This document is a record of the discussions in the project supervision workshop on 18/10/2016. The workshop discussed ways of improving supervision for undergraduate and Masters' individual projects.

It is not a set of recommendations, and it does not set or suggest any new policies.

There is no right way to supervise a project

We do not have and will not make extensive rules on project supervision. One of the major benefits of our system is flexible, adaptive, personalised approach to project supervision.

But the School does have enormous implicit expertise. The purpose of this workshop was to discuss ideas that could make project supervision better for both students and supervisors. This record is made available to share that expertise further.

Topic I: Are there new ideas that could support our supervision?

Written records

There were several discussions about the value of consistent written records throughout the project, including:

Having students send formal agendas for each meeting (in advance);

Having students keep a dated log book of the work they produce each week/day;

Having students keep minutes of each meeting (and sharing them with the supervisor). This allows immediate insight into whether students are actually taking in the content of discussions in meetings.

Having supervisors keep minutes of each meeting (perhaps complementing the student's minutes -- differences between the minutes may be revealing!)

This was seen as probably the easiest way of getting a better handle on project supervision.

Software tools

There are a number of software/online tools supervisors use to help support students.
Slack to facilitate discussions with project students;
Trello for helping keep on top of student tasks;
Overleaf for commenting/tracking LaTeX documents;
Github or similar public source-control repositories to keep a record of work that supervisors can track;
Google Docs to support low-effort creation of minutes and plans (e.g. on a weekly basis)
All students should be using some form of version control (probably Git, but up to the student/supervisor)

Setting expectations

Making clear to students at the start -- what is a good project? It is not about writing thousands of lines of code or 120 page reports.
The first five weeks of a project are critical. Students have a tendency to squander them -- but that perhaps is because we don't make clear to them the value of this time. This is the only time when students are relatively free of coursework constraints.
Make exemplars available to students. Make clear the structure and expectations about dissertations at the very start.
Run through the marking scheme with students in the first meeting, and if necessary, on every subsequent meeting.

Make sure students have browsed projects in the Hall of Fame and know what a good project looks like.
If a project will be very different for some reason, make to students clear how the structure of their dissertation will differ from prior excellent dissertations.

Concrete images of project commitment

Projects should be ~400 hours of work. Make clear the commitment a project involves to students. *This is about 9 hours a week on a 23 week semester model.*
Alternatively, this is about £6000 of work (at a very cheap programmer rate!) -- consider asking students “is your project really worth £6K”?

Measuring progress

Projects are easier to manage if supervisors can ensure that there is continuous *measurable* progress each week. Helping students make their progress measurable is something that can really support their progress.
Make clear reasonable expectations of progress each week.
Try to get students to commit to writing early.
Consider having students create a complete dissertation skeleton document on the very first meeting, and flesh it out throughout the project.

Other issues

Student Learning Service can provide support for students struggling with language issues. This can let you focus on the academic/CS issues.

Importance of pastoral care. Supervisors have responsibility to look after students beyond doing a good project. Care must be taken to avoid students going AWOL. Warning signs, like missing consecutive meetings should be taken seriously and acted on.

The presentation sessions are awkward, with very small audiences: we as a School need to find a way to improve this (without having staff sit through dozens of hours of presentations).

An important part of a project is teaching students how to be professional. We should be careful not indulge unprofessional behaviour.

There were a number of discussions on how to present project work of disparate types in the rather narrow project dissertation structure

- How does a research project fit into the “product” category, for example?
- How should students reporting on iterative-design project, where there isn’t a standard waterfall development process?

The ability for supervisors to tweak the marking scheme weightings should be made active, and supervisors made aware of the possibility to use this to compensate for different project styles and structures.

Topic II: How could supervision meetings be improved?

Group meetings.

This can work, but personality issues can lead to dominance problems -- 6 students in two hours might mean 20 minutes each; or 110 minutes for one student and 2 minutes each for the remainder.
Shy students may be penalised by group meetings.
On the other hand, group meetings can provide mutual support and also mutual motivation for the students. Seeing others' progress can set a common benchmark.
Whether this works will be highly dependent on the supervisor, students and the specific projects.

Missed meetings

Watch out for this, and avoid students going AWOL. An occasional missed/postponed meeting is not a problem, but consecutive missed meetings indicate something could be going very wrong. Supervisors should propagate warnings back to the year head.

Duration of meetings

Meetings almost never require an hour, and an hour per student is probably impractical at our current class sizes.
Consider half-hour slots.
Packing students together (e.g. 2 hour block with 4x30 minute meetings) can be an efficient way to schedule time, if compatible with student timetables.
Avoid filling up time; meetings are best if short and to the point.
Students who need additional time should be supported, but only within reason.
Maybe an extreme model with 2x10 minutes meetings each week would provide more value at less time invested.

Content

Consider testing/run student code live in the meeting (e.g. clone from GitHub, compile and run it). This is a quick test if the project is really going anywhere and if any implementation progress is being made week-to-week!

Students have a natural tendency fixate on specific, immediate problems. It may be part of the supervisor's role to consistently keep the big picture in view, and contextualise what may be rather minor details.

Meetings have different structure throughout the runtime of the project; it may be useful to adapt

the structure of meetings for the different phases of the project (discussion, planning, implementation, dissertation writing, etc.)

There are always assessed exercise time conflicts. Be prepared for students to be unable to make progress, especially in weeks 7-10 or immediately after hackathons.

Meetings arguably should be businesslike and professional. Part of the goal of the project is arguably to teach students how to make use of meeting time effectively, and to be effective participants in business meetings.
Consider the three P's model for meetings: Progress – problems – plans

Proposal: Project assessment workshop

A follow-on workshop could be held to discuss the marking and assessment of projects and how this can be streamlined and more consistently calibrated.

Appendix: Roles

The following is list of potential roles that a supervisor might takes on from the perspective of a student, and the roles that a student might take on as a supervisor. It is not definitive or comprehensive. It was presented at the workshop as a stimulus to provoke thought about what project supervision means.

Student roles, as seen by supervisors

Burden: who wastes time and effort.
Learner: who picks up skills from the supervisor
Explorer: who explores questions supervisors don't have time to investigate
Worker: who completes tasks, possibly for a wider project
Teacher: who back propagates up-to-date information on new technologies.
Glue: who supports or establishes a collaboration with a company or research group
Prospect: who is "groomed" for future study (e.g. a PhD)

Supervisor roles, as seen by students

Technical expert: who solves technical challenge to allow the project to progress.
Client: who sets the requirements and judges the progress in terms of them.
Assessor: who judges the quality of the work and grades it.
Guide: who knows the ropes and helps to foresee and avoid problems.
Boss: who instructs on the actions to be taken.
Voice of reason: who unmuddles confused thought
Pastor: who looks out for students' welfare