Merging the Cultures of Design and Engineering: A Case Study

Julie Baca[1](✉), Daniel Carruth[2], Elijah Davis[2], and Daniel Waddell[2]

[1] U.S. Army Corps of Engineers, Engineering Research and Development Center,
Vicksburg, MS 39180, USA
`julie.a.baca@usace.army.mil`
[2] Center for Advanced Vehicular Systems, Mississippi State University,
Starkville, MS 39759, USA
`{dcarruth, ecd157, dmw277}@msstate.edu`

Abstract. This paper presents a case study of a web application development project, in which user-centered design (UCD) was successfully integrated in an environment with strong historical and cultural roots in traditional engineering. The case study details the key factors that led to the success of the project in a deeply traditional risk-averse engineering culture, and how the lessons learned apply beyond its scope. Key factors included a design team with experience in the worlds of both design and engineering, a careful selection of UCD methods, and finally a management structure that embraced cultural change.

Keywords: User-centered design · Usability · Prototyping

1 Introduction

In the revised edition of his classic text, *The Design of Everyday Things*, Donald Norman added an entire chapter devoted to "Design Thinking" and another to "Design in the World of Business" [1]. In these additions, from the vantage point of over 30 years of experience since the original text, he explores anew the fundamental differences between the goals and practices of design culture and those of the cultures of traditional engineering and business. Design thinking, he contends, is by nature iterative, exploratory, circular, open-ended, and most importantly, human-centered, with a focus on meeting human needs and desires. A human must always be in the exploratory loop to quickly test and refine ideas as they are generated. This approach inherently embraces risk, and the likelihood of failure in early design iterations.

In contrast, traditional engineering and business methods are linear and unidirectional with clearer beginnings and endings, a more limited ability to backtrack and refine ideas, and, of critical importance, are risk-averse and far less human-centered, focused instead on product qualities such as reliability and verifiability. Successful product design, Norman argues, requires a marriage of the best of both of these cultures. These ideas have been well understood and followed by researchers and practitioners in the fields of user interface (UI) and user experience (UX) design for decades. Why, then,

should we still be discussing them? The answer lies in a statement Norman makes in the final chapter of the book:

"Technology changes rapidly; people and cultures change slowly."

Although the need for human-centered design in software development has been understood in the design community for over 30 years, its inclusion in actual practice remains, at best, hit or miss. Software continues to be developed predominantly from within the cultures of traditional engineering and business, and, cultural change, as Norman points out, is grindingly slow. Beyond simply changing methods or practices, it requires a fundamental shift in thinking, or at the very least, loosening the grip of deeply held beliefs about the appropriateness of the underlying paradigm for all aspects of product design. Ideally, it is an evolutionary shift that does not discard traditional engineering methods where appropriate, but includes and integrates those of human-centered design where needed.

This paper presents a case study of a web application development project, in which user-centered design (UCD) was successfully integrated in an environment with strong historical and cultural roots in traditional engineering. Although the methods and techniques used are not novel in the UCD world, the effort merits study to gain insight into why it succeeded in such a deeply traditional, risk-averse engineering culture, and how the lessons learned apply beyond that scope. In overview, its success can be attributed to three key factors: (1) the "multicultural" composition of the design team, i.e., all with prior experience in or exposure to the methods of both design and engineering, (2) the particular choice of UCD methods used in the project, and finally, (3) a management structure that embraced cultural change.

2 Case Study

2.1 Design Team

This case study examines the integration of UCD in the software lifecycle for a product called the Ground Vehicle Interface (GVI), developed by the U.S. Army Corps of Engineers (USACE) Engineering Research and Development Center (ERDC). GVI was unusual from the outset in that it was conceived by management with the specific goal of providing intuitive web access to a suite of analytical tools for a wide range of high performance computing users, with no requirement for extensive knowledge of the underlying modeling and simulation capabilities. The leadership focus on the needs of users exhibited a major advance in cultural understanding over previous generations, regarding the importance of ease of use to product acceptance. That emphasis alone set the course of the project in a successful direction. Attaining this goal, however, would further challenge and ultimately alter the established software culture and practices. The first steps toward change began when the development team created an initial version of the software. While it was functionally robust, both developers and leadership recognized the need to include available expertise in user-centered design and usability.

The UCD team was led by a trained usability specialist with over a decade of experience in User Interface (UI) and User experience (UX) design, as well as a strong previous working background in software development. The team also included a cognitive scientist with dual training and experience in both cognitive science and computer engineering, a graphic designer experienced in delivering designs in an engineering environment, and a computer science graduate student, focusing on human-computer interaction. The multidisciplinary nature of the team embodied an atypical blend of design thinking and skills with experience in an engineering culture. This created a team mindset that could adapt to existing cultural practices without compromising core design values.

Critical to the success of the project were the selection of UCD methods and the specific manner in which they were applied that recognized the limitations of the culture while seeking to push beyond them. These included: (1) usability walkthrough, (2) user advisory panel, (3) iterative prototype review with formative usability testing, and (4) summative usability testing.

2.2 Usability Walkthrough

Ideally, UCD teams begin to gather usability requirements simultaneous to the development team gathering functional requirements. In traditional engineering cultures, however, UCD teams are rarely included in a software project in the earliest stages. In this case, again, the support of management enabled the inclusion to occur relatively early in the lifecycle. Although its inclusion occurred after an initial UI had been developed, it was still well before any testing or production cycles were eminent.

Once included, the design team had to determine the best course of action to set the foundation for a good UI design. In theory, restarting the cycle to gather usability requirements with a representative set of users might seem the best choice. But this ignores several unspoken cultural issues that can present barriers to acceptance of the design team by the developers. Requirements gathering is time-consuming, does not offer a rapid actionable result, and thus, developers can quickly lose patience as their own deadlines loom. Also, while they may recognize deficiencies in the existing UI, developers are not educated in UCD, so do not fully understand why, and thus how, to get from point A (the existing UI) to point B (the improved UI). Frustration builds over the unknown and the unpredictable, both anathema to the traditional linear deadline-driven model; in an iterative UCD design process friction with the design team can result. It is a classic example of the conflict between the linear thinking necessary for sound engineering, and the cyclical nature of UI design.

Therefore, the first action of the design team was to simply conduct a usability evaluation of the initial UI. This action was chosen because it can be done fairly quickly, i.e., as little as 1–2 weeks rather than months, and yields immediate visible and tangible results. More importantly, it rapidly educates the development team and offers insight about the nature of the problems and how to solve them. The evaluation was conducted using a variant of a heuristic method, the "usability walkthrough" [2–5]. This technique typically identifies potential usability problems in a user interface in the design stage. It entails a usability specialist (or a team of specialists) who "walk through" a set of interactive exercises with a user interface to identify potential

problems, based on known usability heuristics and principles. The technique has been reported to identify a significant percentage of usability problems in the design stage [5]. This does not replace formal testing with users, which can be conducted later in the development cycle. Of particular importance to this effort, it also helps developers to see the possible need to revisit requirements gathering with users with respect to usability. Also significant, the UCD team employed mockups and sketches to show possible solutions to the problems, tangibly illustrating a "Point B" goal for the linear developer mindset to target. *Understanding and insight reduce frustration and thereby, increase developer patience with the process.*

Heuristics. A set of general usability heuristics, such as those defined in [6], are considered in an evaluation of any user interface. Since GVI is a web application, a more specific set of heuristics and guidelines were also considered. Those for general websites are first, followed by those specific to web applications. A comprehensive set of web usability guidelines are given in [7–13]. Those relevant to GVI include:

(1) Clearly establish identity and purpose on the home or opening page [7, 8, 10]
(2) Place all critical information and interactions "above the fold" before the user must scroll [7–10]
(3) Use strong focal points and a hierarchy of visual cues to draw the user into the site or application and guide the user through the interaction space [8, 14]
(4) Provide persistent or "sticky" navigation allowing users access to the main site navigation bar from anywhere within the site [8, 10, 15].

Distinctions between websites and applications that have important usability implications are described in [16–18]. The main purpose of a website is to present content to the user with limited interactivity beyond search engines or contact forms. In other words, the overall purpose of a website is more informational than interactive. Conversely, the purpose of a web application is to support users interactively performing frequent and/or repetitive complex tasks. This latter distinction requires a greater emphasis on the following elements and activities:

- Simplicity in visual design to counterbalance complexity of interactions
- Clearly defined user profiles, including novice and expert roles
- Detailed task analyses for each category of user profiled.

Walkthrough Results. Highlights of issues noted on the home page that intersect both the website and web application usability heuristics are presented here. Detailed results of the walkthrough are given in [19].

Home Page Analysis. The home page is the entry point into the site and thus critical to establishing ease of use. The walkthrough identified three main usability issues on the home page, shown below in Fig. 1, in regard to guidelines 1–3 above. These pertained to identity, purpose, and the use of visual cues.

First, many strong, but similar visual elements are positioned on the screen. Without proper focal points and visual hierarchy, these elements compete for the user's attention. This makes it more difficult for the user to focus, discern the identity and purpose of the application, and choose an interaction. Figure 2 shows the issues in more detail. Note the dark grey headers are the strongest individual visual elements on the page.

Fig. 1. GVI home page
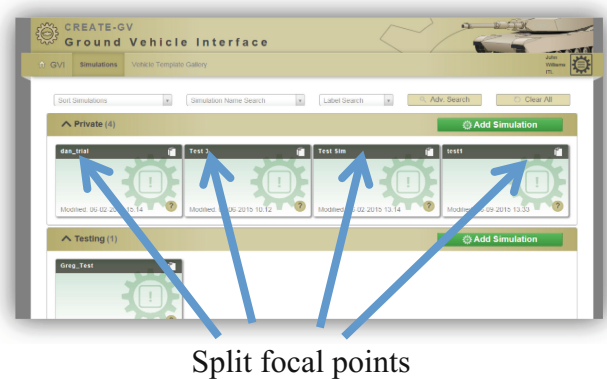

Split focal points

Fig. 2. Home page usability issues

The horizontal listing of simulations is the strongest *group* element on the page, as shown in Fig. 3. However, it isn't clear what these elements contain or the interaction they offer the user. Each simulation rectangle is fairly large; taken together as a group in the horizontal listing, these act almost like "bars" on the page pushing users away visually, rather than inviting them to delve further. In addition, the number of these elements renders other key interactions inaccessible "above the fold" before the user must scroll.

This illustrates highlights of the major issues identified in the walkthrough. To summarize, the main issues identified related to identity and purpose, use of proper visual hierarchy, and navigation.

Walkthrough Recommendations. As a final step to solidify developer acceptance and confidence, the UCD team presented mockups of possible solutions to key problems noted in the walkthrough. As noted, this technique offers critical appeal to the linear thought process of the development team, by illustrating a visible path from "Point A" to "Point B." Figure 3 shows a mockup of a refined home page that
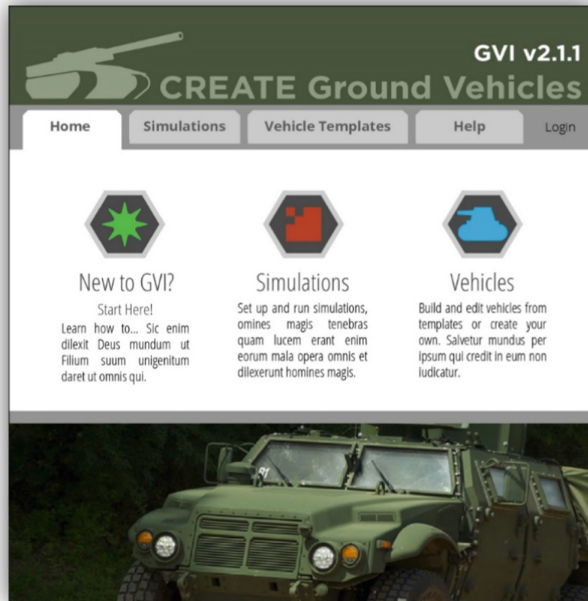
Fig. 3. Redesigned home page mockup

addresses key issues identified in the walkthrough. A more detailed discussion of this page is given in Sect. 2.4.

Subsequent to presenting the walkthrough results and mockups, recommendations for further actions were provided to the development team. The recommendations emphasized that the software provided rich, complex, and greatly needed functionality to users in the HPC M&S community. This acknowledged the effort and skill of the developers, another aspect of the overall process critical to success, but often overlooked by UCD teams without experience in both design and engineering cultures. To fully tap the potential of the software, however, it was recommended that representative users be included in the usability design process. As noted in [16, 17], since GVI is a web application, particular emphasis would need to be given to two activities:

1. Clearly defining user profiles, including novice and expert roles
2. Performing detailed task analyses for each category of user profiled.

These activities would need to be conducted iteratively with the development of an increasingly interactive series of mockups and prototypes which users and developers would review and test. This would require: (1) assembling a properly composed user advisory panel, and (2) executing an appropriately detailed prototyping process. These are described further in the sections below.

2.3   User Advisory Panel

The rationale and methods for assembling user advisory panels (roughly 2–5 users) is argued in a variety of sources, including [19, 20]. Studies have shown that, if properly composed, a group of this approximate size can be very effective for conducting user-centered design [20]. This diverges from conventional software development methods in that it focuses on eliciting user needs, rather than developer concerns, to drive the UI design [19]. Therefore, panelists would be chosen to provide the widest representation of typical user needs. To achieve this goal, members of the user advisory panel (UAP) would be selected along at least two criteria, (1) organizational representation, and (2) diversity of experience, i.e., novice versus expert.

GVI would be available to users in three different organizations, The U.S. Army Tank Automotive Research, Development and Engineering Center (TARDEC), the U.S. Marine Corps, and the U.S. Navy. The final composition of the panel included 3 TARDEC users, 2 Marines users, and 1 Navy user. From within this overall mix, 4 users identified themselves as experienced in the use of modeling and simulation (M&S) software tools, while 2 users considered themselves novices.

2.4   Prototyping

The visual, interactive nature of a user interface, particularly for a web application, makes it impossible to capture all usability requirements through verbal interviews with users [14, 16, 18]. User review of interactive prototypes allows design and development teams to iteratively refine, adjust, and "get it right" before moving to a final deployment. The UCD team had created a first phase non-interactive set of mockups to capture the functionality and look and feel of the enhanced design features. The mockups were used as a starting point for the prototyping process. A mixture of horizontal (overall system organization) and vertical (specific feature) prototyping would be used [20–22]. Periodic reviews by the UAP of these prototype designs would be scheduled as they became available. The prototypes would incrementally evolve into a fully functional interface, with more comprehensive usability and error testing scheduled in the latter half of the project lifecycle.

Iterative Prototype Review. The UAP participated in 3 cycles of prototype review, each of which led to refinements that were then evaluated in subsequent review. The first prototype review used a set of non-interactive screen designs, arranged in sequences to simulate interactions. Users were asked to evaluate the screen designs for overall look and feel, i.e., color scheme, layout, and organization and for how well they captured potential interactions that would support user goals. The screens reviewed were not comprehensive, but selected to represent interactions that were either critical or would likely occur frequently in the regular use of the GVI application.

Follow-up phone interviews were also conducted in which users further clarified comments regarding any aspects of GVI, including their goals and needs, how well the screen designs reflected those needs, and suggestions for enhancements. Screens and interactions evaluated included those for the home page, adding simulations and vehicles, and an interactive tutorial for new or novice users, "New to GVI?" The redesigned home page is shown with comments in Fig. 4.
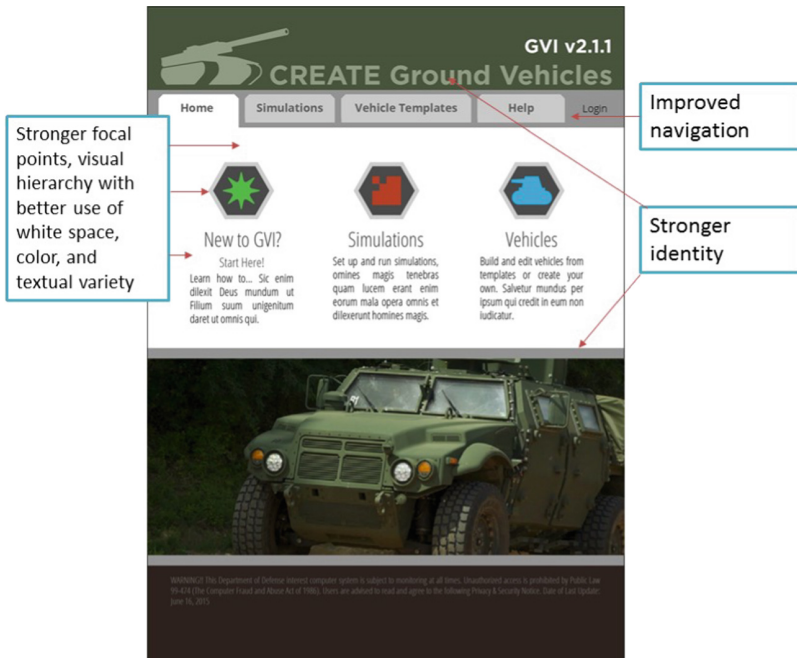
Fig. 4. Redesigned home page

Discussions with the UAP prior to beginning the prototyping process uncovered the need to provide a path through the application for new or novice users. The option "New to GVI?" shown in Fig. 4 on the redesigned home page was designed to meet this requirement. If selected by the user, the screen shown in Fig. 5 begins an interaction sequence that walk new users through the application. Several options for the initial screen were presented for review. The final design for this screen chosen by the UAP is shown in Fig. 5.

From this initial screen, multiple design options for the first 3 steps, 'Add a new simulation,' 'Add tests,' and 'Add a vehicle,' were again presented to the UAP for review. The initial screen in the sequence of the final options chosen by the UAP for 'Add tests,' and 'Add a vehicle," are shown in Figs. 6 and 7.

Results of Review 1. While many important details were gathered from the first review, the critical overall issues, determined from all UAP discussion, both online and via phone, are listed below in order of priority order:

1. Unanimous support for a "project management" option: The UAP requested a method to organize and store all related information regarding a set of vehicles, simulation test conditions, and results in a single location.
2. More emphasis on the concept of "vehicle" over "simulation" as an organizing theme and metaphor. The first prototype focused on a simulation as the central organizing theme. This initial focus made sense as the driving concept of the application was to support advanced simulation tools. However, users all agreed
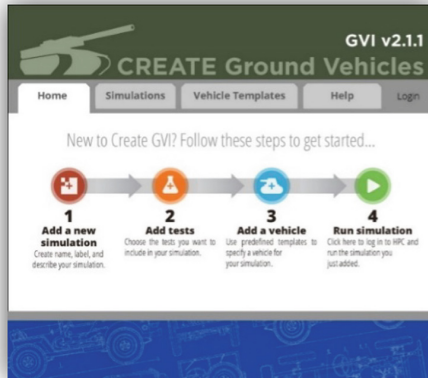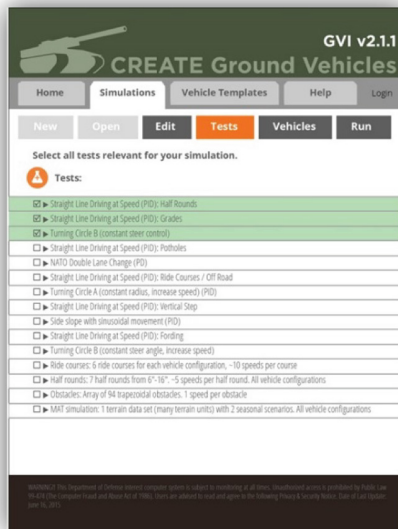
Fig. 5. 'New to GVI' initial screen

Fig. 6. 'Add tests' option

that in their mental processes, the vehicle was first and foremost in their workflow, followed by the operating or test conditions under which they would run simulations. In subsequent prototypes we attempted to better reflect that order in multiple aspects of the designs.

3. Confusion resulting from use of the term "test" to denote the events or conditions in which a vehicle would be evaluated for performance during a simulation. Opinions conflicted on the appropriate candidate, but the following were offered: Test event,

Fig. 7. 'Add a vehicle' option

Operating conditions, Test conditions, Universal agreement that "metric" is not the correct language for this concept.

For the second prototype, we chose the term "test conditions" for these designs since it contained elements of multiple comments, expecting the need for further possible refinement. Again, users have ideas in discussion or on paper that need refining once they experience them interactively.

4. Need for a menu option for vehicle editing that provided maximum information in minimal space. Our second prototype design sought to better fit that need, with refinements expected.

Figures 8, 9 and 10 show highlights of the critical refinements.

Many other screens and interaction sequences were refined to better capture user requirements and preferences resulting from Review 1. These screens show only highlights of critical refinements to the second prototype.

Results of Prototype Review 2. Highlights of the second prototype review by the UAP included the following: (1) Project management option remained unanimously supported with refinements needed for *adding test conditions* and *adding vehicles* to a project in the areas of interactivity and level of detail; (2) A new 'Vehicle Builder' option was positively received with refinements needed in organization/flow; (3) A 'Simulation Builder option was positively received with requests to match the general design of 'Vehicle Builder' as it develops. Once the refinements identified in Review 2 were complete, the UAP was asked to perform one final review before summative usability testing.

Results of Prototype Review 3. Highlights of the third prototype review by the UAP included the following: (1) Further refinements requested in the 'New to GVI' option for both interactivity and layout; (2) Refinements in aspects of interactivity were requested for 'Project Manager' and 'Vehicle Builder.'
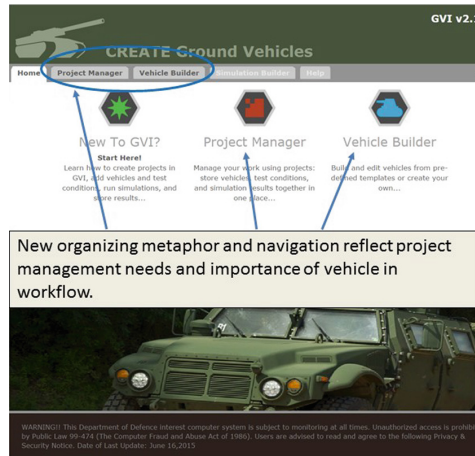
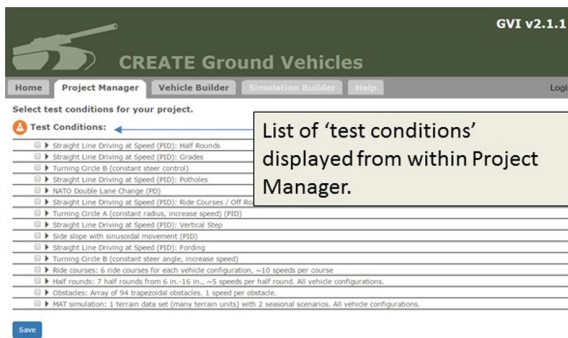**Fig. 8.** Refined home page showing new organization and navigation

Fig. 9. Use of new terminology and navigation for 'test conditions

2.5 Summative Usability Testing

Following the iterative cycle of prototype design, implementation, UAP review, developer review, and refinement, the interactive prototype was presented to naïve users for a summative usability evaluation.

Participants (N = 13) were recruited from the general population to provide the perspective of a naïve user. Each participant was asked to complete five data entry and modification tasks using the GVI. In one task, the participants were asked to use the 'New to GVI' interface to define a new project and run a simulation. In other tasks, the participants were asked to add or modify projects or vehicles without the 'New to GVI' interface. After each task, the participants were asked to complete the 10-question System Usability Survey (SUS) [23]. The SUS provides a quick but reliable evaluation of the usability of a system. It is a 10-item Likert questionnaire.

Overall SUS scores were calculated and responses to each of the 10 items were also analyzed. An SUS score above 68 is considered above average. The average overall
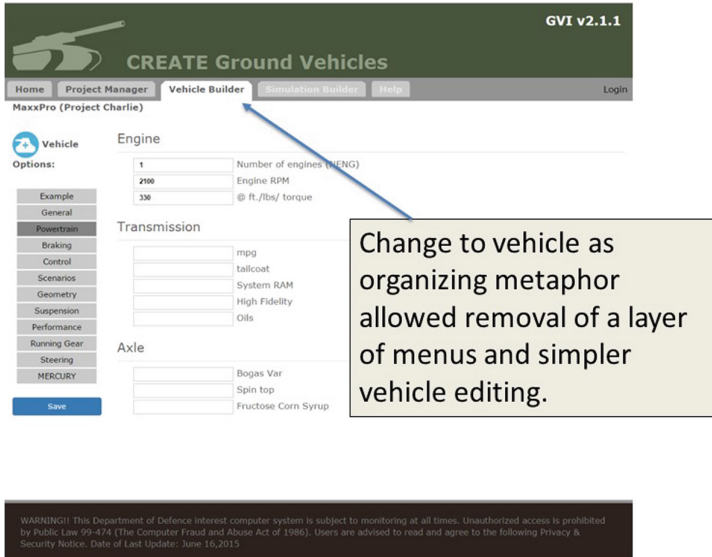
Fig. 10. Vehicle central to user mental model and workflow

SUS score for the GVI was 88.4 (SD = 9.6) indicating that the final design was well received. The mean and standard deviation for overall scores and individual SUS items for the 3 major interface areas are listed in Table 1.

Table 1. Mean and standard deviation for the individual items in the SUS survey by interface.

SUS score/item	New to GVI		Project manager		Vehicle builder	
	M	SD	M	SD	M	SD
Overall SUS	84.0	10.4	88.9	10.2	90.0	9.8
Would use frequently	2.9	0.9	2.8	0.6	2.9	0.6
Too complex*	3.5	0.8	3.7	0.8	3.7	0.9
Easy to use	3.5	1.0	3.7	0.5	3.7	0.5
Need support to use*	3.7	0.5	3.7	0.4	3.8	0.4
Well integrated	3.0	0.7	3.3	0.6	3.4	0.7
Too inconsistent*	3.5	0.8	3.7	0.5	3.7	0.5
Quick to learn to use	3.6	0.7	3.6	0.5	3.7	0.4
Cumbersome to use*	3.5	0.9	3.6	0.9	3.6	0.7
Confident when using	3.0	0.8	3.7	0.4	3.8	0.5
Need to learn a lot*	3.5	0.7	3.7	0.6	3.8	0.4

Note: * Scores adjusted so higher number indicates positive feeling across all SUS items.

3   Conclusions

Engineering and design cultures are inherently opposite in their approaches, the former linear and risk-averse, the latter cyclical, iterative and risk-embracing. This case study has examined how these two cultures can be merged to produce successful software products and how lessons learned may be applied beyond its scope. The key factors contributing to its success factors included a design team with experience in the worlds of both design and engineering, a careful selection of UCD methods, and finally a management structure that embraced cultural change.

References

1. Norman, D.: The Design of Everyday Things Revised and Expanded, pp. 217–293. Basic Books, New York City (2013)
2. Nielsen, J.: Heuristic evaluation. In: Nielsen, J., Mack, R.L. (eds.) Usability Inspection Methods. Wiley, New York (1994)
3. Nielsen, J., Mack, R.L.: Usability Inspection Methods. Wiley, New York (1994)
4. Bias, R.G.: The pluralistic usability walkthrough-coordinated empathies. In: Nielsen, J., Mack, R.L. (eds.) Usability Inspection Methods. Wiley, New York (1994)
5. Spencer, R.: The streamlined cognitive walkthrough method: working around social constraints encountered in a software development company. In: Proceedings of CHI 2000, pp. 253–259 (2000)
6. Nielsen, J.: 10 Usability Heuristics for User Interface Design. NN/G Nielsen Norman Group, 1 January 1995
7. Nielsen, J.: Designing Web Usability. New Riders Publishing, Indianapolis (2000)
8. Nielsen, J., Loranger, H.: Prioritizing Web Usability. New Riders, Berkeley (2006)
9. Van Duyne, D.K., Landay, J.A., Hong, J.I.: The Design of Sites: Patterns, Principles, and Processes for Crafting a Customer-Centered Web Experience. Addison-Wesley, Boston (2002)
10. Nielsen, J., Tahir, M.: Home Page Usability: 50 Sites Deconstructed. New Riders, Berkeley (2007)
11. Pirolli, P., Card, S.: Information foraging. Psychol. Rev. **106**(4), 643–675 (1999)
12. Pirolli, P.: Information foraging on the web: rational analyses of human-information interaction. Cogn. Sci. (2004)
13. Brinck, T., Gergle, D., Wood, S.: Designing Sites that Work: Usability for the Web. Morgan Kaufman, San Francisco (2000)
14. McIntire, P.: Visual Design for the Modern Web. New Riders, Berkeley (2008)
15. Denney, H.: Navigation issues: sticky menus are quicker to navigate. Smashing Mag. (2012)
16. Jovanovic, J.: Designing User Interfaces for Business Web Applications. Smashing Mag. (2010)
17. Nielsen, J.: Best Application Designs. NN/G Nielsen Norman Group, 23 April 2012
18. Beier, B., Vaughn, M.W.: The bulls-eye: a framework for web application user interface design guidelines. In Proceedings of CHI 2003, pp. 489–496 (2003)
19. Baca, J.: GVI Usability Evaluation and Recommendations, 29 July 2015

20. Preece, J., Rogers, Y., Sharp, H.: Interaction Design: Beyond Human-Computer Interaction. Wiley, New York (2002)
21. Norman, D.A., Draper, S.W.: User Centered System Design: New Perspectives on Human Computer Interaction. Erlbaum, Hillsdale (1986)
22. McCurdy, M., Conners, C., Pyrzak, G., Kanefsky, B., Vera, A.: Breaking the fidelity barrier: an examination of current prototypes and an example of mixed-fidelity success. In: Proceedings of CHI 2006, pp. 1233–1242 (2006)
23. Brooke, J.: SUS: a quick and dirty usability scale. Usability Eval. Ind. **189**(194), 4–7 (1995)