Redesigning Engineering Curricula around Studios

Roger Hadgraft, Julia Prior, Justine Lawson, Tim Aubrey and Rob Jarman
University of Technology Sydney
Corresponding Author Email: roger.hadgraft@UTS.edu.au

CONTEXT
Engineering curricula must develop three categories of skills and knowledge – (i) design and problem solving in complex, socio-technical situations, (ii) the technical knowledge to support the design process, and (iii) the interpersonal skills to support engineering team processes. Traditional curricula, based on teaching the technical knowledge and skills, under-deliver developing the design and interpersonal skill sets. Since at least 1974, several universities worldwide have implemented various forms of project-based learning (PBL), although PBL has not become the norm in engineering curricula as it has in medicine. This university is now embarking on a bold move to implement studios in each of its engineering programs, extending the work of the last three years in developing and implementing the Software Development Studio.

PURPOSE
The paper provides insight into the processes being used to reconceptualise several engineering programs around a studio spine. Examples will include environmental engineering, biomedical engineering and data engineering, each with a somewhat different approach. This paper explores the nature of studios based on our experience and the key differences that should distinguish them from more familiar forms of project-based learning as they are practised in engineering. Academic concerns are also explored.

APPROACH
A series of staff development workshops have been held, in which the desirable attributes of studios have been explored. Staff identified key issues to be resolved in the design and implementation of studio experiences for students. Program teams met to consider the big ideas in each program, how these will be assessed, and how each student will tell their own story of career development through their e-portfolio.

RESULTS
Staff engagement in the project teams and in the workshops has been enthusiastic, with each program adopting its own approach to studios including the development of fundamental concepts in years one and two, through the development of core skills in years two and three to the development of specialisation skills in years three and four. Details of studio workshops and the issues raised by staff are reported in the paper. These link to other papers on assessment and e-portfolios at this conference.

CONCLUSIONS
There have been many international calls for more engaging and relevant engineering curricula, which develop the full range of capabilities required for engineering practice – design skills, technical skills and interpersonal skills. Further, engineering practice is increasingly complex and engineering education must reflect that. Studios are an effective mechanism for engaging students in mentored, complex problem solving that will develop the full range of professional skills. Several engineering programs are now being re-engineered around a studio core. In parallel, staff development is focused on the conference theme: the changing role of the engineering educator for developing the future engineer.

KEYWORDS
Engineering curricula; studios; project-based learning; PBL; engineering capabilities; graduate attributes.

The Context

We live in a world of constant change and students will likely experience several distinct careers during their lifetimes. There is increasing evidence (Institute for the Future 2015) that graduates will need to be innovative, with creative and critical thinking skills as well as the ability to engage others with their ideas. At a time of significant global challenges, we need to graduate engineering and information technology professionals who are future oriented.

UTS is committed to produce graduates who (UTS 2015a):

1. are equipped for **ongoing learning and inquiry** in their personal development and professional practice,
2. operate effectively with the **body of knowledge** that underpins professional practice and
3. are committed to the actions and responsibilities of a professional and **global citizen**.

To formalise these ideas, in late 2014, the University articulated the Learning.Futures model of learning comprised of (UTS 2014):

1. An integrated exposure to **professional practice** through dynamic and multifaceted modes of practice-oriented education
2. Professional practice situated in a global workplace, with international mobility and **international and cultural engagement** as centre piece
3. Learning that is **research-inspired and integrated**, providing academic rigour with cutting edge technology to equip graduates for life-long learning

Many universities have similar commitments through their learning and teaching strategies. Learning.Futures, however, has mandated key shifts in classroom practice:

1. **Flipped** learning using the best of online materials (not necessarily creating them ourselves)
2. **Collaborative** learning activities, e.g. inquiry-based activities, labs, studios, projects
3. **Real-life** experiences, e.g. internships, community projects, competitions
4. **Authentic** assessment based on authentic tasks
5. Diagnostic **feedback** (UTS 2015b)

The Faculty of Engineering and IT Strategic Plan

The Faculty of Engineering and IT has interpreted these intended outcomes as:

> *To create, develop and disseminate world class technological knowledge, equip engineering and IT graduates to contribute in a global environment, and co-create value with industry and the community.*

Within learning and teaching, our intent is to:

1. consolidate a **flexible, practice-oriented, and inclusive learning environment** that creates graduates who are sought after and globally competitive
2. integrate and encourage **innovation and entrepreneurship** into our courses and research
3. **integrate teaching and research**
4. focus on key areas where we can make a difference to the world through **transdisciplinary approaches** and the science of engineering

We have interpreted the above needs to create a set of key **requirements**. Our learning environment shall be based on:

1. personalized learning as the heart of the student experience
2. practice-oriented learning based on inquiry (question asking is a key skill)
3. development of global citizens with global perspectives
4. access to the professional body of knowledge, which is linked to research.

Implementation – studios, online learning and assessment, e-portfolios

There are **three key ingredients** to building a 21st century learning environment to deliver these requirements:

1. First and foremost, it must be personalised. **E-portfolios** have emerged as a high impact practice in which students can co-create (and document) their emerging futures as global citizens (AAC&U n.d.). Think of it as a continually evolving CV. Whereas a CV is backwards looking – this is what I have achieved in my previous roles – an e-portfolio is also forwards looking – what do I now need to achieve? Where are the opportunities (work placement, studios) where I can build those skills? This step is all about **personalisation**.
2. **Studios**, projects, and work placements are all *experiential* learning opportunities to develop skills and knowledge. Research from Prince and Felder (2006) supports active learning. Whereas traditional teaching and learning is often used in teaching engineering fundamentals, the complexity of modern design challenges, e.g. designing and building the NBN (National Broadband Network), are not amenable to lectures and tutorials as if there is a right answer. This step is all about **professional practice**.
3. Finally, we need to recognise that if you can ask the right question, you can find the answer (or answers). **Online learning** is increasingly the norm (see Lynda.com, Khan Academy, Codecademy, Udacity, …) and we can expect that all fundamentals will soon be available online with appropriate assessment tools. Students will be expected to demonstrate mastery of certain modules (beyond a 50% pass) before they can complete certain studios where that knowledge will be required (Lindsay and Morgan 2016). This is all about flexible **knowledge and skill acquisition** and creation.

Some History and Context

UTS Engineering has a long history of engagement with practice-based learning (Parr, Yates et al. 1997). The revised curriculum from 1998 emphasised professional formation, personal development, and academic development. The curriculum became practice-oriented and learner-centred, embodying environmental and social sustainability:

> *The course components [would] be mutually informing and synergistic, in order that the students experience their development as professional engineers, citizens, and lifelong learners in a holistic and supportive environment. (Parr, Yates et al. 1997)*

Many of the elements of the **core subjects** remain: Mathematical Modelling 1 and 2, Physical Modelling, Engineering Communication, Engineering Computation, Economics and Finance, Engineering Management (now two subjects: Project Management, and Commercialisation and Entrepreneurship). Sadly, 'Engineering through History and Towards Sustainable Futures' has gone and Technology Assessment is an option rather than core (now known as Interrogating Technology). 'Uncertainties and Risk in Engineering' has been replaced by 'Design and Innovation Fundamentals'.

It is interesting to see that almost 20 years on we are still grappling with the issue of what is the core of engineering practice.

The Professional Practice (Internship) Program

UTS also operates the largest internship program in Australia, now almost 50 years old. More than 1,000 students complete an internship each year. For most engineering students, two six-month placements stretch their four-year degree to five years. Internships are usually taken in second and fourth years and many students are already employed by the end of their second internship.

Software Development Studio

Our Faculty's Software Development Studio was designed to emulate a real software development practice, where student teams work on industry-initiated projects. Team members are not peers, but come from different subjects, years of study and degree courses. This mixed team approach mirrors the diverse experience in a real workplace, and encourages peer learning and peer mentoring. The teams also have half-a-dozen industry mentors who spend one to two hours weekly, working face-to-face collaborating with the teams. The students learn how to use sophisticated software development tools that they may encounter in the workplace, such as Bitbucket to share code and assets and HipChat, Jira and Confluence for communication and project management. Students can get credit towards their degrees through partner subjects or special project subjects.

One student said to the director of the SDS a few months after participating in the SDS, "You didn't teach me anything; you created an environment that enabled me to learn!" The approach does encourage self-directed learning and discovery.

A number of students have participated in the studio for more than one semester. A few students have returned for the 4th semester, having started in their first semester of their degree and are now in the latter stages of their course. They experience evolving roles in their teams each semester, formalised for particular students, e.g. some are now valuable team leaders.

A recent Grattan report (Norton and Cakitaki 2016) into Australian higher education states that "IT graduate skills and attributes are mismatched with the labour market." The report goes on to state that IT graduates often lack the necessary communication and interpersonal skills, which puts them "at a disadvantage." One of the SDS's raison d'êtres is to address just this issue. Strong emphasis is placed on the deliberate development and formative assessment of teamwork skills, particularly communication and collaboration.

Key characteristics of studio learning environments are: real projects, industry mentors, and reflective practice. There is a long tradition in the use of this approach in the creative arts disciplines, which is firmly based in Schön (1983)'s work on the reflective practitioner.

The research into student learning in our Software Development Studio environment is strongly grounded. Over several semesters, we have performed action research in the SDS on teaching and learning reflective practice and the development and assurance of complex graduate attributes.

An ethnographer attended most of the studio sessions during the first three semesters that we ran the SDS, observing the students, academics, clients and industry mentors while they were working and learning together, and collecting data in response to the typical overarching ethnographic research question: "What's really going on here?" We have carried out several sets of interviews on reflection and graduate attribute development and assurance. We have collected quantitative and qualitative data on the students' self and peer assessments of the communication and collaboration aspects of teamwork over several semesters. In addition, for each teaching session, students have complete formal feedback surveys (closed and open questions) on their software studio learning experience. We have published papers on the experience and learnings of the pilot SDS (Prior, Arjpru et al. 2014), teaching and learning reflective practice in the SDS (Prior, Ferguson et al. 2016) and the studio as the place where “things come together” (Prior, Connor et al. 2014).

Prior, Connor et al. (2014) shows evidence that “the software studio provides learning that genuinely prepares students for professional practice.” In this paper, we claim that the student learning experience in the SDS “entails dealing with complex technical problems and tools… involves working effectively in groups… results in the building of students’ self-confidence and the conviction that they can successfully deal with the challenges of modern software system development. It is learning that allows the accomplishment of the more elusive professional competencies” [p1].

The Faculty’s definition of a studio is shaped by the Software Development Studios and adapted from the ALTC’s Studio Teaching Project (2015): “*The studio is a learning community of students, teachers and others such as industry mentors and practitioners, interacting in a creative, reflective process to develop some kind of product, in a physical environment/space that enables collaboration*.”

The key ingredients here are: real projects leading to real products, with industry mentors, using collaborative and reflective practice. A detailed comparison of studio-based, problem-based and project-based learning is included in Table 3 in the Appendix.

How do academic staff see studios?

Two workshops were organised during 2016 to grapple with the introduction of studios in Engineering and IT. The first workshop was mostly aimed at Deputy Heads of School for Learning and Teaching, together with some other key teaching staff. The second workshop cast the net more widely for those who had an interest in exploring the issue.

At the first workshop, staff were asked to identify **key issues** that they felt needed to be addressed. They then worked on some of these issues in small groups, with results as follows:

Purpose is a key issue. What are we trying to achieve? Some of the ideas presented include: The getting of wisdom (by both staff and students); learning through doing to enable the development of professional practice skills; allowing and supporting excellence – students can/will exceed scope and expectations; exploring (and stretching) boundaries – institutional structures and systems currently constrain our understanding of teaching, learning and assessment; integrating a number of existing subjects, e.g. across a semester, or longitudinally across several semesters.

Real projects are seen as vital for studios, including: design and build an artefact for a competition, e.g. the Warman competition; cross-disciplinary projects versus subject specific or discipline specific projects; open source (software) projects; research based projects; Engineers Without Borders (EWB) and other NGOs; greater engagement with industry.

Naming issues – is *studio* the best name? How does it differ from what we think of as project or problem-based learning?

The **student experience** is a key ingredient in the UTS learning model (above). Studios can improve the student experience through **flexibility** for student (career) directions; students need to investigate on their own; they need to move **outside of their comfort zones**; we need to define student roles and support them to achieve the intended capabilities and attributes.

Students should be able to **communicate** in a number of modalities and work in teams, including across multiple year levels. No two students will have the same experience and mapping diverse student **achievement** will be a challenge, particularly in the area of technical skills. Studios should support in-depth technical learning in threshold subjects

Student success is a necessary motivator: exciting projects lead to infectious motivation, as many have experienced. The Value proposition is that students build a **portfolio** to get a job, with valuable artefacts to show at a job interview.

Failure needs to be reconsidered. Would we be better to speak instead of 'not achieved yet'? Nevertheless, we will need to help students deal with freeloaders in teams or dysfunctional team members and also with students with differing levels of commitment.

Reflective practice is not well understood by engineering educators; this and other aspects of studio approaches make it difficult to understand 'studio'.

There are some key **curriculum design** issues: studios will be supported by online modules to develop knowledge and skills. We may need a limited number of studios to build core capabilities in each discipline. We need a good supply of projects, including bigger picture, world/societal problems and issues – industry backed, mentored, open-ended. We want to support different ways of learning – guided, not taught; learning on demand (and sometimes teaching on demand); shorter, high intensity, rather than spread over 11 weeks; collaborative; rule breaking; pull, rather than push learning; enduring projects may work best; lots of learning paths; teacher (and students) negotiate objectives. We want to cross-fertilise from studios to other subjects.

Assessment is a key issue. Assessment should be authentic and contribute to the student's portfolio. Assessment should also be holistic and not based on the sum of a series of assessment marks. There should be credibility (both validity and reliability) in demonstration of learning outcomes. Students will need to negotiate intended learning outcomes, particularly when multiple disciplines are involved.

Grading is an issue that we need to consider … or, more to the point, ungraded passes may be a better way forward. Grading leads to teachers' values being imposed on student learning. That may not sit well with a true, student-directed environment built around individual portfolios.

Workload for staff must be accounted for, both academic and professional staff. One concern is **scalability** for large numbers of students. Is there an ideal number for a studio? Space demands will be significant, particularly by encouraging more students to spend more time on campus. Where will they all sit/stand and work?

Staff skills will need to be enhanced. Tutor training will be required, for large classes, in particular. These include: facilitation skills – students are guided, advised, taught on demand (pull learning); professional skills, e.g. resolving team conflicts; and IT skills.

Engagement of others is essential, e.g. industry, as guest and mentors. **Motivation** will be generated by bringing the Faculty together, across disciplines, teaching, research, etc. We need to determine whether we will get buy-in from research-only academics? We will also create a learning environment that is broadly inclusive – team focused: staff and students, young and old; academic, professional, industry mentors working collaboratively – team learning and team teaching.

Timetabling/scheduling faces several challenges – formal classes versus informal team meetings (and space for both); open access to laboratories and equipment – we need a booking system and a certification system for laboratory and equipment access. Fortunately, there is development already happening on this front to allow students access certification based on their student card.

1. Scalability
2. Learning precinct – common digital space
3. Flexible (and safe) access to laboratories and workshops is important

Space includes the physical, metaphysical as well as tools and resources. Spaces include creative spaces; laboratory space for design, build, test; open access, easily configurable;

setup for human interaction rather than overloaded with technology – ambience and atmosphere are important. How specifically does a space need to be furnished and configured? What are the key attributes (group addressable TV screens, large writeable walls)?

Space is also **metaphysical** or logical; after all, it includes many other spaces, e.g. students' homes, as well as transport, cafes, etc. Tools and resources should provide seamless integration of the physical and virtual worlds, e.g. provide a range of computing tools to support team projects, e.g. Trello, Confluence, etc.

It is clear then, that there is much to think about as we introduce studios to our programs.

Reimagining curricula with studios

So far, three programs have been transformed: data engineering, environmental engineering and biomedical engineering, each with a different story.

Environmental engineering was in need of a revamp to attract more students. How can we embed the principles discussed earlier (personalised, practice-based learning, innovation, research-connected, transdisciplinary)?

We drew on Dowling and Hadgraft (2013), which set out a framework for environmental engineering, subsequently endorsed by the Environmental Engineering College of Engineers Australia (Engineers Australia 2016). **Four underpinning principles** were identified: sustainability, systems thinking, an integrated approach, and critical thinking.

Seven industry contexts were also identified: Natural environments; Agriculture; Industrial environments; Built environments; Extraction industries; Utilities; and Transport. We aimed to include as many of these in the studios as possible.

Six Process skills were identified as the heart of environmental engineering practice: Investigation; Modelling; Design; Assessment of impact, risk and sustainability; Environmental planning and management; and Audit, compliance and review.

We took a hard look at many of the subjects that made up the environmental major, finding many subjects heavy with content and limited application (Table 1), which we transformed into five studios (Table 2), which develop the six process skills, above, as well as giving students exposure to several industry contexts: Natural environments (land, soil and water management), Industrial environments (waste treatment and pollution), Built environments (energy), and Transport.

Each of these studios represents half a semester's workload, nominally 300 hours of work. Industry partners will be invited to contribute to the development of each studio.

Table 1 - Environmental engineering major subjects

Principles of Soil Science	Environmental Engineering Analysis	Water Supply and Wastewater Engineering	Renewable Energy Technology	Road and Transport Engineering
Principles of Environmental Engineering	Environmental Chemical Processes	Pollution Control and Waste Management	Environmental Biotechnology	Environmental Planning and Law

Table 2 - New environmental engineering studios

Environmental Engineering Practice (focused on soil and water, pollutants and treatment)	Managing Land and Water Resources	Urban Water and Waste Management	Energy Engineering (including energy audit, efficiency, renewables, biofuels)	Sustainable Urban Transport Systems

Biomedical Engineering offered different challenges. It had already been decided that students would complete two out of four specialties: Medical and Assistive Devices, Biomaterials, Genomics and Bioinformatics, Health Economics and Innovation. Each of these sequences would be made up of three standard subjects. An easy approach to studios in this case was to combine two of the three subjects into a studio, with the third subject serving to develop necessary skills and knowledge in readiness for the studio.

Students also have four free electives that would allow them to undertake a third specialty if that was of interest.

These studios, of course, come late in the program (third and fourth years). We wanted to include an introductory studio, which would help students to understand why they were studying a broad range of subjects: Cell biology, Genetics, Physiology, Anatomy, Health Care systems, Biomedical Regulation and Ethics, as well as Circuits, Signals, Programming, Chemistry.

A *Fundamentals of Biomedical Engineering* studio was proposed to run across semesters 2 and 3, with the intention of students engaging in simple problems from each of the four specialties listed above. This studio introduces these four key areas of biomedical work and creates the reason to learn the medical, engineering and data sciences needed for work in the specialty studios in years 3 and 4.

The new **Data Engineering** program replaces the old ICT Engineering program in 2017. It represents a rethink from a focus on the tools (ICT) to a focus on *data*, which underpins our business systems, such as the world wide web, Google, Facebook, electronic ticketing, accommodation booking, e-government etc.

Each of these data engineered systems must satisfy a set of business *requirements* and it must be built in a user-centred way. The engineered system itself is represented in four parts: data gathering (the user interface); data pre-processing, transmission and storage; data analysis and decision making; and data presentation and action.

Within this broader context, *specialisations* include: advanced data analytics, real time systems, image processing and computer vision, internet science, and cybersecurity.

There are three studio pairs (6 individual subjects), which run across semesters two to eight, with the curriculum represented in three stages: Fundamentals, Applications, and Professional.

The *fundamentals* stage is the first three semesters, which develop fundamental skills – design, technical, and professional. As well as the usual maths and physics, this stage includes Engineering Communication and Introduction to Data Engineering to develop basic design and professional skills such as teamwork and communication skills. Technical subjects included are C programming; Information and signals; Sensing, actuation and control; Network fundamentals; and Introduction to Data Analytics. The *fundamentals studio*, which stretches across semesters two and three, gives students an early chance to integrate the various aspects of data engineering. They might design a 4G network for a sports stadium, analyse data from the public transport ticketing system, or design an app for a new online service.

In the *applications* stage (semesters 4 to 6), students dive deeply into one or more of the technical specialisations above. They may work in a group across the specialisations, for example, an image processing application with aspects of data analytics and cybersecurity. Each of the two studios at this stage are 12 credit points (50% of a semester).

At the professional stage (semesters 7 and 8), students undertake two further 12 cp studios, this time concentrating on the total problem, carefully investigating organisational and user requirements. This stage is supported by the core subjects in Design and Innovation; Project Management; Economics and Finance; Entrepreneurship; and Interrogating Technology.

Discussion and Conclusions

Engineering education is on the cusp of major change. Fundamental knowledge will soon be learned and assessed online. The free availability of such knowledge from websites such as Lynda.com, Khan Academy, Udacity, etc., is ample proof that the price of such materials is approaching zero.

This fundamental knowledge is also already captured in complex and sophisticated software, which means that students do not need to know how to solve the governing equations, though they do need to know how such analytical tools work, at least in principle, and be able to check that the answers that they have received are reasonable. Miscalculation leading to poor design can be fatal.

Studios are intended to give students the opportunity to apply the basics and use the sophisticated tools to solve reasonably complex, real problems. Students will work with industry mentors, in collaborative teams, using reflective practice as a key ingredient to draw out, for themselves, and with guidance, what has been their key learning during the semester. The learning, not the project, is the central activity.

Assessment will need to adapt. Some of our thinking in this area is covered in another paper at this conference.

Finally, the big challenge is to redesign our curricula for these trends. Will curricula eventually be only studios, with online learning supporting each one? Some of them would build basic competencies, such as structural design or design of circuits. Others would extend these skills into more complex applications using advanced computing tools. Other studios would immerse students in even more complex situations, such as resolving transport issues in any of our large capital cities. Other studios would be entrepreneurial, or humanitarian or research-oriented. Many or most of the studios would be conducted with an industry sponsor.

Whatever we do, we need to move away from thinking that teaching standard solutions to standard problems is any kind of preparation for the complex future our graduates will face.

References

AAC&U. (n.d.). "e-Portfolios."   Retrieved 30 Aug 2016, from https://www.aacu.org/eportfolios.

Dowling, D. and R. Hadgraft. (2013). "A Graduate Capability Framework for Environmental Engineering Degree Programs - A Guide for Australian Universities."   Retrieved 25 Mar 2014, from http://www.olt.gov.au/resource-DYD-defining-your-discipline.

Engineers Australia. (2016). "Environmental College - Activities."   Retrieved 30 Aug 2016, from http://www.engineersaustralia.org.au/environmental-college/activities.

Institute for the Future. (2015). "Future Work Skills 2020."   Retrieved 1 Sep 2016, from http://www.iftf.org/futureworkskills/.

Lindsay, E. and J. R. Morgan (2016). The Charles Sturt University Model - Reflections on Fast Track Implementation. ASEE Annual Conference. New Orleans, ASEE. **123:** paper 15487.

Norton, A. and B. Cakitaki (2016). Mapping Australian higher education 2016. Melbourne, Grattan Institute.

Parr, P., W. Yates and K. Yasukawa (1997). The UTS response to the review of engineering education. Australasian Association for Engineering Education Conference. Ballarat.

Prince, M. J. and R. M. Felder (2006). "Inductive Teaching and Learning Methods: Definitions, Comparisons, and Research Bases." Journal of Engineering Education **95**(2): 123-138.

Prior, J. R., C. Arjpru and J. R. Leaney (2014). Towards an industry-collaborative, reflective software learning and development environment. Proceedings of the 23rd Australasian Software Engineering Conference (ASWEC 2014). Sydney, Australia.

Prior, J. R., A. L. Connor and J. R. Leaney (2014). Things Coming Together: Learning Experiences in a Software Studio. Proceedings of the 19th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education 2014 (ITiCSE`14). Uppsala, Sweden.

Prior, J. R., S. Ferguson and J. R. Leaney (2016). Reflection is Hard: Teaching And Learning Reflective Practice in a Software Studio. Proceedings of Australasian Computer Science Week (ACSW'16) Multi-conference. Canberra, Australia.
Schön, D. A. (1983). The Reflective Practitioner: How professionals think in action. London, Temple Smith.
Studio Teaching Project. (2015). Retrieved 7 Sep 2016, from http://www.studioteaching.org/.
UTS. (2014). "UTS Model of Learning." Retrieved 7 Sep 2016, from http://www.uts.edu.au/research-and-teaching/teaching-and-learning/uts-model-learning/uts-model-learning.
UTS. (2015a). "Attributes of UTS graduates." Retrieved 7 Sep 2016, from http://www.uts.edu.au/research-and-teaching/teaching-and-learning/uts-model-learning/attributes-uts-graduates.
UTS. (2015b). "Learning.Futures - What is it exactly?" Retrieved 12 Aug 2016, from https://www.uts.edu.au/sites/default/files/article/downloads/What-is-learning-futures.pdf.

Appendix

Table 3 - Studio-based learning versus problem-based and project-based learning

Category	**Problem based Learning (PbBL)**	**Project based Learning (PjBL)**	**Studio based learning**
Brief Definition	Problem based is based on solving a targeted, medium size problem. May or may not be open ended.	Project Based Learning aims to integrate complex learning through deep involvement in a team based project, often over a full semester, involving design/construction of some object or system.	The studio is a learning community of students, teachers and others such as industry mentors and practitioners, interacting in a creative, reflective process to develop some kind of product, in a physical environment/space that enables collaboration *(STP, 2012)*
Student outcomes	Learning concepts to solve a problem	Integration of prior and emerging knowledge to address a problem	(very) deeply learn to design, to 'engineer'
Activities	Learn and Solve	Learn, Solve, Design, Build	Solve, Design, Build, Reflect
Staff	Tutor	Facilitator	Mentor – coaching and mentoring at team and individual level.
Physical environment	Classroom	Classroom and/or Laboratory	Studio is reconfigurable: group and personal work spaces. 'Studio feels like home'
Theoretical Perspective	Constructivist (weak)	Constructivist (medium to strong)	Constructivist (strong)
Reflection	Maybe	If reflect, then, reflect on what is happening (reflection on action)	Learn to reflect. Move from reflect *on* action to reflect *in* action, Emphasis on personal development.
Task	Usually well-structured for knowledge acquisition	Partly structured	Partly structured to wicked
Skills acquired?	Problems lead to information seeking	Previous subjects plus emerging new knowledge	Previous subjects plus emerging new knowledge
Culture	Team-based knowledge acquisition	Team-based project completion	Collaborative work ethic for product development.
Critique and assessment	Weekly tutor feedback.	Well defined milestones for project review.	Critique should be continual and varied, from peers, mentors, and academics.
Assessment:	Solving the **problem**	Mostly about achieving **product**,	Mostly about achieving **capabilities**,

Pass criteria		some focus on **capabilities**	some focus on **product**
Design capability	PbBL may develop designers	PjBL may develop designers	Studios aim to develop designers