

Designing an online part-time Master of Philosophy with Problem Oriented Engineering

Lucia Rapanotti
Computing Department
The Open University
Milton Keynes, UK
Email: L.Rapanotti@open.ac.uk

Jon G. Hall
Computing Department
The Open University
Milton Keynes, UK
Email: J.G.Hall@open.ac.uk

Abstract

The paper reports on the application of Problem Oriented Engineering (POE) to the design of a highly innovative post-graduate research programme for the Open University, UK, a world leader in supported distance higher education. The new programme, to be launched in October 2009, is a part-time Master of Philosophy (MPhil) to be delivered entirely at a distance, supported by a blend of synchronous, asynchronous and immersive internet and web technologies. POE is a framework for engineering design under development at the Open University. After a brief description of the project and the task at hand, the paper discusses the overall engineering approach taken, key aspects of product design, the mapping between requirements and technology, and the development of one key technological component.

1. Introduction

The Computing Department at the Open University is in the process of developing a new part-time MPhil programme to be delivered at a distance, supported by a blend of synchronous, asynchronous and immersive internet and web technologies. We refer to the programme as the *eMPhil*. The *eMPhil* is innovative in many ways, specifically, in: the adoption of emergent technologies and their use to support the core processes of the programme; the organisation of students around research themes, fostering peer group collaboration to complement the traditional supervisor-student relationship; the potentially very large scale of the target student and supervisor population (the Open University has over 180 thousands students at a distance enrolled on their undergraduate, postgraduate and Continuous Professional Development taught courses). The *eMPhil* project team was faced with a complex socio-technical problem, which required not only the adoption and development of appropriate software systems, but also the definition of new processes and practices, the design and delivery of induction and training activities for staff and students, and the institution of an infrastructure for quality assurance, monitoring and

continuous process improvement. The Problem Oriented Engineering (POE) framework [12], [15], [14] was adopted as a conceptual tool to investigate the problem space and guide product development. The paper reports on the application of POE to the *eMPhil* development to date, its technological infrastructure, and one of its key components, a Second Life [6] campus, called *deep|think*. The paper will focus on the engineering aspects of the project: pedagogical considerations, while paramount to the programme, will not be discussed here, but can be found in [17]. The paper will be relevant to a wide range of practitioners with an interest in developing and/or supporting academic programmes at a distance, mediated by internet and web technologies, from academics who wish to match technology to their needs, to developers and project managers who wish to develop fit-for-purpose solutions.

2. The problem, its context and its requirements

The *eMPhil* was required to meet a number of research objectives, set by the Computing Department as part of their overall research strategy, to:

- enhance and develop the department's provision to their graduate community;
- promote the wider use of technology to support research students at a distance, especially part-timers;
- create cohorts of research students on specific research themes and projects;
- attract more research students, particularly those who feel that they cannot commit the time to a PhD;
- increase the overall amount of research supervision that takes place within the department, particularly involving staff who are not currently supervising research students.

Already the Computing Department has a lively community of residential full-time research (MPhil and PhD) students who, besides having access to their supervisor(s), are given an opportunity to take part in a structured programme of induction and training, facilitated by senior

academics and highly experienced doctoral supervisors. The current programme aims at developing all of the required research skills [5], and also functions as a forum for the student community and as a peers' aid and study group, for which sessions are run on a weekly basis throughout most of the academic year. The programme also fits within the overall university approach to research student induction and training and is compliant with national standards [9] the university has subscribed to. Therefore in addition to meeting the Department's stated objectives, the eMPhil was required to comply with external standards, to provide a comparable induction and training programme for part-timers at a distance, and to create opportunities for them to interact with peers online and to work collaboratively, as well as to develop friendships and a sense of belonging.

3. The project

A study into the feasibility of the undertaking, including the potential market for the product and its main competitors, was conducted at the beginning of 2008. The project started in March of that year, with the project team appointed and some initial planning carried out, coupled with seeking initial approval within the organisation. In June, development started on the technological infrastructure for the programme. The project is still ongoing with the eMPhil due to be rolled out in October 2009.

The eMPhil core project team is composed of four academics, with the first author as project leader. The team drives the whole development and has responsibility for identifying and co-ordinating the activities of other teams, both within and outside the Open University, who make specific contributions to the project.

The following sections provide an account of some key aspects of the development to date, related to the particular approach taken to product design and the development of the eMPhil technical infrastructure.

3.1. Product design

The project leader decided to adopt the Problem Oriented Engineering (POE) framework [12], [15], [14] as a tool to investigate the problem space and guide product development. POE is an emergent framework for software engineering, and more generally, engineering design, under development at the Open University. From the eMPhil project's perspective, the framework was attractive in offering a blend of conceptual tools to tackle problem complexity, with the added bonus that POE expertise exists within the project team and in the Computing Department. Of particular attraction to the project was POE's problem solving view of engineering design, which extends beyond software to processes and organisational issues, as well as POE's ability

to accommodate diverse practices within a unified conceptual framework. In the following, we briefly recall the main elements of POE and describe its application to the eMPhil project. It is not our intention to provide a comprehensive presentation of POE, which can be elsewhere: [13] gives a gentle introduction to POE, while an extensive bibliography can be found by visiting the POE home at [4]. POE sees engineering design as the creative, iterative and often open-ended undertaking of conceiving and developing artefacts such as products, systems and processes (inspired by [18], [11]). In POE's view, engineering design includes:

- the identification and clarification of a recognised need, the requirement;
- the understanding and structuring of the real-world context with which the engineered system will interact;
- the specification of a design for a solution that can ensure satisfaction of the recognised need in context;
- and the construction of arguments, convincing for all validating stake-holders, that the engineered solution will provide the functionality and qualities that are needed.

The basic building blocks of POE are problems, transformations and justifications, jointly called the *POE triad*. A *POE problem* is a collection of named descriptions, one each for the requirement (the identified need), the solution (the artefact subject of design), and the context (that part of the world where the solution will be located to meet the requirement). Such descriptions hold knowledge about the problem and its parts. As they capture knowledge about the real world, they can be of various natures: from precise and formal to vague and inconsistent, from textual to pictorial, etc. *POE transformations* are design steps, changing problems into other problems and giving rise to a step-wise problem solving process. *POE justifications* support stake-holder validation, and are a tool for managing certain downside risks, those of solving the wrong problem and/or designing an inadequate solution.

POE gives equal relevance to both artefact design and its validation. It also advocates early validation as a tool for managing risk, hence to be applied throughout development. These observations, coupled with the transformational nature of problem solving in POE, result in particular design structures, as illustrated in Figures 1 and 2, which give, respectively, a single design step, and a snapshot of the POE design space for the eMPhil after the first ten months of production. The figures illustrate (part of the) POE graphical notation used to capture an abstract representation of the design space. Actual artefacts and their descriptions are kept separate, but are indexed through the graphical notation. Briefly (more detail can be found in [13]), design problems relate to the design of the product itself, i.e., the eMPhil and its various components; each problem includes a representation of context (rectangles), requirement (oval), and solution

design problems

validation problems

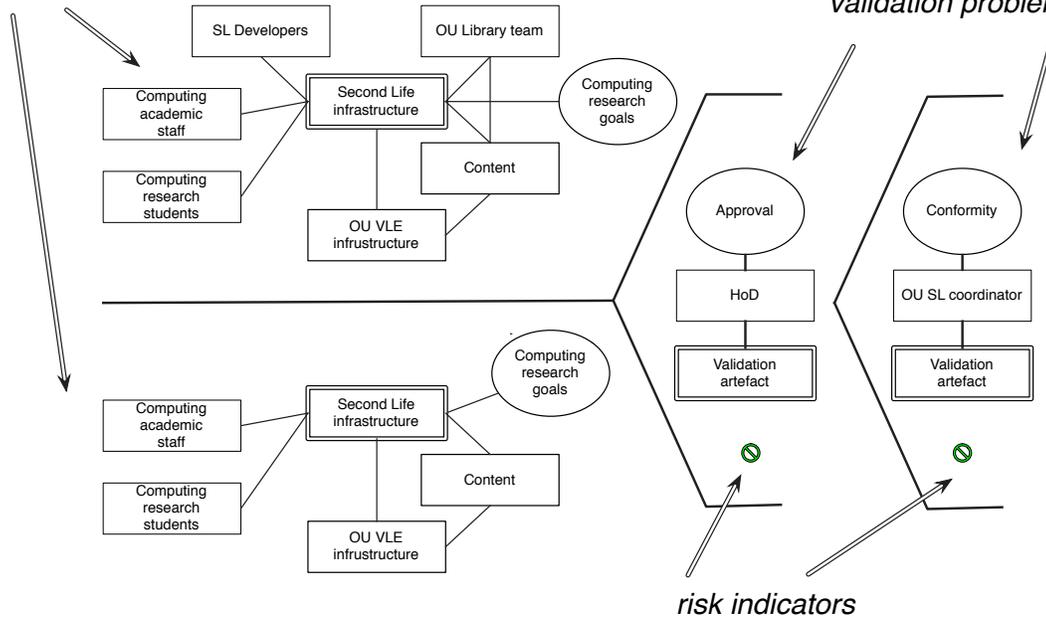


Figure 1. A POE design step

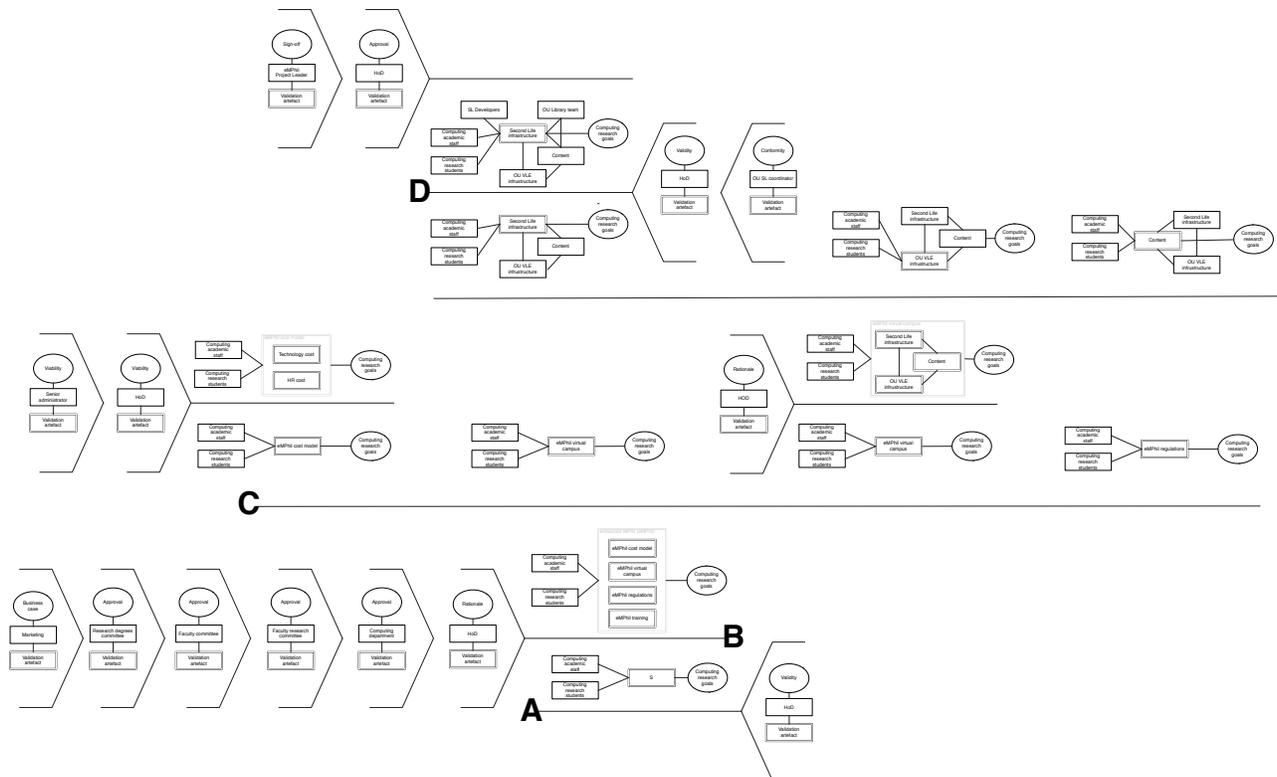


Figure 2. Snapshot of the first ten months of eMPhil development

under design (decorated rectangles). Design problems are transformed into other design problems (horizontal lines represent transformations) as engineering design unfolds, the resulting structure being that of a design tree. To the side of the transformation lines are validation problems whose solution provides a justification of adequacy of the steps.

Figure 1 illustrates one such transformation, capturing the design step of involving third-party developers and librarians in the design of the Second Life campus; a step validated by the Head of Department, responsible for releasing appropriate resources, and the Second Life co-ordinator at the Open University, responsible for the organisational policy on the use of Second Life. As already mentioned, POE sees validation and design as equal partners, so both design and validation spaces have to be explored and represented as part of the problem solving process. Validation is the process by which relevant stake-holders give their approval as to the adequacy of a design step, for instance signing off some stated assumptions on the problem context, or the choice of a particular solution architecture. This sign-off process is a way of managing development risks, as lack of approval may signify that, say, stated domain assumptions are incorrect or a proposed solution is unfeasible. Risk indicators within validation problems provide summarise the progress made in the solution of the validation problems, though colour-coding.

Figure 2 gives an indication of the the ‘shape’ of the design tree after ten months’ of production: based on the strategic goals set by the Computing Department, an initial problem exploration (transformation A at the bottom of the figure) was carried out with the Head of Department as the main validating stake-holder. This led to an initial proposal for the solution, the eMPhil as a product, which had to go through a rigorous approval process (validation problems to the left of transformation B) involving stake-holders at various levels of management throughout the university. From the initial solution architecture, a number of sub-problems were then identified (transformation C) each addressing complementary aspects of the solution, such as the design of its technological infrastructure, a related cost model, a programme of user induction and training, a system of monitoring and evaluation, etc. Each sub-problem was then taken forward through further transformations and related stake-holder validation, with the design problems at the top representing either solved sub-problems or open problems in the process of being addressed. The problem related to the development of the Second Life campus is the design problem above transformation D: the problem was solved with validating stake-holder the Head Of Department, as the customer funded the development, and the project leader, who signed-off on the agreed contract.

3.2. Technology Summary

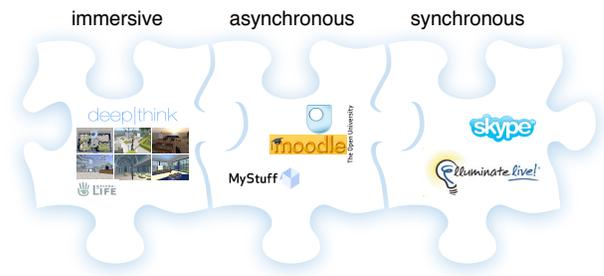


Figure 3. Overview of the technologies adopted

The vision of the eMPhil infrastructure was to provide a rich and flexible Virtual Learning Environment (VLE), which could accommodate a large variety of users’ need and attitudes to technology: a virtual space where research students could come together for both serious business and fun, both aspects being essential to the development of a vibrant research community. This was deemed appropriate for a research programme, which, contrary to a taught course, is by and large driven by students and their supervisors on an individual basis, and carries on over a long period of time. Our mapping of requirements and domain processes onto technology, led to the following blend of technologies (summarised in Figure 3). Asynchronous technology, including Moodle [3] and a custom-made ePortfolio system, was chosen the basis of the main eMPhil portal, to provide overall access and support the programme and the community with materials organised around research themes and key processes in the degree, such as induction, probation, monitoring, etc. Synchronous technology, including Skype [7] and Elluminate Live! [2], was chosen for day-to-day supervision, group working, as well as for monitoring and assessment, e.g., for probation and mock viva voce examinations, remote slide shows, etc. Finally, immersive technology, namely Second Life [6], was chosen to increase the sense of participation for remote users and to support socialisation [10], [16], as well as complement the other synchronous technology for supervision and group working.

3.3. deep|think campus

A Second Life campus was commissioned from a third-party software development company, with a track record in Second Life educational sites for UK academic institutions. Some expertise in the use of Second Life for education also existed within the eMPhil team. This combined expertise was sufficient to draw an initial generic specification as the baseline for the contract. It was, however, acknowledged that such a specification was very generic and the project highly experimental, so that further refinement or even some

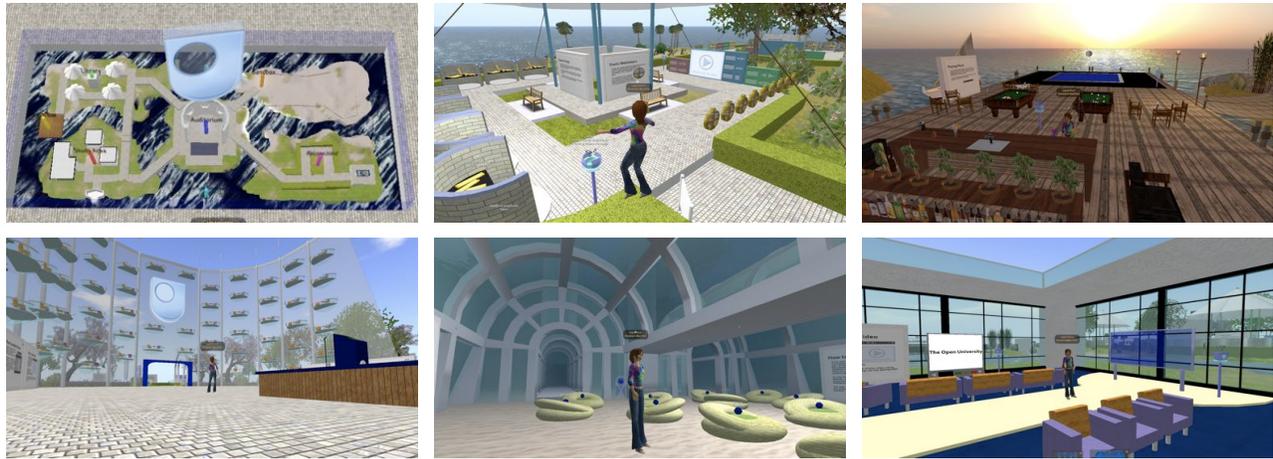


Figure 4. Snapshots of [deep|think](#). Clockwise from top-left: 3D map of [deep|think](#); Library Search Area; Beach Bar; Students' Common Room; Underwater Theatre; Main Amphitheatre

rethinking of the original requirements might be necessary. The eMPhil project leader and her counterpart on the software development team, agreed that an Agile approach [1] to development should be taken, with frequent customer-developer interaction and short incremental iterations driven by user requirements. The time scale for the project was rather compressed with 90% of the development carried out within 6 weeks.

True to the POE stake-holder centric view of design, the project leader was driven to consult a wide constituency of expert stake-holders in order to refine the original specification into a successful product. Consultation took various forms, from a competition within the Computing Department to choose the name of the campus, [deep|think](#), to the involvement of a team of librarians from the Open University Library to come up with innovative ways to organising and offering library services in a 3D virtual world. Software developers brought their own design flare to the project, and both teams work together to achieve a distinctive look and feel. A recurrent theme in the interface is lightness and openness, with few Open University landmarks scattered around the campus, but without reproducing the real-life equivalent. Star Wars [8] provided some inspiration for the main amphitheatre, and both land, sky and water¹ were used imaginatively to create a variety of social and collaborative spaces. Project iterations occurred on a weekly basis, including inworld² sessions as the means for discussing requirements, and for customer's inspection and validation. A more detailed description of the virtual campus and related pedagogy can be found in [17].

4. Early evaluation and conclusion

The experience of using POE as a tool for artefact design and project management on the eMPhil has been very encouraging. At the start of the project, we had to acknowledge that a gap existed between POE as a theoretical framework and its practical application, and a number of practices had to be adopted to fill in the gap. Particularly, despite the many requirements and software engineering tools available, we could not identify an appropriate subset which could in concert support the framework in its entirety—most focus on specific aspects of software development or specific modelling techniques. Also, the idea of switching between a disparate set of tools manually was deemed too cumbersome and time consuming for the project. Therefore, we resolved to a low cost, low tech approach consisting of keeping a manual repository of the various artefacts on a dedicated computer, with automatic hourly backup and a naming convention to cross-reference artefacts in the repository and elements of the POE graphical models. The project leader also kept a project diary, to log any difficulty or other issue raised by the adoption of POE. Overall, the experience so far has clearly indicated that the conceptual tools offered by POE, coupled with the practices introduced for the eMPhil, were able to cater for all relevant aspects of the project. The POE graphical notation provided a powerful summary of the development, with all critical decision points clearly exposed. It was also considered an effective communication tool: its relative simplicity and abstraction allowed even non-technical stake-holders, like senior managers in academic and academic-related units, to grasp the essence of the project with very little explanation required. The abstraction level provided by POE was able to accommodate the diverse practices which were required to tackle the different design and validation problems, while still allowing the essen-

1. Second Life worlds usually take the form of sunny islands.
2. In immersive systems, the term 'inworld' is used instead of 'online'.

tial tracking of the activities carried out and the artefacts produced, and showing that the light-touched approach to engineering that POE affords makes it a flexible and versatile tool. Clustering and indexing the various project artifacts through the graphic notation allowed the team leader to locate information efficiently, and dip in and out looking at information at various levels of abstraction. However, the lack of automated tools was a real drawback. In order to be true to the approach, a great deal of discipline and care was required of the project leader in manually recording artefacts and their transformations, something which could not be normally expected in a commercial development. On the other hand the need to combine the scientific investigation into the use of POE with the demands of a live project including concrete milestones and deliverables, meant that the whole approach had to be distilled and streamlined to be of practical use.

In terms of project progress, various production tasks have been successfully completed. The basic technological infrastructure is now in place, including both the [deep|think](#) development and a Moodle site customised for the programme. The next priority is to work on content, which will require the project team to interact with and co-ordinate the work of a small number of academic teams tasked with providing specific research materials and resources within identified areas of research in Computing, as well as generic research skills development. Materials will vary from traditional texts, to dynamic content, such as podcasts and animated slide shows, to online synchronous group working sessions. Trials of the technology are also being undertaken both for evaluation and training purposes.

An evaluation programme for the eMPhil is currently being designed, aimed to evaluate the suitability and effectiveness of the technology offered to support the main processes of the programme, and to collect reliable data on workload for both students and supervisors to inform the workload planning task within the Computing Department and a scalable cost model for the programme.

Acknowledgments

Many thanks go to the other members of the eMPhil production team at the The Open University, namely Leonor Barroca, Marial Vargas-Vera and Shailey Minocha, as well as our Open University librarians, Non Scantlebury and James McNulty.

References

- [1] Agile Alliance. <http://www.agilealliance.org/home>. Last accessed: 11th February 2009.
- [2] Elluminate. <http://www.illuminate.com>. Last accessed: 11th February 2009.
- [3] Moodle. <http://moodle.org/>. Last accessed: 11th February 2009.
- [4] POE Design Home. <http://www.solvehappy.com/>. Last accessed: 30th January 2009.
- [5] Research Degree Skills. <http://phdskills.open.ac.uk/>. Last accessed: 11th February 2009.
- [6] Second Life. <http://secondlife.com/>. Last accessed: 11th February 2009.
- [7] Skype. <http://www.skype.com>. Last accessed: 11th February 2009.
- [8] Star wars. <http://www.starwars.com/>. Last accessed: 11th February 2009.
- [9] UK GRAD, Joint Skills Statement of Skills Training Requirements. <http://www.grad.ac.uk/jss/>. Last accessed: 11th February 2009.
- [10] Edward Dieterle and Jody Clarke. Multi-user virtual environments for teaching and learning. In *Encyclopedia of multimedia technology and networking (2nd ed)*. Hershey, PA: Idea Group, Inc., 2008.
- [11] ECSA. Definition of terms to support the ecsa standards and procedures system. Technical Report G-04, draft 6, Engineering Council of South Africa, Standards and Procedures System, January 2003.
- [12] Jon G. Hall and Lucia Rapanotti. Assurance-driven design. In *Proceedings 3rd International Conference on Software Engineering Advances*, Sliema, Malta, October, 26-31 2008. IARIA, IEEE Computer Society Press.
- [13] Jon G Hall and Lucia Rapanotti. Requirements analysis in context with poe design. Technical Report TR2009/2, The Open University, 2009.
- [14] Jon G. Hall, Lucia Rapanotti, and Michael Jackson. Problem oriented software engineering: A design-theoretic framework for software engineering. In *Proceedings of IEEE Software Engineering Formal Methods (SEFM 2007)*, London, UK, September 2007.
- [15] Jon G. Hall, Lucia Rapanotti, and Michael A. Jackson. Problem oriented software engineering: Solving the package router control problem. *IEEE Transactions on Software Engineering*, 34(2), March/April 2008.
- [16] Guido Rößling Martin Leidl. How will future learning work in the third dimension? In *Proceedings of ITiCSE'07*, Dundee, Scotland, June 23–27 2007. ACM Press.
- [17] Lucia Rapanotti, Leonor M. Barroca, Maria Vargas-Vera, and Shailey Minocha. deepthink: a second life campus for part-time research students at a distance. Technical Report TR2009/1, The Open University, 2009.
- [18] G.F.C. Rogers. *The Nature of Engineering*. The Macmillan Press Ltd., 1983.