This report contains feedback from students about a course taught in the School of Informatics during the 2020/21 academic year, in response to the following questions:

- What advice would you give to a student taking this course in future?
- What did you find most valuable about the course?
- What improvements, if any, would you make to the course?
- Please add any other comments you have about workshops, tutorials or labs on this course.
- Please add any other comments you have about the presentation of course materials online and their accessibility.
- Reflecting on your experience of hybrid teaching and learning on this course, what has worked well for you?
- Is there anything else you'd like to tell us about your experience of hybrid teaching and learning on this course that would help us improve our approach?

Each course organiser receives this report as well as statistics on multiple-choice responses. All these reports, together with student feedback about individual members of teaching staff, are collected and sent to the Director of Learning and Teaching.

Please note that these are personal responses from individual students: some courses only have a few responses and a small sample can be unrepresentative.

Stereotyping and bias, especially unconscious bias, is a serious concern in any survey gathering personal responses. All students received the rubric below before completing the surveys, and you can read a brief introduction to issues of unconscious bias on the university web pages at http://edin.ac/2iypZBv

This information is provided for students and staff at the University of Edinburgh: **you may not redistribute or reuse it without permission.** If you would like the information in another format or want to use it in your own publication then please contact the Informatics Teaching Organisation at http://www.inf.ed.ac.uk/teaching/contact

---

**Rubric given to all students taking the end-of-course feedback survey**

We value your opinions on the courses you take here at the University, as they allow us to shape future delivery and development. We welcome constructive comments about your courses, whether positive or negative, and ask you to give details about any issues in order to help the course organiser to understand and address them.

We encourage you to be aware of the potential for bias in the completion of these questionnaires, so we have developed resources which may be helpful to you:

- Equality, Diversity and Unconscious Bias ([http://edin.ac/2iypZBv](http://edin.ac/2iypZBv))

You also have a responsibility to provide feedback in a manner which does not breach the University’s Dignity and Respect Policy:

- University of Edinburgh Dignity and Respect Policy ([http://edin.ac/1Cq0VZY](http://edin.ac/1Cq0VZY))

The results of the questionnaires will never be analysed in a way that seeks to identify individual students from their responses. However, should you wish to remain anonymous, please do not identify yourself in your answers to the survey questionnaire implicitly or explicitly.
1. System Design Project

What did you find most valuable about the course?

- The ability to learn independently, without any particular framework or objective of what is to be learned.
- Collaborating with certain team members was enjoyable and represented a good change from the more isolated experiences of most courses.
- The ability to manage and co-ordinate the project was very valuable.

Groupwork skills

- Having the opportunity to work in a team to complete a larger-scale project.

- How the teams are put together - the fact that each one has several potential leaders, and that there are both people with extra experience and people with none makes the group work overall much easier. Also the skills learned from it.

- I did like that a lot of skills were touched upon, and I liked the fact that we had freedom, though it was rather intimidating.

- I found working with my fellow students and interacting to build a project with people with different expertise to be deeply rewarding.

- I learnt how a lot about working as a team which I had little previous experience of.
  I also got a lot of hands-on experience with lots of different things that I hadn't used much before (Git, Android development, asynchronous programming, sockets, concurrency).

- I learnt how I'm interested in UX/UI and working in a team. Other than this, I learnt how important how good course organisation is, by experiencing a badly organised course from a staff who are more interested in what they're interested in, rather than what's best for students.

- I learnt how to deal with
  - stress
  - deadlines
  - bad team mates
  - doing all of the work in the team
  - the pain of enduring a poorly organised course
  - the unfairness of flippant marking
  - the incompetence of terrible course organisers

- I learnt how to work in a team

- I love that we can work in teams to develop a robot. It is challenging and at the same time cool to work with other students to develop something together

- I struggle to find anything of value I have gotten out of this course, since most of what the course is intended to teach is the aspect of working in a group on a significant project, of which I already have a decent amount of experience in (2 industry internships and 2 years actively involved in a university student club).

- I've learnt more about team management and organisation.

- It encourages group work, collaboration, lateral thinking, and challenges the student to produce something interesting, unique, and involved using all they have learned thus far.

- Learning to work in a group, communication skills.

- Learning to work in a team was very valuable. Having a lot of creative freedom is great. Good to work on a larger scale project.

- Making people work in teams

- None. This course has worsened by university experience which I cannot say for any other course I have taken

- Our mentor for our group was great and helped us get through this course. Since, he took the course last year he was very knowledgeable about the running of the course and some of the challenges of doing it online.

- Project management, dealing with other people

- Teamwork

- Teamwork, I guess?

- The ability to learn in a team environment

- The course is the first big group project that we did in the university. We gain some group work experience based on this course.
The expert feedback and the workshops with guest speakers were really good. I also really like the 3 people teaching this course, Dr. Tonneau in particular is very good at explaining things!

The opportunity to work with a large team

This course was awful

This course was not valuable in the sense that it had no active teaching, and was simply an exercise in doing stuff that you have already done and teaching yourself.

Was really good meeting and working with other people.

Working in a team and learning new skills.

Working on a larger scale project

Working on a real project with other people is a good way to learn how to use standard resources, how to organise and communicate within a team

Working within a team and overcoming the challenges that come with it

Communication with your group mates and good cooperation.
What improvements, if any, would you make to the course?

- It would be useful if each member had to submit a mini-report/paragraph alongside the demos, detailing what they have contributed in the past X weeks. This would at least encourage more engagement with the project due to accountability.
- Feedback did not reflect sentiment of the client (and vice-versa).
- Even when the group read over and accounted for all negative comments in previous feedback reports, this didn’t seem to significantly alter the grade percentage.
- Video time limit and page limit are slightly too short, an extra 5 mins and one or two pages would allow for a better report, particularly for larger groups with lots of different contributions.
- More personalised workshops for Webots set up (like a one hour session for a group with an assigned TA to set up webots)
- More attempts to add more of a social element to the course (inter-group quiz), especially in the pandemic
- Staff have more flexibility
- No appreciation of the current global situation
- Outdated resources
- Mark system of demo provides little to no value
- give more guidelines or help or more time to come up with a project idea. Our group came up with like 2 ideas and used one - maybe this is because of lazy team mates but please make it a bigger deal to come up with ideas since the whole course rests upon this. My experience of the course would be improved if I actually was interested/inspired about the project I was doing.
- don't make robotics the focus during a pandemic - developing robots remotely is painfully hard, especially since we can't turn up and get help. instead we have to create a ticket and get a reply pointing us to a website we've already looked at a few days later.

1. Restructure the deadlines. Right now it’s a lot of things to be done in the first couple of weeks (which is understandable, and, in hindsight, helpful), then a lull in the middle when everyone is working, and then too many things to do in the last two weeks (last report + video + user guide + changed video + finalised website + creating the room in gather.town) were all due over the span of 4 days, which is TOO much work. And a lot of people can’t finish any of those tasks beforehand, because the whole system is only done then, unless some of the deadlines are changed (e.g. the last demo is a week earlier).
   From this point: a lot of things seem to be repeated over the semester - e.g. in project plan, three sections describe the exact same things; website is basically a repeat of all the demos and the user guide; the video is a repeat of each demo. A lot of effort goes into writing the same things again and again, which feels counterproductive and like a lot of unnecessary work
   2. Install ROS on DICE. It’s impossible to work on it on your own machine unless it runs Linux, and legitimately half of the time working on the project was spent trying to figure this out. I understand that part of the ‘learning’ is installing software, but this is something that a lot of groups used and a lot of groups had problems with.
   3. Have a more extensive workshop about simulations (W ebots, or even ROS, which would be amazing). Whatever documentation they have on their websites is not enough for the projects, and right now the only workshop that we had was just following a tutorial from the website (and we only used one feature from that workshop, while spending hours trying to find other necessary ones)
   4. Depending on the situation next year: if it's online, allow more informal group-to-group interactions. The bloopers chat was nice, but not really active, and felt like not enough to know what others were working on.

Although at the start of the course it was emphasised that hardware implementations would be viewed in context (i.e. with complication arising due to the pandemic) I rarely found this to be the case. Often feedback would be returned, focusing heavily on the failure to implement certain hardware elements although often this aspect was completely out of our control. Generally, I found the expectations for what would receive an A grade to be unreasonable, thus would suggest either (1) clarifying this or (2) lowering these expectations.

Better organization and more consistent marking throughout the course

Due to the universities common grading scheme there was very little motivation to put in extra effort - the grade we got after a demo where we were praised for the improvements we made was 1% better than the one where the lecturers questioned if we have done any work at all (we did!)
   Also, the courses expectations cannot be done within 200 hours of a 20 credit course. All of us worked at least double (some 4x), after which the N(60,3) grades are disappointing. Much of the time is spent learning the specific programs and devices (especially in this year with webots and other software), which is hard to show in demos. Either make it a 30-40 credit course, or lower the expectations

Have less demos (2, maybe 3). I feel like we spent SO much time on demo prep which we could’ve used for the development of our system if there hadn’t been that many demos.

Having fewer demos to allow the team to make more noticeable improvement between demos. Having more guidance for videos - in particular the pitch video. More guidance on working with a web application and Webots. Having less of the mark rely on the team’s contribution.

I guess the course is meant to teach us about system design and team management - but I feel like it’s only made me appreciate how stressful things can get with a lack of organisation. I think more involvement with each group in organising them and walking through who does what could be really helpful, at least for the first few weeks. No one in our group was really experienced or keen to be a team manager - some clear instructions and help with that could really have been helpful.

I think it’s a pretty insane workload - especially with the report writing all the time. I’ve been pretty stressed for other courses / coursework in the past, but this is the first time I think it’s really impacted my mental well-being (although I reckon the covid pandemic is also a factor). I feel like the number of demo reports could be reduced, or less could be expected for them, or more formal demo reports and 2 more informal feedback / live demo sessions? It could be such an enjoyable, interesting and useful course if it wasn't so stressful all the time.

I was really looking forward to this course before the start of this semester but I'm sorry to say that there weren't many moments of thus course I enjoyed. I firmly believe that this course should not be compulsory. Having this course compulsory leads to having a lot of students who aren't keen to take part in the groupwork. I tried my best to work with my teammates but it did feel like nobody wanted to be doing this course and made this a painful process.
In my opinion there is too much focus on the commercialisation aspect of the robot and little emphasis on the technical aspects. For most computer science students this is not enjoyable. I felt that feedback we received for our demo reports was unnecessarily harsh considering the current pandemic and often did not match what our clients said in the Q&A session. There appears to have been no changes to the marking scheme from last year and no adjustments for us not having physical access to the robots. Compared to other courses who were understanding with the coursework this semester our course was not flexible in the slightest. Having a demo every 2 weeks was stressful and the online working made it hard to get a lot of hardware work done before the next demo.

The industry presentation aspect of the course is something I'm dreading as I don't feel comfortable having some of my marks decided by people outside the course considering they don't have an understanding of the issues we have faced this semester.

- I would increase the number of credits allotted to the course, to more accurately reflect the workload it requires.
- I would try to lessen the workload in the first and last weeks. Also I would change the weighting of the project plan, 15% in the second week while people are still finding their footing was stressful. Also I would remove the in-group peer review of how much other team members worked, it feels slightly uncomfortable and like I'm snitching on my teammates.
- Increase the number of credits this is worth.
- Make it 80% individual work, 20% group work. This is by far the worst course I've taken in my 3 years here and it is entirely due to the fact that it's group work. That means I need to rely on other people to determine my grade, and therefore my degree classification.

Your degree classification is supposed to reflect your own merits and this course is the antithesis to that. I have to deal with the fact that because my group has 3 people that do virtually no work, my grade is directly harmed by factors outside of my control. I then have to hope that my grade gets scaled to reflect my efforts. It is completely unfair and quite frankly infuriating.

- Make the deadlines more separated.
- Make the marking better. We never know what to improve as the marking is not very consistent and sometimes there is contradiction on feedback between different demos. Also, we do not have access to real equipment but we always lose points on being not very realistic (difficult in Webots).
- Make the marking instructions clearer and more consistent.
- Make the marking better. We never know what to improve as the marking is not very consistent and sometimes there is contradiction on feedback between different demos. Also, we do not have access to real equipment but we always lose points on being not very realistic (difficult in Webots). Could have given access to robots. Even for connecting to Raspberry PI's it was a mess.
- Many things:
  - Rename the course and change the course: This isn't a "system design project", this is a "robotics practical project", system design in software engineering deals with scalability & reliability. I learnt this stuff while preparing for my interviews. This course is only useful for the small proportion who'll end up researching useless robots. Most students don't fall into this criteria.
  - Make a rubric for each group. Students are expected to write a project plan, let the client write a rubric based on it and mark us against it. Without this, our group marks just depend on how our client is feeling that way. Our group has been impacted by unconscious bias from [OUR CLIENT]. For one demo, [OUR CLIENT] was in a very happy mode one day and gave us more marks, while for the next one he was having a bad day and gave us less, while we did more work for the second one.
  - Let students choose the group. This is the main reason why student satisfaction goes down in Edinburgh. One person in the group ends up doing three people's job, and its just stressful. Why should the student who score a higher mark need to suffer from this?
- Many. First of all, this course was probably amongst the ones most affected by covid, since we couldn't work on our hardware like they used to do in previous years. I believe it would have been at least reasonable to allow students to work on a more software-oriented project because of this, but not! Instead, we were quite discouraged from working on anything that didn't involve (much) hardware even though we wouldn't be able to lay our hands on a single piece for the entire course! Instead of gaining what there is actually to be gained by working in a hardware-related project, we had to spend much of our precious time in making the simulation work, figuring out all of Webot's weird quirks and never knowing for sure how our product would work in the real world.

Also, I'm not an expert on simulations but I could find really little helpful material on Webots on the internet. Yes, there was a good amount of documentation but nearly zero forum discussions of common problems etc, which made debugging Webots-related issues REALLY frustrating.

One thing that was very disappointing was the marking. The client feedback system was nice, but the marks we received for our submissions always were ridiculously low. Even when we put substantial amounts of effort we didn't see the difference we expected compared to previous submissions, which was really disappointing for us. I would expect that an "excellent submission would receive marks ranging 85-100, whereas a "good but can be improved in a few ways" submission would be within the 70s-85s. Marks of 55-70 would be for a "fine" submission, etc. From my understanding, there are 30 marks available for "excellency". Now, "excellency" is a very strong word, and I understand that it's not to be taken lightly. In my opinion, deducting 30 marks because the submission was not "excellent" is simply, too, much. I haven't heard of any team getting a mark in the 80s (I might be mistaken), which means that either marking is too strict or all teams are underperforming... My team really struggled to get a 70+ grade for the last demo and we didn't manage. In the (bad) feedback that we received, it seemed that we were deducted points for "missing" things that were actually there in our submission. None of us was motivated enough to even appeal for this, because the course hasn't been anything but disappointing so far.

- Many:
  - SDP is by far the worst ROI course I've ever taken. SDP requires an inordinate amount of time and effort, for extremely little payoff. There is little to be "learned" in SDP, much of it is just tedious slugging through respective reports and demos.
  - For students who already have significant experience in industry and student clubs, SDP provides virtually no value: SDP is just a poor attempt at imitating what such an environment would be like and is a far cry from the true experience. The fact that SDP is a mandatory course is truly shameful: I much rather would have spent a valuable 20 credits in another manner.
  - Don't have such a broad and challenging problem domain like robotics. Robotics is an extremely complex field and many students
underestimate just how intense it can get. There are many limitations to what a group of third year undergraduate students with no/ little prior robotics experience can achieve in 3 months.
- Have more transparent marking. The marking was frequently extremely vague and contradictory. Often we followed direct advice from previous feedback only to be marked down for following this exact advice on the next piece of work.
- Be clear what the expectations are for student work: it was never clearly articulated whether the business aspect, the technical aspect, or the group work aspect was to be prioritized. If all three are important then that's of course also fine, but the issue was that there were conflicting impressions given by different instructors on what students should focus on.
- The course was extremely poorly organized. Many last minute ad-hoc announcements made over email, which of course get lost in the inboxes of most students.

- Perhaps, not punishing teams for heavily relying on technicians because they have taken more hardware focus than simulation. As both were introduced as viable standalone options.
- Reduce the number of reports that need to be handed in. It felt like 50% of the time was spent making user guides/reports/videos and not actually producing any software.
- Replace the organisers with people who a) respond to questions b) give actually useful feedback and don't then contradict themselves afterwards when you follow their feedback.
- The course is the only course I have come across in the Inf school that marks so closely to the Common Marking Guidelines (or whatever their proper name is) which can be very concerning when its not mentioned before the first grades come back. Essentially this year they should have been clearer about lab access.
- The marking criteria are too harsh. I don't get too much help from the experts.
- The timescale to decide a project from a simple prompt is very tight. The amount of work needed to get just the bare minimum marks is much higher than other courses. If group members slack, there is not much anyone can do other than report it at the end. The entire group suffers if even a few refuse to participate, and it affects everyone's marks which affect their degree. More concrete guidance and resources should be offered for groups struggling with active participation, not everyone is an expert at conflict resolution and many find reporting on their teammates intimidating.

  If you aren't interested or good at robotics. I feel you are at a massive disadvantage. A robot with not-so-complex software stands a better chance than a complex software with relatively simple robotics in everything but the industry day, which is even then only 15% of the grade. It also affects your ability to know what is possible and thus plan ahead, having one or two robotics experts, as some groups had, is not enough to succeed.

  This course feels like it forces a lot of understanding of non-informatics concepts. There are too many things to cover to many students may have never touched, like project management and business pitching. It is rather overwhelming at times. I understand these skills are important but they are also not universal to everyone.

  The workload is WAY, WAY too high. I know everyone says this, but I think it's pretty outrageous to make students go through this given the current circumstances, especially without lowering the difficulty of the course. SDP has been one of the most stressful things I've ever done and it has been mentally and emotionally taxing. As well as several meetings a week + all the coding/design work, we have to stay on a call until 5 in the morning every other week to get work done and it's still barely enough to get all the work in on time. That's not even mentioning all the work I have to do for my other subjects.

  I have probably spent more time on SDP this semester than for all of my other subjects combined, and I have missed most of my lectures and tutorials for everything else as a result. I have a learning disability and I don't feel it's something that's been properly considered in the planning of the course. I'm usually allowed extra time for coursework but because SDP is group work I don't get any. I've therefore been forced into prioritising SDP because it's the only work that I can't push back and therefore it's almost always the closest deadline. I know this is out of your control but I have had groupwork in the past and I have never struggled with it in the same way I have with SDP. Now that demo 4 is in, I've got 2 courseworks due in the next few days, and that's WITH an extension for both of them.

  I think I might have been able to handle the workload if SDP wasn't done remotely this year, but the remote learning has really pushed it over the edge. Many people with learning disabilities struggle with communication, so trying to communicate with 10 other people through a screen over the internet for tens of hours a week can be gruelling. Also note that many people don't even know that they have a learning disability, and don't have access to the same accomodations that I'm lucky to have.

  Obviously nothing could have been done about the remote aspect, but given that SDP relies heavily on communication, I think it's wrong to expect the same amount of work when all communication is done online which can be buggy, unpredictable, and frustrating. That's not even considering different timezones which can be nearly impossible to work around.

  I think a really quick way to lower the intensity of the course without changing it drastically would be to take one of the demos out. It would give everyone a little breathing room to deal with their other subjects and it would let people spend more time actually working on their projects. I know the 4 feedback sessions are really helpful, but you could always put a formative/less formal feedback session in place to make up for the missing demo.

  The workshops were quite vague and many of the skills that were supposedly taught couldn't really be applied. Maybe it would be more helpful to have something longer or recurring to make sure that the workshops teach the skill correctly.

  This course is bad. There are no two ways about it. Especially COVID has shown this. The fact that they planned for in-person and nothing else is embarrassing. That fell through and they clearly scrambled to put something together. People pay so much for this uni, and I expect far higher standards than this. Feedback has been vague and inconsistent. Advice on many topics has been lacking and the decision to use Webots was a bad one. Expectation was the same as when the course was physical, which is not possible to uphold. This course is a victim of incredibly poor planning, done by people who are individually incompetent, yet lack understanding of the students' perspective.

  This course needs to be completely overhauled - it was the worst done at University, unclear with its aims and how to get good
grades, totally contrived and all of these issues were just made worse by the pandemic. Clearer more prescriptive tasks and robots need to be assigned to stop staff not liking student's ideas.
2.1) What advice would you give to a student taking this course in future?

- Please try to avoid if the course is still not compulsory.

- Avoid it if you can. Choose ILP instead if you have a choice between the two. If it's compulsory, be prepared to work hard and not receive the mark you thought you would since the markers are extremely strict with their grading.

- Avoid this if you can

- Be aware that a lot of the course is teaching yourself how to do things.

- Be engaged with your team for the entire semester and make use of the experts' office hours

- Be prepared for a heavy workload and do not slack off for your group. Explicitly track and distribute workloads carefully and do not be afraid to flag up people who are slacking, if you are lenient it will be too late in the end, unfortunately. This will be a difficult course. If you are not interested in product design/robotics/business this will be daunting, be prepared as best you can and always contact your mentor if you need help.

- DO NOT TAKE SDP - it is a waste of time and effort

- Do not take this course.

  If you do take this course, make sure that you put in maximum effort during Semester 1 and your other classes during Semester 2, so that your average can at least remain high despite getting a bad grade in this course.

  Set your grade expectations very low and come to terms with the fact that ultimately you are 1/9 people working on it, and therefore there is so much you can do. I averaged over 85 until this point, and had never gotten a grade below a 75 at my 3 years here, and yet somehow I am getting 50s and 60s consistently in this course. Be prepared for this. Focus on your other classes and do not go above and beyond, since the probability of you doing the work for 9 people to get an A is virtually 0. Your efforts are better spent on other classes.

- Don't be too put off by marks which don't seem to accurately reflect the work you've put in. Try as much as possible to communicate the work you've done transparently to your teammates and overall focus on ensuring you have made a good contribution to the team.

- Don't expect to get any more than 70% in this course, as it's not possible...

- Don't leave a lot of work to be done on reports until day of submission.

- Don't put off tasks for later, it will never work

- Don't take it unless you have to. Realise that this course is only worth 20 credits. This course has made me literally physically sick (not exaggerating) each time I think about it, especially towards deadlines, I do not think this is healthy at all. The university staff maintain that this course is only worth 200 hours per person, but this is unrealistic. No other course has made me feel this way. It has killed any ambition I have to work in software engineering or the tech sector in general.

- Don't take it. Research ways to skip the course. Find loopholes. Switch to AI and Math degree for a year or switch to cognitive science. Do internships instead. This is the worst course I've taken designed by some incompetent course organisers. DONT TAKE IT.

- I dont think any advice would be relevant to post COVID students.

- If possible I would advise them to not take this course however it is compulsory for many students. This course has been genuinely stressful and I wouldn't want other students to go through the same ordeal our year group has.

  I would advise them to focus on the project management and commercialization aspects as the markers seem to put little emphasis on the technical aspects. Also would advise them to stick to the report templates and page limits as deviating from these at all will lose you marks.

- If you want good marks do not take this course.

  Be prepared to work really hard. It takes more time than all of the other courses combined(40+ credits).

  If it is still a pandemic, do not expect to build a real robot

- It's a lot of work, after which you will probably get a mediocre grade. But the freedom to work on your own project is great! If you are creative and like to work with others, you will definietly enjoy it

- It's more of a time investment than you think.

- Learn how to use Git early on.

- Learning from your mistakes as a team is the best thing you could learn.

- Make sure you communicate with your team members and let them know immediately if you're struggling with anything so that tasks can be redistributed and completed in time. Also, you can contact your client outside of demos and ask them for further feedback/tips on what to include in the next demo/etc. Nobody told us that information till very late and it would've been very useful to know before.

- Make sure you finish other coursework as soon as possible, as this course will take up all of your time

- Prepare to not have any time for other courses
Prepare yourself

- Probably get to know your teammates over the holidays and start thinking about ideas earlier, to make at least the first couple of weeks easier/lighter. Also figure out what software might be needed as soon as possible, and install it in the very beginning.

- Really focus on time management skills to help balance SDP with other courses.

- Spend a lot of time on it, but not on parts that seem to go nowhere. Make sure communication is open and has a low barrier to entry. Follow up on all team members to make sure everyone is doing what they are supposed to.

- This course is not worth taking.

- Try your very best to not do this course - if you go on exchange you will miss it for example. Working in such a large team is a nightmare.

- Work in advance.

- Develop more professional skills and learn to communicate with group mates

- If you can, DON'T TAKE THIS COURSE. This course is the worst course I've taken at university by a stretch. I did not enjoy it at all, did not find it stimulating and did not find it grew me in any helpful dimension. This course is painfully organised and thought-out. It feels like they threw it together last minute. I find it very hard to understand how a university of Edinburghs caliber has allowed this course to happen. I'm taking computer science but if I could redo my time at Edinburgh, I would change my degree so I would not be subject to this disrespectful course.
3. System Design Project

3.2) Please add any other comments you have about workshops, tutorials or labs on this course

- Expert meetings were very useful and provided insightful information.
- The practical workshops weren’t particularly useful, not much can be demonstrated in a short period of time, and students would be better served by finding resources online that they can work with for longer periods of time without the urgency of a live tutorial, as well as learning more in the process.

- Badly designed for the pandemic
- Didn't learn a single new thing in this course from any of the材料s
- Every workshop was irrelevant to the actual project

- I found there to be a large number of practical workshops, very few of which particularly aiding in my understanding. The one notable exception I would make to this would be the QA workshops, which I found to be very useful and informative.
- I have been so busy with demo reports, demo videos, programming and trying to keep up with my other courseworks that the workshops have mostly just gotten in the way (especially frustrating when they are mandatory). I don't really remember anything from any of the workshops because I've been so busy.

- It was sometimes unclear which workshops were mandatory and which weren't.

- Maybe it is due to covid but they have been worthless, how are you meant to debug something like an installation process through a ticket system? Usually there are cascading errors that appear only after solving the first error. In these scenarios, (1) you find a bug (2) try to fix it yourself, spend hours, cry (3) create ticket (4) 2 days later get answer for ticket, (5) OPTIONAL find out you didn't explain the problem properly + create new ticket + wait 2 days, (6) fix problem, cascading error occurs. REPEAT (2) to (6).

- Not really useful, often only attended by those who were working within that area.

- Some of the workshops were useful such as the one on QA testing, but the rest haven't particularly benefited the team.

- The QA + tech report writing workshop was definitely useful, although could be a bit more interactive in the last part (restructuring the sentences is very useful, but we only got a couple of examples - having breakout room discussions of some tasks might be more helpful). It would be nice to have more simulation workshops - why did previous years have ~3-4 hardware workshops while we only get one simulation? The careers workshop was nice as well, but could benefit from more examples - sometimes you really can't think of skills on the fly, or good/bad answers to an interview question, so those would help a lot.

- The labs with tech team were very good but provided slow progress since they were spread so thin

- The only experts which provided any useful help were Garry and Doug i.e. the SDP technician team

- The technicians on this course were great but there were simply not enough of them to cope with 20+ groups hardware. They tired their best at the online labs but were often working on 10+ groups at once. This led to them being distracted and meant we often had to wait hours to get anything done.

- The workshop leaders were not 'experts', the labs were poorly presented, with little enthusiasm, and no acknowledgement that everyone's personal computer situation will be different.

- The workshops are overall helpful but should be better organized.

- The workshops were mostly good, but instructions on webots was sorely lacking. When the entire course focuses on webots, teach more webots.

- The workshops, computer labs, and drop-in sessions were extremely useful. They helped clarify many things regarding the course and helped with Webots, commercialization, and QA.

- The QA workshop in particular was extremely good and useful and Barbara was very helpful in terms of ensuring we understand how to test our system.

- They don't feel like they've helped

- They were nice but often so packed with information and so close to the deadlines they applied to that their assistance was somewhat limited. Having experts on hand was great, however.

- While the business part of workshops were helpful, there was not enough support for the software side of things. The webots tutorial was useful, but since there aren't many resources available besides webot's own tutorials, some more pointers would have been appreciated. I realize this year was unique due to covid and that this might not be a problem in other years.

- workshop is very helpful
4.1) Please add any other comments you have about the presentation of course materials online and their accessibility.

- The way we were notified of changes was not effective, learn notification are not in order, emails were poorly formatted.
- Alot of the resources were from another course and butchered for use in SDP, this show the effort of the staff to produce and deliver the resources for the course

- Cancel this course, it's not a computer science course. At best it could be a business course, it's not about the system it's about marketing the system this has no place in the school of Informatics.

- Course organisers are poorly organised. Whenever they upload things the links are often broken.

- Didn't consider remote learning properly. Give more thought to it.

- Material for this course is a bit hard to find. Maybe you could some more visible links on the welcome page.

- Materials needed to complete the course have not been accessible. To create a complex robot, ROS is needed. ROS is not accessible to the students. The most updated ROS is only supported for macOS Mojave (10.14). macOS changed from Mojave to Catalina on the 7 October 2019. So if a Mac user (many many students have MacBooks) wanted to run ROS, they would have to download all of their data to a large external drive, remove the operating system from the Mac and then install the 2 year old operating system and then download all their data again. This is not accessible. Alternatively, a Mac user could dual-boot ubuntu which is much more easier to download ROS. However, what if a student does not have enough space on their Mac to dual-boot? What if they are sharing/borrowing a laptop and don’t have permission to reorganise the laptop structure? Another option is virtual machines, however they are not recommended by webots (the main platform for running simulations) since the graphics do not render properly and there are many other problems that come with it (I have encountered many of them myself).

I have tried everything under the sun including virtual machines, docker, AWS and DICE, to get ROS and webots working. I have spent 50 hours - yes I have recorded it - on this task. I feel absolutely disrespected by the course organisers for having to do this. This is not reasonable. Oh and by the way, in the end I wasn't even successful - our group had to use some shoddy packages instead. Ridiculous.

- N/A

- Please make a single structure for accessing materials on Learn that all courses follow.

- See above

- The Learn page is very difficult to navigate and important resources appear in different places. The announcements section is particularly misleading as the announcements do not appear in date order which means its easy to miss new important announcements.

- The SDP guide and other resources are easy to find. Better breakdowns of what the staff are looking for when marking would be useful (the mark scheme for the peer demo marking isn't particularly clear).

- The announcements on learn were in a weird order meaninng sometimes new announcements can be missed. Some information was only available via emails which is inconvenient.

- The course materials are easy to access.

- The fact that all the necessary information was in emails, announcements, reference document, gitlab, course information, course materials and assessment was incredibly confusing (and Learn's bug with Announcements being out of order definitely did not help). A lot of Informatics courses have one table in course materials with links and documents that are needed each week, and it might be really useful here, especially for the last couple of weeks when there's so many things to do.

- The learn page was easy to use. Often several deadlines would be split across several emails or the same info would be presented slightly differently in different parts of an email. This sometimes made it difficult to follow what the deadlines were.

- The materials offered were abundant and presented nicely. It was, at least, easy to see where everything was. It was nice to have a lot of resources.

- There exists a general lack of transparency about what constitutes an A grade and generally the marking criteria seem reasonable although often I find marks being deducted for incredibly specific considerations which are not always entirely relevant to the intended functionality of the robot.

- Was very thankful that the sessions were recorded

- very detailed
7. System Design Project

7.1) Reflecting on your experience of hybrid teaching and learning on this course, what has worked well for you?

BAD. The whole idea of SDP is to build a physical thing therefore it translates very poorly into hybrid learning.

Having a timetable of demos and deadlines helped to an extent as with the online teaching it's been really challenging to keep track of everything/where I am in the course.

I like the ability to make the product entirely simulated.

Meetings on teams were quite effective.

N/A. There was no in person element to my teaching.

None of my courses this semester were taught in a hybrid model as they were all entirely online. All of my other courses adjusted very well to online teaching but SDP did not work online at all.

Not a lot.

Not at all for this course.

Online Teaching made this course much more tough and worse, as most of us were unable to access the University physical resources and were unable to construct stuff in real life.

Stop calling it hybrid, it's only been online, no one is falling for it

Teams

The expectations placed upon us in regards to the hardware version of our robot were frankly unreasonable, and I believe entirely as a consequence of our lack of physical access to the building. I appreciate the work the technicians put in to the projects, however, ultimately it will not be the same as having an entire group working on it over the course of a semester, nor do the technicians have the same depth of knowledge about the project as the group. Therefore I think the marking criteria in regards to this specific part of the course was deeply unfair. Other than that though, I did find hybrid teaching to not impact too substantially on work completed on the simulation, website, and producing reports.

The experience was all online so nothing went well.

The meetings with the "client" were a good form of feedback.

The more private office hours

This course made good use of Learn, and made the most it could of a bad situation with simulators and lab technicians available.

This course should be used as an example - of what not to do!

This course was well organized and being online didn't harm my learning at all. Probably the best course in terms of online organization!

Very little - it was extremely difficult to work on a simulated robot when everyone was in entirely different time zones

Very little. Working remotely has made every part of collaborating in a group more difficult. Everything takes longer to do and it's frustrating.

Works well. but hard to find people offline.

gather.towns was a neat experiment, but truthfully the novelty fades off quick and at the end of the day is just in no way a suitable replacement for a project fair that is actually in person

it has worked ok, but building a robot over teams is an extremely difficult task. This course hasn't been hybrid, just purely online.

online teaching
12.04.2021

7.2) Is there anything else you'd like to tell us about your experience of hybrid teaching and learning on this course that would help us improve our approach?

- Approving ethics research and also building groups with less time zone difference, though that's only relevant for online learning

- As said previously, more interaction between groups - e.g. gather.town could be open throughout the semester, and people could hang out there and compare progress or projects if they want to. Even though this online project has worked surprisingly well (props again for team creation!), it still felt isolated.

- DO MORE LIVE.

- I think this course struggles with the lack of in person meetings within groups

- I've found that it's very difficult to clearly express myself without real face-to-face communication, which is especially important when you're designing a complicated system with lots of connected components. There are things that I didn't even consider previously, like how important hand gestures are when trying to get someone on the same wavelength as you. An okay substitute for this is to share your screen and draw/move your mouse about but it's not the same thing.

We found as a group that we could make the most of working remotely if we all agreed to set aside the same hour every day for SDP and then we could just drop in a call whenever if we had any questions for eachother (or we would sometimes all join and then mute our mics). I guess this is pretty close to how collaboration would work if we were in Appleton.

- It's a course which is entirely group work based, although we haven't been able to meet in person at all. I find it quite hard to communicate online sometimes, and I feel like some slight support could have been given to groups. Perhaps the workload could have been reduced slightly to account for groups finding this communication hard.

- Make it hybrid next time.

- More attempts at creating a social atmosphere. One of the main things of SDP is to be able to interact with those in your year.

- More detailed tutorials for the software necessary for hybrid work

- N/A (2 Counts)

- N/A. There was no in person element to my teaching.

- No adjustments were made in the marking scheme (testing, user guide) that address the fact that this is a project done in SIMULATION and not in the real world.

- Please recognise how much more effort this course takes when having to be done completely remotely.

- SDP doesn't particularly work as a remotely taught course, especially as there was little guidance or adjustment to take this into account - especially in terms of consideration of the challenges posed by the remote aspect and how this impacts the team's progress/work. SDP has taken up a considerable amount of time - to the point where it has impacted the other subjects (all of which have increased their coursework to compensate for remote teaching) causing me to really struggle this semester.

- SDP is a course that just should not be done in a virtual environment.

- School of Mathematics started giving out resources to students. I think people from the School of Informatics will benefit from it a lot. Please give out borrowable resources such as:
  - GPUs to do machine learning coursework
  - Laptops that run Linux (helpful for Inf students) (Library laptop run windows & its hard to do our coursework on them)
  - Tablets to tutors. I'm tired of seeing squiggly lines in my tutorials. Help them do their job!

Redesign your websites:
- The internal Informatics websites are horrible. Now that everything is online, it hurts my eyes to see ugly websites every day. This is right from filling time-sheets to course organisers website. Use Edinburgh's boilers plates. Now that everything is online, you should definitely level up your websites because students have to start at it for hours.

- Should allow more software-oriented projects since access to hardware is limited/impossible.

- Stop calling it hybrid, it's only been online, no one is falling for it

- Terrible due to materials not being accessible and little to no support being available for setting up said materials.

> Copied and pasted from another section:

Materials needed to complete the course have not been accessible. To create a complex robot, ROS is needed. ROS is not accessible to the students. The most updated ROS is only supported for macOS Mojave (10.14), macOS changed from Mojave to Catalina on the 7 October 2019. So if a Mac user (many many students have MacBooks) wanted to run ROS, they would have to download all of their data to a large external drive, remove the operating system from the Mac and then install the 2 year old operating system and then download all their data again. This is not accessible. Alternatively, a Mac user could dual-boot ubuntu which is much more easier to download ROS. However, what if a student does not have enough space on their Mac to dual-boot? What if they are sharing/borrowing a laptop and don't have permission to reorganise the laptop structure? Another option is virtual machines, however they are not recommended by webots (the main platform for running simulations) since the graphics do not render properly and there are many other problems that come with it (I have encountered many of them myself).

I have tried everything under the sun including virtual machines, docker, AWS and DICE, to get ROS and webots working. I have spent 50 hours - yes I have recorded it - on this task. I feel absolutely disrespected by the course organisers for having to do this. This is not reasonable. Oh and by the way, in the end I wasn't even successful - our group had to use some shoddy packages instead. Ridiculous.
- This course is almost impossible to do in hybrid teaching circumstances. Teammates and I are trying our best to achieve the work they want us to do. But most of the time it's very hard in the pandemic situation.

- This course should not be taught online as it simply doesn't work. It's nearly impossible for the large teams to coordinate and build a hardware robot without being there in person. I understand this was not known at the start of the semester but there have been very little adjustments as the semester progressed.

- This course should not have run this year, this is very clear from enduring it for a semester.
11. Thank you -

Thank you very much for taking the time to complete this questionnaire. Your response and comments will be fully considered.

Please provide any additional comments you may have about the course, the teaching on the course or the resources that support it in the box below.

- As a final note, I found the staff running the course to be very petty and inflexible with respect to the current global situation.

Specifically, Dr Tonneau, who frequently would not review the documents and videos various SDP teams produced for the demo's and would ultimately give incorrect marks in bad faith if not confronted in the demo meeting.

- Cancel this course, it has no place in the school of Informatics

- I understand that to make this course is a very difficult task. However, this does not excuse that lack of quality.

I feel disrespected by how inadequate this course is. I am paying thousands of pounds to be a student and this is the quality of the course? Please make this course better. Move from robotics (during the pandemic at least - maybe in person this course is very enjoyable). Make the materials needed for the course accessible (provide alternatives to webots and ROS in this case or move away from using robotics). Make marking more consistent and based against a clear rubric. I understand marking is subjective, but it should not be so incredibly different week to week and group to group.

A few questions you probably won't read:
- How much thought was put into how students would be able to handle a practical robotics completely remote in general?
- Were there any alternatives considered?
- Was the main reason why the organisers chose robotics for this semester that it has always been a robotics course and so therefore easier for you to deal with?
- If yes, I am truly disappointed as you have put your own interests ahead of your students. There are many alternatives for designing systems that would suit being online. E.g. any software engineering project ranging from apps, websites etc. Look at the honours projects for inspiration.
- How much thought was put into how students could be able to engage with Webots and ROS remotely? Particularly setting up the environment.

I am ashamed to even talk about what I am working on to my peers at other universities. It is embarrassing.

- I understand why many enjoy this course. The organisers seemed eager and professional and I do not have any grudges with them or the concept of the course, I personally don't like the premise as I am rather neurotic and easily stressed, but these are just issues with me and not the course. I think the SDP premise is very interesting.

However, I cannot say I enjoyed the course, regardless. The amount of work required to even approach a B goes well beyond the 200 hours recommended even if everyone in the group puts in the 200 hours. If some slack, others must take on the burden as no considerations are offered. I understand we need to solve tough scenarios, but at the end of the day, all that happens is everyone who did participate suffers a lower grade at the expense of the slackers. oftentimes it is too late to rectify slacking members before the deadline when you determine they are slacking.

SDP requires exceptionally to get anything above 70. I understand this is standard, but it is exceptionally in non-informatics concepts. It is unclear and often deliberately vague what constitutes exceptional. A lot of effort goes into trying to make reports or parts of the project exceptional but it feels these marks need rather insane levels of dedication to the project, which will only ever merit you a few exceptional marks. Even so, making a few mistakes takes away any chance at an A, which feels rather harsh. You should be able to have an imperfect but excellent project and still get an A3 at least. I fully agree A1 should be reserved for exceptional groups, but those that WOULD earn exceptional marks often cannot.

SDP requires well over 200 hours to get anything close to a B. I feel this cut into not only my free time and sleeping hours but my time for other courses too. I suffered from my mental health at times trying to get everything done. Some of this is my fault for poor delegation on my part, but some of it was due to the strictness of the marking requiring me to feel very pressured to try and make everything I did was perfect.

I feel this course could be fun, however this year it was far too much pressure for little reward. I am glad it is finished with and I really hope others at least enjoyed it more.

- It should have been cancelled this year

It was a very interesting course where I learned a lot, but it was too compact leading to a very high workload and some stressful periods. It is unclear to me the purpose of the in group peer review as it seems to have been said that it would be used for mark adjustments but also not have an impact on marks. Also, it's said not be a zero-sum game but we are ranking people on how much work they did above the group. The idea of marking my peers "*" or "-*" feels like snitching on them and I can't see how it maps to an industry setting. Gather town was a good choice to use and the industry day was enjoyable.

Overall I am satisfied with the course, I learned a lot from it, and I (mostly) had fun doing it thanks to the great team we had. My only problems with the course cannot be simply handled by this course's organisers alone, so please do not take it personally. I am thankful and grateful for all the work you put into this course.

Overall, I like the idea of this course. But the execution and organisation could be better - when the whole school of Informatics uses SDP as a boogeyman for younger students, it's a sure sign that there's something wrong. Like it has been said many times before, it's a course that you hate doing, but love after you've done it - but people should be able to love it during the semester too! And it definitely has the potential and the capacity to be better/more fun and less anxiety-inducing.
Another thing is the inconsistent 'excellency' mark: not only it differs a lot from other courses, it also differs from demo to demo. If a group did something above and beyond for one demo (and got extra marks), and continued doing that for the next demo, why do they not get extra marks again? That's still going beyond, because it's building on previous "excellent" work. And the feedback for the demos could be a bit clearer - sometimes it's hard to understand why a team did something wrong, especially if they were asked to do it before specifically for the next demo.

It feels like a lot of negative things, but there were positives as well. I understand how hard it must be for all staff in these times, and in general, they did their best, and we're grateful for it. The mentors are really great and helpful too; I also love the fast response time on Piazza.

- Provide proper feedback to students.

- The intuition for this course is good but this course has to be reformed. Especially during the pandemic situation, course organizers are still seeking a way to insist on the previous demo procedure, which is not a wise decision.

- This course needs to be majorly revamped and changed dramatically.

I would say this course is principally inconsistent with what the School of Informatics wants from its students. For the first 2.5 years of the degree, there is virtually no group work, except for professional issues which is only 10% group work (20% of the wiki, which counts for 50% of the course). The Software Engineering group work also has the option to be done individually, or you get to choose your pair, and therefore individual students have a LOT of impact in terms of their final grade for that coursework.

However, for some reason in 3rd year, the School of Informatics then decides that having a course that is 90% group work is now a valuable addition. Why? If we have barely had group work for the first 2.5 years, then it follows that the School does not put a high emphasis on group work. Therefore, why is an honors level class now determined almost entirely by group work? They should either have high levels of group work throughout the first 3.5 years, or not at all.

Additionally, the problem of slacking members needs to be thoroughly and directly addressed. I have 3 people in my group that have contributed extremely minor things, while the 6 remaining members have had to work even harder in order to make up for their slack. Suddenly, this is no longer a normal workload for a course, and is now by far my most demanding course - all for a grade that those 3 slackers will get. How is this fair?

Moreover, if we work with the assumption that a hallmark of a "good" course is one whose grades are highly correlated with students' performance on other courses, since that means that what the School considers to be a successful informatics student appears across all courses, then this course, at least in my case is an abject failure. How can I average over 85 in my first 2.5 years here and suddenly be getting by far my worst marks in this course specifically?

I have the following recommendations for the course:

1) Make it OPTIONAL for Computer Science students. This should in no way be a required course. If people want to opt-in to group work that is fine. However, if you want to avoid it you should be allowed to.

2) Make is at least 50% individual work - you can do this by expanding the weights of the individual reports, and increasing the amount of scaling that group members receive as rated by other group members and their mentor. There can be an appeals system if a student is unfairly targeted by their group, but this will likely only happen in a small minority of cases.

3) Make it a 1st semester course. Many students back load their credits this year, and I know many people who split their credits 50/70 or 40/80 across Semester 1 and Semester 2. Moving this course to Semester 1, and switching it with ILP will mean that students have more time to work on it. Additionally, if they are unhappy with their grades, it means they have more semesters to try as hard as they can to make up for their bad SDP grades. I know I certainly would have tried much harder Semester 1 if I had known that my average was going to drop this much due to SDP.