

The Language of Capstone: A Translator

Dan Harel, John Kaemmerlen, Reginald Rogers, James Vallino, Elizabeth DeBartolo
Rochester Institute of Technology

“But that doesn’t apply to MY project.” “We can’t participate in multidisciplinary projects because our work is too specialized.” In an academic environment, where we may spend all day surrounded by people who know the same sorts of things, use the same terminology, and do the same sort of work, it can be easy to fall victim to silo mentality. The real world, however, is not quite so neat. As capstone design instructors it is our job to prepare students for a world where electrical engineers and industrial designers work together to design children’s toys; or where software and biomedical engineers create surgery simulators. This often means providing project opportunities where students collaborate across disciplines. In order to function effectively in this world, students and faculty need a common understanding of each discipline’s respective design process. This paper highlights several different disciplinary design processes and compares the timing and tasks across disciplines and across design process phases.

Keywords: Interdisciplinary, multidisciplinary, collaboration,

Corresponding Author: Elizabeth DeBartolo, eademe@rit.edu

Introduction

The goal of this paper is to outline several types of design projects in order to facilitate (1) collaboration on multidisciplinary teams and (2) the interpretation of published work in the literature between different types of capstone projects. At the Rochester Institute of Technology (RIT), we offer a variety of single- and multi-disciplinary capstone design courses, with regular collaboration between them. The collaborating students tend to work as multidisciplinary or interdisciplinary teams according to the definitions of Pak and Choi.¹

Real world challenges require input from multiple disciplines to solve. After graduation many engineering students will find themselves assigned with people from outside their own discipline to work on complex problems requiring collaboration. The work may allow individuals to work in parallel but each must still understand how their colleagues’ work fits in with their own. The work may be more integrated, where people from different backgrounds work closely together and must overcome “language” differences due to their own disciplinary jargon. In either case, it is useful to have a translator that helps each participant understand what the other is doing and how their work relates.

Design Process and Project Types

This work is based the general underlying design process that applies to a wide range of different disciplines, which can be broken down into roughly five distinct phases. We (1) define the problem, (2) conceive of a solution concept(s), (3) develop the details of the

most promising solution, (4) implement and test the solution, and finally (5) deliver the solution to the client and disseminate the results. Throughout the process, we report back to stakeholders with progress updates.

While the design process is fairly general, different types of projects require the use of different tools to complete the work, and the different phases may have different durations. The types of projects considered here are engineering hardware design, software design, product design, and industrial process design – both with and without implementation. These categories were chosen because they are represented at our institution and we have experience collaborating between these disciplines; other types of projects that could also be considered are hardware design with no build, hardware design beta prototypes, or civil engineering design with no build.

Engineering: Hardware Design

We start with an outline of the typical process and deliverables as it applies to hardware-based engineering capstone projects.² We base this on the fact that a recent study indicated that 54% of respondents report a prototype as the final project deliverable.³

1. Problem Definition, 10%: Identify and interview stakeholders to establish use cases, requirements, and constraints. Write a problem statement, begin project planning and risk management.
2. Concept Generation & Selection, 13%: Perform functional decomposition, brainstorm solutions, create and evaluation system-level design concepts,

perform preliminary feasibility analysis. Refine requirements and articulate a high-level final design proposal. Continue to plan and manage risk throughout this and all remaining phases.

3. Detailed Design, 30%: Refine and add detail to proposed design, continue to perform additional feasibility analysis in order to support design decisions. Deliver final design package, including detailed drawings, schematics, fabrication plans, bill of materials, and test plans. Formal report-out on preliminary design at midpoint of phase.
4. Build & Test, 33%: Construct and test elements of the full system. Progress through full system integration and testing. Update design documentation as the design changes.
5. Deliver & Disseminate, 13%: Address any remaining issues and prepare for project handoff. Deliver the final prototype to the client, including supporting documentation and design archive. Disseminate project work: paper and poster.

Industrial Engineering: Process Design, With Implementation

A process design project is contrasted here with a hardware design project. Rather than design a tangible electromechanical system, these teams design a process and implement it in a manufacturing (or similar) environment. They may follow the six-sigma “DMAIC”⁴ (define, measure, analyze, improve, control) framework or the Toyota A3 Problem Solving approach.

1. Problem Definition, 10%: Same as prior example.
2. Concept Generation & Selection, 13%: Same as hardware design example. Students may create a process flow diagram, a value stream map, or a fishbone diagram, to represent their system in a way that enables meaningful analysis.
3. Detailed Design, 30%: Same as hardware design example. The final design package for a process innovation project typically includes floor layouts, process flow charts, discrete event simulation models, user interface analyses, etc.
4. Build & Test, 33%: Implement and test elements of the full system, working up to full system implementation. The human interface very often becomes critical at this point. Students work with local management and client team members to experiment with them, so the process designs can be evaluated and improved with client buy-in. This is sometimes accomplished through a series of small “kaizen” events.⁵
5. Deliver & Disseminate, 13%: Address any remaining issues and prepare for project handoff. Deliver the final process to the client, including supporting documentation. Disseminate work.

Industrial Design: Product Design

This process mirrors the engineering hardware design process, with emphasis on Design Thinking principles.⁶

1. Research & Design Brief, 10%: Establish project context and identify the user, target market, and use environment for the project. Conduct benchmarking, and identify business needs and potential technologies for the project.
2. Product Ideation & Concept Development, 13%: Conduct initial ideation, produce 2D hand sketches & CAD drawings, develop initial (diverging) physical models, develop test criteria and observation methods, perform initial product testing and refine design.
3. Model Building & 2nd Product Testing, 30%: Create CAD renderings for selected design and build full-scale appearance model. Develop 2nd testing criteria and refine design based on product testing results.
4. Refine & Finalize Design Direction, 23%: Start constructing and testing elements of the full system. Progress through system integration and testing. Update design documentation as problems arise and design changes. Continue to plan and manage risk and problems.
5. Project Completion, 23%: Address any remaining issues and prepare for project handoff. Deliver the final prototype to the client, including supporting documentation. Disseminate project work.

Software Engineering: Software Design

These capstone projects have a software deliverable, as opposed to embedded systems development. These types of software-only projects will more often follow an agile software development methodology. RIT software engineering teams use a process based on OpenUP as the strategic level process structure.⁷

1. Inception, 15%: Work with product owner to gain an understanding of the problem domain and scope. Gather requirements in terms of user story statements placed in the Product Backlog. Develop system level requirements that lead to a proposed architectural design and technology selection. This phase primarily addresses high-level project and requirements risks.
2. Elaboration, 15%: Implement spike solutions to explore technology choices. Implement architecturally significant user stories to validate the proposed architectural design. This phase primarily addresses architectural risks.
3. Construction, 60%: Build out the product in two week sprints using Scrum practices.⁸ Each sprint delivers incremental value to the project sponsor with each implemented user story being a vertical

slice of completed functionality. User stories are sized to allow the team to meet the goal of no work-in-progress left at the end of a sprint. Prior to being moved into the Sprint Backlog, the development team and product owner refine the user story with Acceptance Criteria and Solution Tasks that evolve the understanding of the requirements and detailed software design. User stories are implemented in the priority order set by the product owner. Stories can be added, deleted, or re-ordered on a sprint-by-sprint basis. Each sprint closes with a retrospective for continuous process improvement.

4. Transition, 10%: Deployment of the final product along with transition documentation, such as user manuals, installation/deployment instructions, and finalized design documentation.

Chemical Engineering: Process Design Without Implementation

RIT's Chemical Engineering program is a design-only capstone. This is not uncommon for Chemical Engineering programs, where a typical project is a plant design: difficult to prototype.

1. Problem Definition, 7%: Receive project problem and assess current situation. Lay out potential process flow. Identify safety hazards of chemical process and define protocols needed to mitigate hazards. Begin project conceptual design phase.
2. Conceptual Design, 13%: Prepare process flow diagram outlining major components needed to accomplish desired outcome. Perform hand calculations on paper and in Excel to assess physicality of solution. Incorporate proposed solution in CHEMCAD process simulator to compare hand calculations to simulation results. Present conceptual design in a 5-minute briefing.

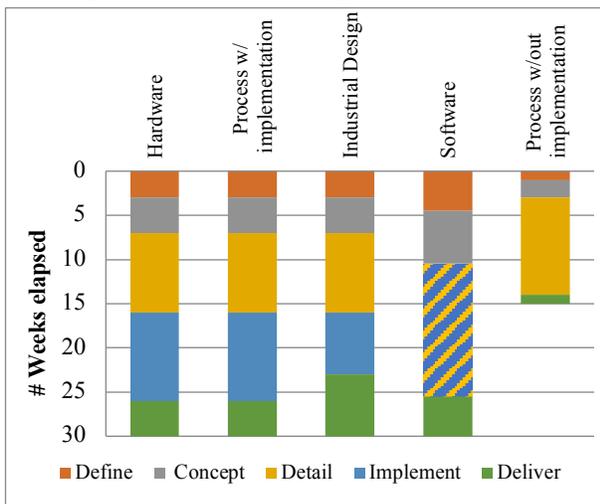


Figure 1. Relative timing of the different design phases across a variety of project types.

3. Final Design, 73%: Refine and add detail to proposed design, continue to perform additional feasibility analysis in order to support design decisions. Receive instruction on and create proper development of Piping & Instrumentation Diagrams (P&ID). Discuss proper sizing of heat exchangers, pipes, valves, tubes, fittings, and pumps. Finalize all calculations to meet objective of project. Formal report-out on preliminary design, which includes short economic analysis and recommendations, at midpoint of phase.
4. Deliver & Disseminate, 7%: Formal presentation (~40 minutes) describing project goal and recommendations. Be able to suggest possible steps that could be taken to improve process design.

Observations

Figure 1 is a side by side comparison of the duration of each of the design phases, by discipline. Note that the plant design portion of RIT's Chemical Engineering capstone is only 15 weeks long, and does not include a build & test phase, but the percentage of time spent on the remaining phases is comparable to those of the other disciplines. Also note that the Software Engineering Construction phase includes elements of both Detail and Implementation. Since our multidisciplinary program already includes projects for traditional hardware design & build, process design, and collaboration with Industrial Design students working on product design, those three types of projects' phases are well aligned.

When the tasks and deliverables for different types of projects are compared by phase (Table 1), some clear similarities emerge, if with slightly different language.

- Understanding the customer may be referred to as user stories, use cases, or user context.
- Students may manage risks or hazards.
- To lay out the high level solution path, a student may create a functional decomposition, system architecture, process map, or value stream map.
- Feasibility analysis may take the form of hand calculations, simulation, testing concept models, or implementing spike solutions.
- Designs must be detailed, with part/assembly drawings, flow charts, floor layout, piping & instrumentation diagrams, circuit schematics, deployment manuals containing all the instructions necessary to implement the design.
- Students write test plans, acceptance criteria, or user test criteria prior to implementing solutions.
- Implementation may include building prototype hardware, implementing a user story, or conducting kaizen events on a factory floor.

Conclusion

There are similarities in the duration and type of work done by all the capstone design project types outlined here. Using this information, a capstone instructor could lay out a roadmap for students from multiple different disciplines who want or need to work together on a project, as well as facilitate communication between students from different disciplines who struggle with jargon. Future work will include compiling and presenting case studies to illustrate how these collaborations have worked, including timing, work breakdown, and deliverables

References

1. Choi, Bernard C K; Pak, Anita W P. (2006). Multidisciplinarity, interdisciplinarity and transdisciplinarity in ... *Clinical and Investigative*

Medicine, 29, 315-364.
 2. Ulrich, K.T. and Eppinger, S.D. (2008) *Product Design and Development* (5th Edition).
 3. Howe, S., & Poulos, S. L., & Rosenbauer, L. M. (2016, June), *The 2015 Capstone Design Survey: Observations from the Front Lines*, ASEE Annual Conf., New Orleans, Louisiana.
 4. Borrer, Connie M.. (2009). *Certified Quality Engineer Handbook* (3rd Edition). American Society for Quality (ASQ).
 5. Baker, B. (2004). Lean lexicon. *Quality Progress*, 37(7), 96.
 6. Lockwood, T. (2009). *Design thinking: integrating innovation, customer experience, and brand value*.
 7. *OpenUP*. Retrieved Jan 6, 2018, from epf.eclipse.org/wikis/openup/
 8. *What is Scrum?* | *Agile Alliance*. Retrieved Jan 6, 2018, from www.agilealliance.org/glossary/scrum

Table 1. Side-by-side comparison of discipline-specific tasks and deliverables using native terminology.

	Hardware	Process w/ implementation	Industrial Design	Software	Process w/out implementation
Define	<u>Problem Definition</u> interviews, use cases, requirements, plan, manage risks	<u>Problem Definition</u> interviews, use cases, requirements, plan, manage risks	<u>Research & Design Brief</u> establish context, identify user & target market, benchmark, identify business needs	<u>Inception</u> user story statements, system level requirements, address high level risks, design architecture	<u>Problem Definition</u> assess situation, identify hazards, define mitigation protocols
Concept	<u>Concept Generation</u> functional decomposition, brainstorming, system level concept, feasibility analysis, refine requirements	<u>Concept Generation</u> process flow diagram, fishbone diagram, value stream map, system level concept, feasibility analysis, refine requirements	<u>Product Ideation</u> conduct ideation, create sketches & CAD, develop test criteria, perform initial testing	<u>Elaboration</u> implement spike solutions, validate system architecture & address risks	<u>Conceptual Design</u> process flow diagram, outline major components, hand calculations, simulation
Detail	<u>Detailed Design</u> feasibility analysis, CAD, schematics, fabrication plans, bill of materials, test plans	<u>Detailed Design</u> feasibility analysis, CAD, bill of materials, floor layouts, process flow charts, simulation models, test plans	<u>2nd Product Testing</u> CAD renderings, full scale appearance models, develop test criteria, perform testing	<u>Construction</u> size user stories to fit sprint, write acceptance criteria, implement user story, reflect and improve	<u>Final Design</u> feasibility analysis, economic analysis, size mechanical elements, CAD/P&ID
Implement	<u>Build & Test</u> build & test system elements, integrate full system and test	<u>Implement & Test</u> plan implementation with client, conduct kaizen events	<u>Refine & Finalize Design</u> build & test system elements, build full system, refine design		(n/a)
Deliver	<u>Deliver & Disseminate</u> deliver prototype, write paper, present poster	<u>Deliver & Disseminate</u> deliver process w/documentation, write paper, present poster	<u>Project Completion</u> Deliver prototype, supporting documentation, disseminate work	<u>Transition</u> Deploy final product, write user manual and document work	<u>Deliver & Disseminate</u> final presentation, make recommendations