is reflected in your project objectives.