Orchestrating Design Concepting to Envision Application Requirements

By Jacob Burghardt | Flashbulb Interaction, Inc. | www.flashbulbinteraction.com

This article outlines a perspective on requirements definition for complex software applications that involves much more than "gathering" exacting needs. Software definers can expand their focus into orchestrating the early visualization of potential design directions in their cross functional teams. In this role, analysts can facilitate the collaborative mapping of ideation opportunities and exploration of application design possibilities, with the goal of uncovering inventive sources of value in end users' activities. Taking on this role in the collaborative process of design concepting should not be mistaken for "taking over" design. Instead, requirements definers can add considerable value in their teams by structuring key aspects of generative design exploration, guiding the synthesis of divergent application directions, and driving the informed selection and communication of a compelling big picture vision for an onscreen offering. In short, analysts can valuably support their teams as they develop a shared, visualized idea of the role that their product might play in users' experiences.

Pull of details over big picture requirements

In order for a computing application to effectively support a workplace activity, the creators of the tool must meaningfully rationalize the work that they are striving to support. As requirements definers know, envisioning effective technologies begins with a clearly articulated understanding of a product's essential boundaries, contents, and pathways.

The process for articulating that understanding can take many different roads, each with different strengths and drawbacks. From the perspective of someone who views these definition processes from a design orientation, many of these roads share a common weakness, especially in their practical application in the real world: they do not pay sufficient attention to a product's big picture, given the defining "pull" of many needed details.

Requirements definers are typically valued for their ability to capture and make sense of a technology's manifold particulars. After all, software can be excessively complex — especially when thinking about the types of onscreen tools that are adopted by specialized knowledge workers. In the face of that complexity, many product teams see the need for someone to capture important specifics, at some agreed upon level of detail, so that the rest of the team can deliver. To quote a commonly cited maxim of the influential designer Charles Eames, "the details are not the details, the details make the design."

Challenges to requirements definers traditional sources of value

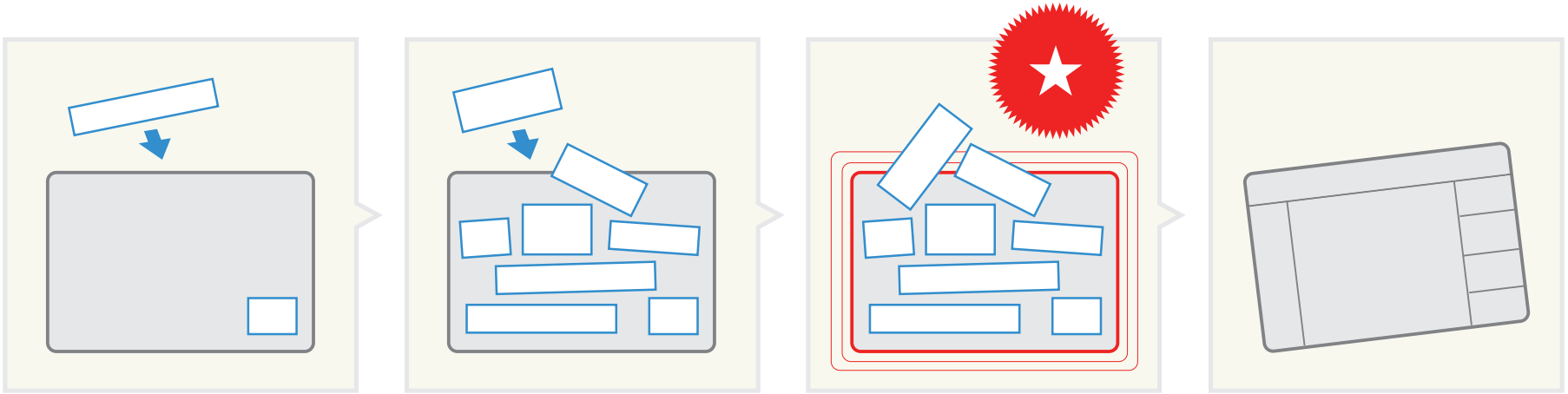
What happens to this detail orientation in product environments where teams are focusing on rapid, Agile iteration, where details are evolved in a just-in-time way, not defined in advance? While Agile processes require a base level of requirements, many teams practicing this development method prefer to keep that base to an absolute minimum. Figure 1, on the following page, presents a critical analysis of that perspective.

The concluding column in this diagram, on the right, hints at important room for improvement in applications for the knowledge workplace. When it comes to contemporary tools for knowledge work, products that are considered essential are not always satisfactory. There is considerable room for both thoughtful tailoring of typical features, as well as ongoing invention and re-envisioning of design for targeted user motivations – developments in user experience that go beyond what is expected to enrich workers' practices and outcomes.

That is not to say that, for some types of applications, the "capture details as we go" perspective will not result in valued tools. This can be especially true when teams have an inherent, shared vision for what they are trying to accomplish, often because the product being developed is in an established genre that teams already understand, not a complex product in a domain that is difficult to develop expertise around.

Even while surrounded by calls for innovation and advancement, many teams want to get started as soon as possible on a making a product "real." These teams do not seem to focus on exactly how or where they will make valuable leaps forward, only getting something done. Iterative implementation can breed iterative, incremental evolution within a well known frame. Our shared conception of what onscreen tools "should" be cuts both ways — it allows product teams to work from a shared vocabulary that they and their users understand, but it also limits novel exploration of tailored solutions to complex workplace activities.

Figure 1. Magic Happens Expectation in Iterative Application Design Process



Begin creating individual features, without spending any time in the space between high level product strategy and detailed product implementation

Iteratively add more discrete parts, without considering overarching ideas about how the application could mediate knowledge work

Until magic happens, somehow unifying the aggregation of separately created minutiae

And a cohesive, or at least satisfactory, application supposedly emerges

In reality, such products may be deeply and frustratingly flawed, driving poor user experience and lesser outcomes in targeted knowledge work

Design driving a shift toward big picture definition

Given this trend away from intensive requirements analysis in the software industry, requirements definers may want to tune into a major trend outside the industry that can stand as a clear counterpoint: the increasing emphasis on holistic product design. Over the last several years, design has become a key differentiator in a range of industries where it had previously not been a driving factor, delivering new sources of value and differentiation.

Tying this trend back to specialized software, a holistic emphasis on design could break the mold of contemporary application development, where definition is often viewed as a separate endeavor, removed from design process. Product teams could try on a larger change in mindset, pushing rough, schematic design further forward in the project process in order to collaboratively uncover and explore various directions that requirements might eventually take.

This exploratory location and fleshing out of constraints, as opposed to the iterative capturing and listing of requirements, is a hallmark of design-lead endeavors. For example, the latest consumer gadget that everyone is talking about did not come from the first "engineered" attempt at a known target. Instead, its design was almost certainly selected from among many different conceptual design options, each of which had been elaborated upon as a way of understanding meaningful directions that the creators might take. And when you think of a cutting edge building near you — designed from well characterized needs and not only some top down artistic perspective — it goes without saying that the team that envisioned that complex structure had to try out many different options before arriving at the final direction (which was then iteratively designed at the level of countless details).

For many requirements definers, this forward movement of design into definition may feel like an infringement. It should not. It represents an opportunity, not an erosion. As proponents of the "design thinking" idea have made clear, this sort of conceptual exploration need not be the realm of the solitary design genius. To the contrary, it can be cross-functional, collaborative and process driven. In fact, the skills that many requirements definers already apply every day may make them particularly well suited to supporting this sort of collaboration between designers and other parts of their teams: invention that targets compelling user experiences in complex domains with multiple stakeholders can benefit from effective facilitation, insightful analysis and clear communication.

Rethinking early software definition as an ideation process

Many requirements definers begin their processes with a considerable emphasis on conceptual definition to set the stage for an effective product. Analysts often begin their processes by creating meaningful high-level maps, allowing them to then dive into the details of potential requirements within specific regions of that bounded space.

These high-level mappings can be a central bridge between initiating product strategy and detailed definition, design, and implementation. When they accurately represent a team's ideas, such maps can become highly effective tools for understanding system scope and coordinating the focus of a dynamic cross-functional group of practitioners. These representations underscore the importance of gaining agreement around a product's big picture, before getting into its many details.

What if, instead of moving on to defining granular requirements within an overall mapping, requirements definers worked to define the boundaries of a solution space, rather than defining a specific set of carved-in-stone constraints? What if, instead of focusing only on what the eventual application design "must do," requirements analysts also went in search of insights that could suggest opportunities for their teams to ideate potential designs – locations where clear cut answers are not known and compelling solutions might be found through a research process of informed, exploratory sketching?

Early, "iteration zero" activities in product development can become an opportunity for analysts to drive generative exploration in their teams, rather than acting as talented scribes that "capture" requirements that are presumably already in existence. Orchestrating these activities does not mean "taking over" the role of anyone who currently accomplishes design within a given product team. Instead, it can mean supporting cross-functional design discussions by pointing out where the team might want to consider different options, facilitating the discussion about where to refine or invent, and then capturing the resulting outcomes and shared understandings.

Figure 2., on the following page, presents one overview for how requirements definers might guide their inclusive teams through the process of envisioning an onscreen application and then following through on that vision.

Potential benefits of this type of process can include the following:

An emphasis on inventive requirements and design

For those that agree that the contemporary applications for specialized knowledge work are in need of some inventive advancements, this sort of exploration arguably stands a better chance at envisioning cohesive and powerful new tools.

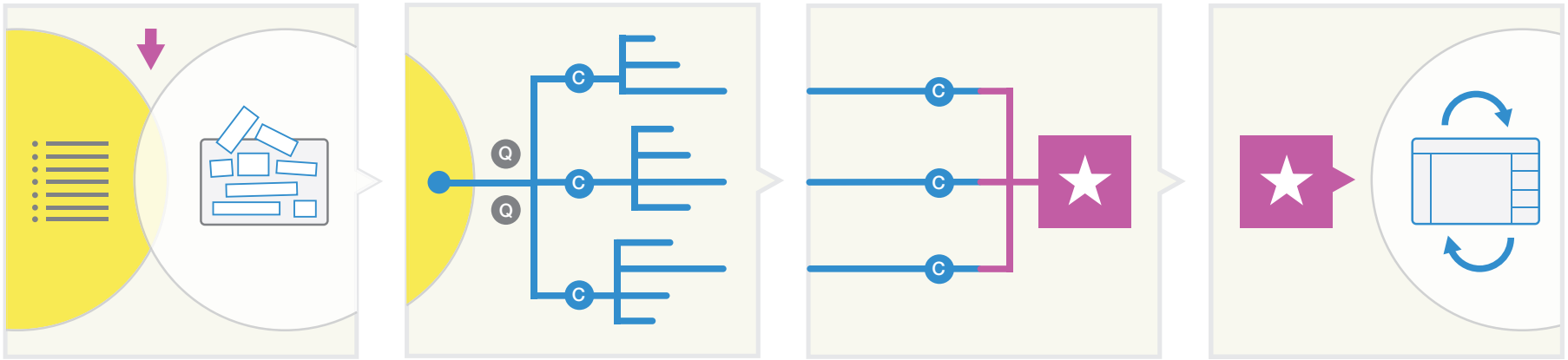
Robust requirements analysis value proposition

For those that want to see requirements analysts expand their strategic role in the development of new technologies, this sort of design-lead process can place requirements definers in a highly collaborative, orchestrating role. In this role, analysts structure a cross-functional teams' activities as they envision valuable connective tissue between high-level product strategy and the typical product development process, clarifying a project's focus and direction.

Shared vision for implementation minded practitioners

For those that are concerned primarily with the long haul of product implementation, application concepting can provide product teams with big picture "narratives" that explain essential constraints for a successful solution in simplified, rational, and emotionally resonant ways. Analysts can carefully assemble a cogent set of the product team's big picture design sketches and guiding user experience ideas, synthesizing a product vision and outlining spaces where a team might go on to "fill in the blanks" in compelling ways.

Figure 2. Proposed Application Envisioning Process



Spend more time in the space between high level product strategy and detailed product implementation

Meaningfully question what it could mean to mediate certain knowledge work activities with technology: observing and talking with targeted workers, collaboratively modeling the problem space, and sketching diverse design concepts

Strategically synthesize the fittest overarching vision and concept for your product from among an ecosystem of envisioned futures

Then move forward with your chosen design strategy and design concepts, expanding upon details, iteratively implementing and gathering further input

Potential application envisioning activities

The following are general descriptions of some potential cross-functional, collaborative, application envisioning activities that requirements definers could be central in orchestrating:

Setting the stage for design ideation

- Promoting competitive analysis and other forms of secondary research, such as user experience trend mapping
- Gathering and visualizing relationships in high-level requirements
- Identifying and visualizing opportunities for generative sketching

Facilitating design ideation

- Ensuring a quorum of cross-functional team participation
- Bounding the topics and focus of ideation sessions
- Guiding a fruitful, positive discussion of user experience possibilities

Facilitating synthesis and elaboration

- Capturing the breadth of the team's ideation outputs in accessible ways
- Synthesizing themes in ideation outputs
- Reflecting that synthesis back at the design team for iterative discussion
- Collaboratively pulling together meaningful theme assemblies that represent different conceptual directions for the application

Mapping strategic and business implications

- Charting the potential strategic impacts of different conceptual directions in the context of larger product strategy discussion
- Mapping the potential user experience characteristics and challenges of different conceptual directions
- Characterizing potential business risks and benefits of different conceptual directions

Facilitating the selection of a single concept

- Ensuring a quorum of cross-functional design team participation and management participation
- Informing decision making with compelling analysis of different conceptual directions
- Facilitating the informed selection of an application concept

Distilling and communicating the selected application concept

- Extracting key attributes and requirements for the chosen concept
- Consolidating the core information into a meaningful message and narrative that the product team can keep front and center
- Communicating that story broadly within the organization

Ensuring that the application concept remains a guiding vision

- Support the product team through the process of iteratively turning the application into a reality
- Ensure that the selected application concept retains its relevance throughout ongoing development, potentially leading to updates

Conclusion

To move past the limited user experience advancement found in many types of complex software – such as onscreen tools that many specialized knowledge workers use every day – requirements definers can transition toward a new role that may very well be a natural extension of what they are currently doing. This new role can build on analysts' existing skills of effective facilitation, insightful analysis and clear communication, applying those skills to a new process: orchestrating early, collaborative envisioning of divergent concepts for application design within product teams.

This transition toward application concepting can signal a clear acknowledgement that defining software is not simply about cataloging requirements that are somehow waiting to be gathered. Instead, requirements definers can drive collaborative exploration and invention within product teams, promoting compelling advancements in customer productivity, user experience, offering differentiation, and other key areas of application value.

Request for feedback

This article presents an argument for requirements definers to weigh in their own practices, not yet-another-new-process-edict. Any input on this proposed shift in focus for software requirements analysts would be very much appreciated.

I look forward to hearing about how these ideas hold up in the context of your own product development challenges.

jburghardt@flashbulbinteraction.com

Related free e-book for application definers

This article is a continuation of a line of thinking first presented in the Creative Commons e-book, "Working through Screens: 100 Ideas for Envisioning Powerful, Engaging, and Productive User Experiences in Knowledge Work."

Written for use during early, formative conversations, "Working through Screens" provides product teams with a broad range of examples and considerations for setting the overall direction and priorities of new or iteratively improved applications for thinking work.

This free e-book is available online in html, pdf, and summarized pdf "idea cards" formats at:

http://www.FlashbulbInteraction.com/WTS.html

About the author + Flashbulb Interaction, Inc.

Jacob Burghardt is a research, strategy, and design consultant at Flashbulb Interaction, Inc, a specialized studio that collaborates with clients to drive vision at the forefront of knowledge work user experiences. Flashbulb Interaction consultants act as trainers, advisors, and facilitators — augmenting and inspiring client teams as they envision new and improved onscreen offerings.

Jacob works with clients to collaboratively investigate, interpret, and visualize application design challenges in a way that drives decision making, shared vision, and meaningful innovation. He has consulted to industry leading clients on a range of different workplace computing tools, including products for engineering workflow, aviation navigation, scientific instrumentation and data analysis, and financial trading.