

MAKE IT USEFUL

“There is nothing worse than a crisp image of a fuzzy concept.”

—Ansel Adams

What Are We Trying to Do and How Will We Know If We Did It?

Being able to clearly articulate what a team is expected to do and what the desired outcomes are is the first step on the road to success. That sounds obvious, but I’ve been in countless kick-off meetings where neither of those two questions could be answered by the project stakeholders. The leadership teams had some vague statements to share about the high-level direction such as “Improve the UX” or “We want to be the Apple of our industry,” but nothing that is specifically actionable.

This isn’t just my observation. According to an article published in the *MIT Sloan Management Review*, “Only one-quarter of the managers surveyed could list three of the company’s five strategic priorities. Even worse, one-third of the leaders charged with implementing the company’s strategy could not list even one.” (sloanreview.mit.edu/article/no-one-knows-your-strategy-not-even-your-top-leaders)

If you try to shoot the flock you won’t hit anything. You need to pick a goose and target it specifically. The same thing is true in business. The first thing you need to do to succeed is to identify and document your goal.

In contrast, one customer of mine came to me with a very specific goal. They knew from their analytics that customers who bought specific products had higher average order values (AOV). Their goal was to increase the number of visitors to their site that purchased those specific items.

With that clearly focused goal in mind, I researched how their competitors were attempting to solve the same problem. In this specific case everyone seemed to be doing the same thing, so there wasn't any inspiration to be found. Since I couldn't find anything helpful while reviewing the competition, I completed a "clicks to complete" assessment to quantify the complete task from start to finish. A clicks to complete assessment simply measures each click required to complete a specific task or use case. Knowing the number of clicks (or interactions) it takes a user to complete a specific task can help identify issues and help validate potential solutions.

In the original version of their site this task took 50 clicks to complete. I could tell from my evaluation that there were a few big problems. The first and biggest problem was that the existing user flow required the user to jump back and forth between different types of tasks. This can cause users to lose context and abandon the process. The second was that all those clicks increased the user's "time on task" and the longer it takes users to complete a task, the more likely it is that they will get sidetracked.

After verifying both of those issues via their usage analytics, I began exploring options for how we might eliminate the loss of context and reduce the number of clicks to complete. I figured if we could do that, we would likely see an increase in our conversion rate and average order value Key Performance Indicators (KPIs).

My initial efforts were focused on reducing the loss of context. To do that I reviewed various interaction pattern libraries (www.welie.com/patterns and www.patternfly.org/v3) in search of inspiration for something that would eliminate the need for users to move back and forth through the process multiple times to complete their transaction.

In the end, I decided on a wizard-based user flow so that users could complete subtasks one at a time and move through the process step by step. Wizard interfaces break down complex user flows into individual screens for each step. This allows the user to focus on one task at a time and also provides space in the interface for more instructions. That solved the first problem and also reduced the number of clicks significantly but created screens that were a bit dense with form fields. When I first tested those wireframes with some coworkers, they mentioned that seeing all those form fields that needed to be completed in the new design was intimidating. They were right.

This is the same problem that Disney has in its theme parks. If visitors could see the actual length of the line they would need to wait in to be able to get on the ride, they

might choose to not even try. As the amazing design book *Universal Principles of Design* points out . . . to solve that, Disney uses a technique called “progressive reveal.” All that really means is that Disney hides a lot of the line from view by including the line in the building that houses the attraction and zigzags the line so a visitor only ever sees the first group of people instead of the entire line. The actual length of the line gets progressively revealed as the user moves through the process.

Seeing only the first part of the line is less intimidating, so visitors are more likely to get in the line. Once they are there, they become more and more invested in staying in the line because of the time they have already invested in being there in the first place.

I decided to go back to the whiteboard and incorporate the progressive reveal technique into my design by using the accordion interaction pattern with the wizard interface I created. An accordion interface provides summary data for each element that needs further review. The first element that needs input is expanded so the user can complete the required interactions, while the others are collapsed in order to save space and not overwhelm the user. This approach reduced the number of form elements the user sees, making it more likely that users will start the process. Once they start the process, they start building an investment of time and become less likely to abandon it.

Once I had a clickable prototype of the new accordion/wizard hybrid interface ready, I tested again with more coworkers because we didn’t have a budget to test with users. It wasn’t ideal, but testing with people even if they are not a direct match for your user demographic can be better than not testing at all especially if the system you are working on is used by the general public.

This new interface reduced the clicks to complete by 50% and eliminated the loss of context issue that the previous version had, so I was pretty confident that testing would go well.

The coworkers were able to complete the tasks well and didn’t have any further feedback, so we moved this solution on to development.

This entire process, from the first meeting where the team discussed what they found in the analytics until the new interactions went live on the site, took only two months and cost less than \$35K in development costs. Immediately after this solution went live, we saw an increase in conversions and an estimated \$9.00 increase in average order value. At that rate, it only took a few days of orders to pay for the entire effort, and the rest of that AOV increase going forward has created a very healthy return on the investment.

There was only one designer and one contract developer on this project, and a lot of the larger user experience design process was missing from this effort due to time and resources. You don’t need a multimillion-dollar budget and a team of hundreds to

make a real impact on your business. The first things you need are a clear goal and the knowledge of how you will measure your progress along the way.

Creating a Constitution for Your Project—The Framework That Empowers the Team

Choose your own adventure . . . do you want to read another anecdotal story to help provide context for the material that follows? If so, just keep on reading. If not, skip to page 5.

The road to vaporware is paved with tradeshow promises.

If you have been in the software business for very long, it's likely you have been there. It's tradeshow season and nothing else matters but getting something together that will help the sales team demonstrate enough new value in your product to justify the cost increase the board of directors is trying to cram down everyone's throat. Everyone is in a tough position.

The product people know they are the ones who are supposed to find this new value and articulate it in terms of a plan that can be executed in some overly aggressive time frame. It doesn't take too long for the product team to start "thinking different" and eventually come up with some ideas that sound great in the echo chamber, but in reality would take at least five years of time to develop *if* it was actually the right direction in the first place.

The designers know they are rushing the process and that everything they are doing is based on the assumptions of the C-level folks and the product team instead of being vetted with user testing and/or market analysis because there simply isn't enough time. They also have a sneaking suspicion that the elaborate screens they are creating for the demo likely include functionality that is basically impossible to build. That said, they have the power of Sketch and InVision to make all these terrible ideas look fabulous.

Then there are the developers. They have lived this cycle so many times it barely fazes them. They are about to be handed a nightmare and be wedged firmly between an ambiguous design spec full of dubious assumptions and a rock-solid tradeshow deadline.

The dev team pulls together an entirely rigged demo full of holes and fake data with the promise from the leadership team that after the tradeshow we'll take some time to "do this right," refactor, and in general create a viable product.

To help avoid many of these problems modern software teams face, I leverage goals, strategies, objectives, and tactics to establish consensus and protect the team from needless pivots, sprint injections, or otherwise being derailed.

In order for this to work you need to invite decision makers from the product, development, UX, design, research, and QA teams to a workshop where you will define your goals, strategies, and objectives together. I'll explain how to do this in the next chapter but for now, just know that consensus is essential at this phase and also know that it is OK if you don't have members of all those teams. The core team, or as some companies call them, "squad," should include members of the product, UX, and development teams to ensure that your plan has buy-in from the teams most connected to the building of your product or service.

Once the squad has created the constitution for the project the squad can protect it effectively to ensure it stays on track.

If the head of sales comes back from a tradeshow with customer feedback that "has to be addressed immediately," members of the squad can refer to the constitution for the project and ask what objective of the project this customer request will positively impact. If there is a compelling argument that this new idea will help achieve an existing objective, the team can consider shifting priorities. If there is no compelling argument to be made, then add this new idea to the backlog to be revisited when the current objectives have been addressed. This isn't to say that the constitution cannot be amended, because there are circumstances in which it will be necessary. This process simply helps to make sure changes in direction are not taken lightly and that voices from key team members are heard as part of the process.

Goals, Strategies, Objectives, and Tactics—The Plan the Team Will Work From

Once you recognize the need for a unifying plan for your teams, you will need to begin the process of getting enough buy-in from your organization to justify the time needed for the initial workshop.

Over the years I've learned that opinions are not worth very much. If you base your business decisions on opinion the best you can hope for is tentative cooperation, but there will always be those that naysay your plans and undermine your team's ability to work as a cohesive unit. When leading from your gut instincts it is also more likely that you will lead your team in the wrong direction.

Data and information provide a solid foundation to make observations. Using data to back up your decisions also provides a defensible position for recommendations you

make for your team. It's not always possible to use data to make every decision, but you should try to whenever possible.

The first rule of buy-in is inclusion. In my experience people are far more likely to fall in line with a plan that they were involved in from the start and when their voice has been heard as part of the process. A wonderful side benefit of including the team in the process is that you will likely also uncover more insights and identify more robust solutions.

The first step is to reach out to the key members of your organization with a Monday morning email asking them to take 15 minutes to document their understanding of the biggest challenges facing your business. See the following sidebar for an example.

Hi <name>,

We are kicking off a user experience project for our organization and would like to include your feedback in the process. If you can, please take 15 minutes before the end of the day Wednesday to reply to this email with a list of what you see as the biggest issues currently facing our business.

Feel free to reach out with any questions you have,
<your name>
<your contact info>

Once you have as many responses as possible on Thursday morning, create a spreadsheet and do the analysis required to remove the duplicates and then prioritize the issues the team identified. The list of issues you documented will serve as a great starting point for your workshop.

When you are done you should have something that looks like Figure 1.1.

The next step is to create a workshop agenda that can be used to help participants better understand what to expect, and hopefully get them excited about being part of a process that has been proven at some of the world's biggest and most profitable organizations. I use a spreadsheet to kick off the process of creating my agendas. In the end, the final product is usually a slide deck that gets used during the workshop but the spreadsheet is a fast way to document the process and iterate on content, timing, and presenters. The spreadsheet is also helpful as an artifact that can be shared with your team to gather feedback and get approval.

When I'm done with my first pass at a workshop agenda, it looks a lot like what you'll see in Figure 1.2.

You can find a link to the agenda spreadsheet template here:

www.chaostoconcept.com/workshop/agenda

Survey Results – Issues and Awareness – Fall 2019									
File Edit View Insert Format Data Tools Add-ons Help 100% \$ % .00 123 Default (Arial) B I A 10 Last edit was seconds ago Share									
Issues	A	B	C	D	E	F	G	H	
Issues		Sales	Marketing	Design	Development	Leadership	Support	Awareness	
1									
2	New to market competition is leapfrogging us in terms of functionality	X	X			X		3	
4	Our tech stack is aging and actively preventing us from being able to respond to the market in a timely manner			X	X			2	
6	Customers do not understand our billing methods resulting in anger, confusion, and extended support calls						X	1	
8	Our app looks dated because we haven't done anything to maintain or improve the UX since we originally got funding 3.5 years ago.	X	X	X	X	X	X	6	
9									

Figure 1.1: An Issues and Awareness Document can help expose your team to issues they were previously unaware of while acting as a discussion starter for a priority-setting exercise.

Workshop, Day 1 – Feb. 2, 2020						
Est. Time/Hrs.	Agenda Item	Description	Location	Deliverables	Team	Optional
8:45 AM	Continental Breakfast		3rd Floor Great Room			
9:00am – 9:30am	Kick off	1. Welcome 2. Team introduction 3. Reviewing the goals of the workshop 4. Reviewing agenda	3E	Meeting Notes	Kevin Lisa Nadine Tracy Tony Julie Kris Gabrielle	
9:30am – 10:00am	The Goal Driven Design Process	Review how goals, strategies, and objectives serve as the foundation of our design process	3E	Meeting Notes	Kevin Lisa Nadine Tracy Tony Julie Kris Gabrielle	
10:00am – 10:45am	Company and Goals	Understanding company and goals 1. High level business goals 2. Business goals for this project 3. Sales positioning: How does your sales team sell? 4. Organizations within (i.e. Sales, Marketing, Product, Design, Engineering) 5. Success metrics	3E	Initial GSOT Document	Kevin Lisa Nadine Tracy Tony Julie Kris Gabrielle	
10:45am – 11:00am	Break					
11:00am – 12:00pm	Review Study Findings	Review findings from study and how they correlate with assumptions / information	3M	Research Study Report	Kevin Lisa Nadine Tracy Tony Julie Kris Gabrielle	
12:00pm – 1:00pm	Lunch		LUNCH DELIVERED			
1:00pm – 2:30pm	Review Competitive Usability Analysis	Review competitive usability analysis and how it correlates with assumptions / information	3M	Competitive Review Report	Kevin Lisa Nadine Tracy Tony Julie Kris Gabrielle	
2:30pm – 2:45pm	Break					
2:45pm – 3:45pm	Develop Strategies	Discuss and document any current and newly identified strategies related to the stated business goals	3M	Updated GSOT Document	Kevin Lisa Nadine Tracy Tony Julie Kris Gabrielle	
3:45pm – 4:00pm	Wrap-up	Discuss Day 2 agenda. Wrap-up		Meeting Notes		

Figure 1.2: Workshop agendas help participants prepare and keep the team on track.

Once you have a solid draft, it's a good idea to share it with a small set of key team members to get some feedback. Use their feedback to create a final draft version and use that as part of your invite to the workshop.

Let the participants know that a draft agenda is attached to the invite and make sure they are aware of what the goals of the workshop are along with what their time commitments will be. I find that this is a good time to also include a link to Human Factors International's quick video on the ROI of UX. If you have some doubters in the mix, that video will hopefully convince them that the time invested in this process will be worth it. The video in combination with your thoughtfully prepared agenda should prove to be pretty compelling:

www.youtube.com/watch?v=094kYyzqvTc&t=160s

Once you have the workshop scheduled and have at least members of the core team (a product team leader, a UX team leader, a dev team leader, and hopefully an executive that will act as the voice of the C-level) confirmed as participants, you can go on to create your workshop presentation slide deck.

A slide from my presentation deck might look something like what you'll see in Figure 1.3.

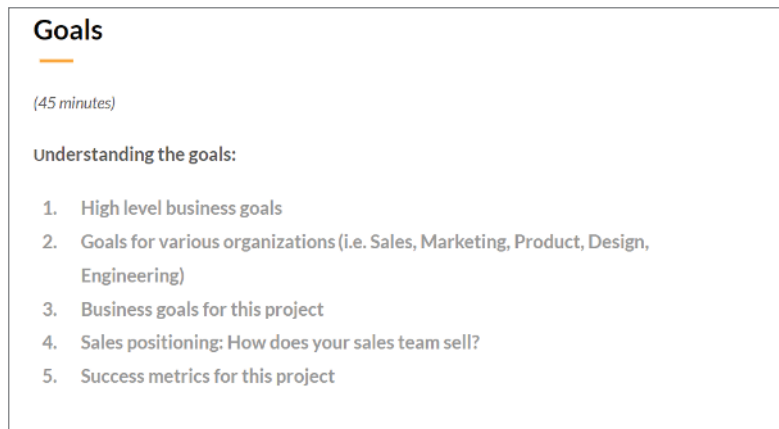


Figure 1.3: A simple slide deck helps participants follow what's being presented and gives presenters a place to showcase support materials.

You can find a link to a template of my deck at www.chaostoconcept.com/workshop/sides.

Once you have your slide deck prepared, it's a good idea to walk through the process with a trusted colleague if this is your first time doing something like this. It will help you become more familiar with the process and will also help ensure that what you have planned will lead to the desired outcomes.

In this workshop it's very important that your outcomes include:

- **The high-level goal of the project.**

An example of a goal might be to sell more of a specific product or service because this will increase a metric like average order value (AOV) or conversion rate.

- **The strategies that are identified as being likely to help achieve that goal.**

A strategy for the defined goal might be to make it easier for users to find the product. Another strategy might be to provide more information about the product on the product detail page (PDP) so users can compare products more easily and feel more confident in their choices, making it more likely they'll complete a transaction.

- **The measurable objectives required to be able to execute on each strategy.**

An example of an objective could be updating product descriptions with specific keywords so that the search algorithm will surface the target product in more search results, making it easier for users to find. Objectives need to be measurable so you can easily report on whether or not this objective has been met. In this case we could measure this objective by documenting that we need to update the descriptions of 10 key products with at least 5 new keywords that have been identified as key search terms via the site's usage analytics.

The spreadsheet illustrated in Figure 1.4 shows what a Goals, Strategies, Objectives, and Tactics (GSOT) artifact might look like. If something the team is considering conflicts with the GSOT document, that topic should be put in the icebox for later consideration or the team will need to meet to agree on amending the GSOT to account for this new change.

You can find a link to a template of the GSOT artifact at www.chaostconcept.com/workshop/GSOT.

- **Personas or “user segments” that are related to each objective.**

Personas serve as a quick way to communicate some basic information about the user that you are trying to serve. When done well, personas are prioritized based on their relative impact on the business. Prioritization at this level helps product teams track and promote work that is most likely to positively impact the objectives.

There are two key types of personas. The first type is backed by data and this is usually simply referred to as a persona. The second type is not backed by data. This type of persona is called a proto-persona and is created using anecdotal information gathered from the stakeholders. The first type, and most valuable to your organization, is the research-backed persona. These personas are built using data that your team has gathered from usage analytics, input from direct conversations with customers, and demographic information from the sales and marketing team. The proto-persona can be valuable as well but only as a placeholder for a full persona. If you don't have the data it's still better to document your team's understanding of your users so that you can begin the process of validating your understanding and identifying areas where

Site Redesign GSOT Document Version: 1.0 Prepared by: Kevin Braun 12/6/2013				
Goals (The business outcome)	Strategies	Objectives	Tactics	
Increase top-line (revenue) by X% by Y Time	Launch a newly designed Website (using leading practices in UX and front-end development) to support the major marketing push of acquiring new clients	Generate X number of qualified new leads within the first Y number of months after launch	SEO to drive organic traffic Paid search to drive targeted traffic Lite design with scrolling Info graphics – is this the right terminology? Potential use of video Responsive design for tablet and mobile use	
		Content lift that provides prospective clients clear messages on services (methods of measuring success - time on site, page views, conversion)	Develop content for Ecommerce and content management solutions – CMX Develop content for Managed Care Develop content for infrastructure Clear calls to action	
		Use Sitecore 7 to showcase the work	Certify staff on Sitecore 7	
Improve bottom-line (operational efficiency)	Create and follow SDLC like 'UX design' process	Be able to complete the requirements phase of a site redesign project within X number of weeks	Review current SDLC and make recommendations for streamlining	

Figure 1.4: A GSOT document serves as the core document in the constitution for your project and everyone on the team should have access to it so they can refer to it when questions arise.

your team's assumptions are incorrect. The UX process isn't a one-time exercise, so iteration on your personas shouldn't be considered a problem. UX is nothing if not a continuous iterative process.

- **Scenarios that are related to each persona.**

Scenarios are very short stories that communicate what the persona is trying to accomplish. Scenarios also serve the purpose of providing context for use cases (more on those shortly) and should be directly related to both a persona and at least one objective.

An example scenario might be something like "Average Dad Brad wants to buy a birthday gift for his son." Notice how short that is. There is no need for a lengthy, overly detailed description that is filled with information that will distract from what really needs to be focused on. In this case we know what persona it serves and that it is related to the objective of purchasing a gift.

These can also be written as user stories if you are working in a Agile environment. It is pretty simple to translate this scenario into a user story. ***As an*** average dad ***I would like to*** be able to purchase a birthday gift for my son ***so that I*** will not be late giving him a gift and he will get what he asked me for.

If your team walks out of the workshop having consensus on those key outcomes, you have succeeded at this stage of the process. With that done, you are ready to move on to documenting the current status and performance of the systems you are working to improve with this project.

State of the Union

Where does your system stand in relation to UX industry standard heuristics as well as when compared to your competition?

Heuristic Evaluations

Heuristics can be thought of as a way of assessing how your product measures up to a set of UX industry standards. The most widely used set of heuristics was created by Jacob Nielsen. You can find an introductory article along with more in-depth information on the Nielsen Norman Group website at:

www.nngroup.com/articles/ten-usability-heuristics

I'll be referencing other information from the Nielsen Norman Group among others throughout this book. In most cases I'll share a link to the original work because each link will also serve as a great resource for further reading, and because I wouldn't want anyone to think these ideas originated with me. I've been on a 20+ year journey learning from many sources and my hope is that you'll find value in accessing those sources directly yourself.

A quick overview of Nielsen's heuristics includes the following 10 key elements to consider:

- **Visibility of system status**

This has been achieved if users are easily able to understand where they are in the system, where they can go, and where they have been. Another element of this has to do with the status of system processes, such as loading a document, saving your work, etc. In general, if users find themselves lost in the system or wondering what is currently happening with the system, this criterion should be noted as needing more attention from your team.

- **Match between system and the real world**

Systems that fail to meet this criterion usually do so because their interface matches the programmatic structure of the system rather than the user expectations (or mental model) of how it should work. An example of this could be a database of items in a grocery store. The items in the grocery store might logically be listed in the database in alphabetical order. This would make perfect sense from a software development point of view, but if we then organized the items in the grocery store using the same logic, a shopper could be standing in an aisle that included paper towels, puppy food, and petunias. Users have come to expect that paper towels would be in an aisle with other paper products like toilet paper and that petunias would be in the florist area of the store, so this would indicate a poor match between the system and the real world.

- **User control and freedom**

In general, a great experience includes being able to explore a system without worrying that you are going to break something or have to spend a bunch of time recovering from a mistake. Users often encounter issues related to this criterion when they find themselves in the situation of needing to ask someone for help because they have taken an action that either resulted in something they don't understand or that was undesirable without knowing how to fix it. Providing a clear way to understand what will happen when an action is taken along with quick methods to undo unwanted actions will help to resolve most issues represented in this criterion.

- **Consistency and standards**

You'll read multiple times in this book that a user's experience is greatly impacted by their ability to scan, read, and comprehend the content within the system. This criterion impacts a user's ability to understand what is on any given page more than almost anything else. Consistent header styles allow the user to understand where to find that information from page to page without having to consciously think about it. Consistency and standards help users know what to expect when moving through the system, and that confidence allows them to focus more of their attention on completing their tasks.

- **Error prevention**

The only thing better than helping users quickly recover from their mistakes is to help them never make a mistake in the first place. One of the best ways to help users avoid

errors is to ensure that you provide clear instructions, content, and messaging written using terms that are familiar to the user.

- **Recognition rather than recall**

I often hear product owners say, “we’ll make sure we provide training for users on that.” Whenever I hear that I think of this criterion because in a perfect system, users would be able to recognize exactly how each element of the system works without training. That’s obviously not always practical for many reasons but ensuring that your system is as intuitive as possible will improve engagement and retention and reduce training-related costs.

- **Flexibility and efficiency of use**

The interface that is the easiest for a beginner to understand will often not be the most efficient interface for an expert user. I make sure to consider this when kicking off a project. If I’m working on a website that users won’t use very often, I’ll lean toward making it as easy to understand as possible. This would include using text labels that take up more room than icons, for example, so that users don’t have to guess what will happen when they click.

If I’m working on software that will be used in a busy call center, I’ll make sure the interface and interactions are as efficient as possible. In some cases that might mean that I use icons instead of buttons with full labels because I can fit more icons on the screen in the same amount of space. In this case new users might not know what to do and potentially make mistakes, but expert users who use the system all day every day as part of their work will appreciate being able to get as much done as possible with the fewest interactions and in the smallest amount of time.

Those examples are on the extreme ends of the spectrum, and for this criterion to truly be met a hybrid approach should be used that provides an interface that is explanatory and intuitive for new users while simultaneously including shortcuts and preference settings for expert users so they are not slowed down by the interface.

- **Aesthetic and minimalist design**

Aesthetics are difficult to quantify because everyone has different taste. To meet the first part of this criterion I look for what I call *thematic appropriateness* in the interface. One example of this is a website that sells baby clothing. Pastel colors, handwritten gestural text in hero images, and baby animal imagery (chicks, ducklings, baby lambs) could all be appropriate thematic choices. There are countless other choices that could be made that would also be appropriate, so when I’m evaluating this, I’m usually looking for elements of the interface that clearly contradict the branding, concept, or theme of the system. I usually encounter issues with the aesthetic part of this criterion in off-brand color choices or the use of buttons and icons that don’t support the overall messaging of the system.

Development frameworks including Bootstrap can have a negative impact on the thematic appropriateness of a website because they make it so easy to get from 0% to

80% done with the design that many teams stop there. Not considering that last 20% can dilute your brand and make a bad impression on customers.

The minimalist design part of this criterion ties in nicely with Edward Tufte's precept about "data to ink." Tufte states that the best data visualizations include only the ink (or pixels) required to communicate the meaning of the data. Over the course of the last decade almost all major operating systems including Microsoft Windows, macOS, iOS, and Android have transitioned a design style called "flat design." The foundation of flat design is the idea that the interface should only comprise the elements required to communicate the meaning of the content and facilitate the interactions needed for users to complete their tasks.

This concept can be taken too far. If designers oversimplify interface elements in an attempt to satisfy the minimalist design criterion, they risk making interactions undiscoverable by removing all affordances.

- **Help users recognize, diagnose, and recover from errors**

If you have ever filled out an online form, you have likely encountered systems that fail at meeting this criterion. When a form is submitted with errors the system should present the user with notifications about what is wrong, where it is wrong, and ideally with instructions that explain how to fix the issues. It's best to message users in context so they can easily make the connection between the error message and the specific item that needs to be fixed.

- **Help and documentation**

Even in the best designed systems users will need documentation to help them learn what features are available and how they work. Users will also need help from time to time. Documentation needs to be comprehensive and easily accessed. Help is best when provided in context with a strong focus on solutions for the immediate need along with information on how to avoid errors in the future.

A heuristic evaluation isn't a replacement for usability testing with representative users, but it is still important because it helps to provide an overview of issues that need to be addressed and can help your team make progress while awaiting qualitative information from usability testing sessions.

Figure 1.5 shows what a heuristic evaluation might look like. You can find a link to a heuristic evaluation template at www.chaostococoncept.com/heuristics.

Expert Reviews

Heuristic evaluations and expert reviews go hand in hand. I'll often perform an expert review before working on the heuristic evaluation so I can become familiar with a system and with the basic user flows that are key to the system's success. This process helps me get my head in the game for kicking off usability testing, for completing a heuristic evaluation, or simply to be able to discuss viable next steps with a new client. Whatever the purpose, this is usually my first step when kicking off a new project.

[Client Name] Heuristic Evaluation - [Date]			
# Heuristic	Yes, No, N/A	Severity	Comments
1000 Visibility of System Status			Location
1001 Does every display begin with a title or header that describes screen contents?	Yes	1	
1002 Is there a consistent icon design scheme and stylistic treatment across the system?	No	2	
1003 Is a single, selected icon clearly visible when surrounded by unselected icons?	N/A	3	
1004 Do menu instructions, prompts, and error messages appear in the same place(s) on each menu?			
1005 In multipage data entry screens, is each page labeled to show its relation to others?			
1006 If overtype and insert mode are both available, is there a visible indication of which one the user is in?			
1007 If pop-up windows are used to display error messages, do they allow the user to see the field in error?			
1008 Is there some form of system feedback for every operator action?			
1009 After the user completes an action (or group of actions), does the feedback indicate that the next group of actions can be started?			
1010 Is there visual feedback in menus or dialog boxes about which choices are selectable?			
1011 Is there visual feedback in menus or dialog boxes about which choice the cursor is on now?			
1012 If multiple options can be selected in a menu or dialog box, is there visual feedback about which options are already selected?			
1013 Is there visual feedback when objects are selected or moved?			
1014 Is the current status of an icon clearly indicated?			
1015 Is there feedback when function keys are pressed?			
1016 If there are observable delays (greater than fifteen seconds) in the system's response time, is the user kept informed of the system's progress?			
1017 Are response times appropriate to the task?			
1018 Typing, cursor motion, mouse selection: 50-150 milliseconds			
1019 Simple, frequent tasks: less than 1 second			
1020 Common tasks: 2-4 seconds			
1021 Complex tasks: 8-12 seconds			
1022 Are response times appropriate to the user's cognitive processing?			
1023 Continuity of thinking is required and information must be remembered throughout several responses: less than two seconds.			
1024 High levels of concentration aren't necessary and remembering information is not required: two to fifteen seconds.			
1025 Is the menu-naming terminology consistent with the user's task domain?			
1026 Does the system provide visibility, that is, by looking, can the user tell the state of the system and the alternatives for action?			
1027 Do GUI menus make obvious which item has been selected?			
1028 Do GUI menus make obvious whether deselection is possible?			
1029 If users must navigate between multiple screens, does the system use content labels, menu maps, and place markers as navigational aids?			
2000 Match Between System and the Real World			
2001 Are icons concrete and familiar?			
2002 Are menu choices ordered in the most logical way, given the user, the item names, and the task variables?			

Figure 1.5: A heuristic evaluation helps customers see exactly where their issues are and where their system stands against a set of industry standard measures.

To successfully complete an expert review, you need to have a solid understanding of what users are trying to achieve when using the system. Next, you should do the research necessary to understand what user expectations are based on industry leaders in the space along with researching best practices for each of the key interactions that make up the user journeys. This gets easier with time, but even very experienced professionals should still take the time to verify that there haven't been advancements in trends or approaches that they are not aware of.

With that foundation in place, I begin the process of walking through each step in each use case and document any issues I find that are likely to impact a user along the way. Oftentimes the issues I find are related to the system not providing enough context or information for the user to be able to make an informed decision about how to proceed. Other issues I frequently encounter include visual hierarchies that are hard to scan, read, or comprehend along with interactions that do not facilitate an efficient experience for the user.

An example of how I might document issues I find while performing an expert review is shown in Figure 1.6. This also illustrates something else that is very important to remember: Don't always do what Apple, Google, or Amazon do without question. They are all amazing teams with a lot more design talent than I personally possess, but that doesn't mean everything they do is right. The figure is an example from the very first page of Apple's Human Interface Guidelines. I've highlighted their guidelines, and where

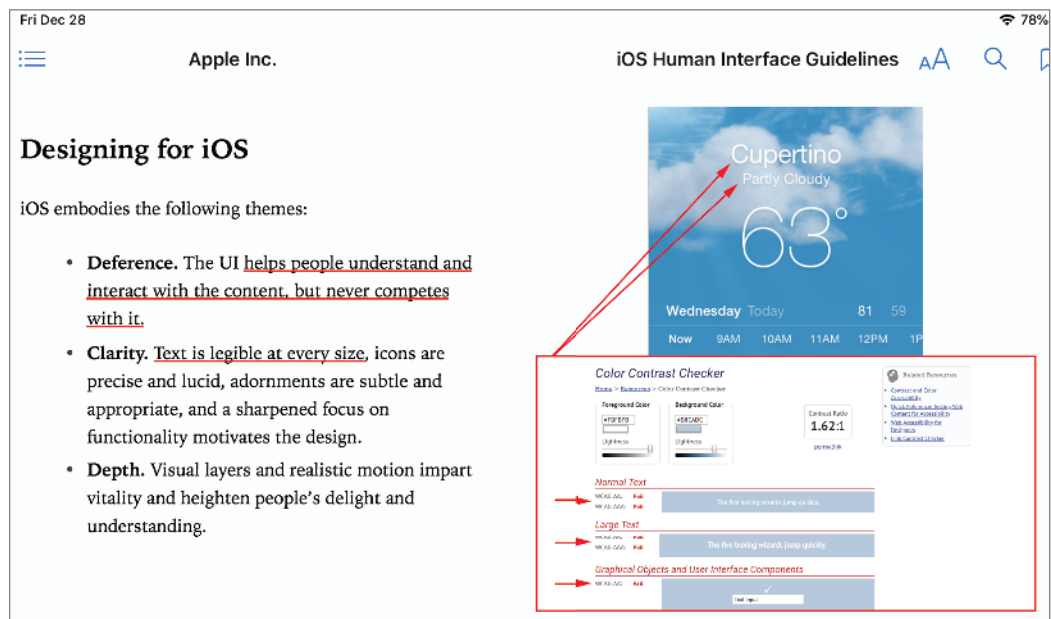


Figure 1.6: I've never worked for Apple but when I was reviewing their Human Interface Guidelines, I was shocked to find this issue on the very first page.

their example of the right way to do it fails basic accessibility testing. When I find issues like this, I document them using a similar format so I can share them with my clients to help in the prioritization process.

The concept of expectancy is very important when trying to understand what users will find easy to use. Oftentimes a user’s repeated interactions with Apple, Amazon, or Google products will lead them to expect your system to function like theirs do. It’s very important to keep this in mind and leverage it where possible to avoid re-creating the wheel. I provided this example to make it clear that all of us should also use our critical thinking skills and not simply follow the leaders blindly.

When I find issues in an expert review, I add them to a spreadsheet that I’ve created (see Figure 1.7) to track problems and help facilitate the discussions on how to resolve them. It’s a fairly simple sheet that helps to keep track of where the issue happened, a brief description of the issue and how to repeat it, a link to an annotated screenshot, potential resolutions, projected impact, and perceived difficulty to resolve. This is my process and it’s always evolving. I offer this version to you as a starting point for creating your own.

[Client Name] Expert Analysis - [Date]					
Section	Issue Description	Screenshot	Potential Resoluti	Impact	Difficulty
Section 1					1
Section 2					2
Section 3					3

Figure 1.7: Like the heuristic evaluation, an expert review document like this can help a team work together on a review and allows customers to easily see and track their issues.

You can find a link to an expert review template at www.chaostoconcept.com/expert-reviews.

Competitive Analysis

Other helpful information can be gathered by performing a competitive analysis. Understanding how your solution is functioning in relation to your competition consists of the four key elements I’ll discuss here. Most companies cannot accurately report on all four, so don’t get discouraged if you start off with only a fraction of what is recommended. Even the preamble to the Constitution of the United States recognizes that nothing starts off perfectly, and also acknowledges that perfection may not be possible by stating “... in order to form a *more* perfect union.” No matter what you are building, take comfort in knowing that no part of the UX process is a one-time fix. Iteration based on data creates a solid foundation and sets your team up with a repeatable process for continuous improvement.

The key elements of a competitive analysis include establishing a strong understanding of the Market Share, Product Perception, Feature Matrix, and Value.

Market Share

The formula to calculate your market share is pretty simple:

Market share = Your total revenues divided by the industry's total sales during a fixed period

To determine what an industry's value is it will be easiest to review reports put out by industry organizations or look at the annual reports of some of the biggest players in the industry.

If your team can accurately understand market share and penetration, you'll be able to make strategic shifts in how your business is run. Early in the game acquisition is essential, while later in your product's life cycle, hopefully you'll be able to shift to a more defensive retention focus. Either way the first step is to gather the information required to understand your position. The sales and marketing teams will likely be your best source of this info. If you are working at a small startup, the founders are your best bet and hopefully you're already working very closely with them.

Product Perception

Your team can track brand perception in a number of different ways. The methods I've seen most frequently used within product design teams are the Net Promoter Score and Google's HEART framework.

The Net Promoter Score (NPS) is a method used by companies around the world to gauge the relative happiness of their customers on a scale from 0 to 10. You are not likely to be able to get hold of your competitors' NPS scores, but you can get some general industry benchmarks.

Retently publishes a set of benchmarks that can be viewed on their site at: www.retently.com/blog/good-net-promoter-score.

To gather the data for your NPS, participants are asked, "On a scale of 0 to 10, how likely are you to recommend this company's product or service to a friend or colleague?"

0–6. Participants who rate your product between 0 and 6 are called *detractors* in this system. Detractors are not very happy with your product and they won't likely purchase again or recommend your product.

7–8. Participants who rate your product 7 or 8 are considered to be *passive*. Passive participants are not actively upset with your product but are not overly satisfied with it either. These participants are likely to switch to another product, and if they recommend your product it will likely be with caveats.

9–10. Participants who rate your product 9 or 10 are labeled *promoters*. These participants are happy with your product and are likely to unconditionally recommend it.

This question is often asked at the end of a survey while gathering other info. Once you have gathered your scores, you'll need to calculate them using the following formula:

$$\text{NPS} = \% \text{ of promoters} - \% \text{ of detractors}$$

To calculate that you'll first need to know the % that were promoters by using the following formula:

$$\% \text{ of promoters} = \# \text{ of promoters} / \# \text{ of respondents}$$

To calculate the % of detractors you would need to use the following formula:

$$\% \text{ of detractors} = \# \text{ of detractors} / \# \text{ of respondents}$$

Once you have that done, you can go back and plug the numbers into the NPS score formula and perform the final calculation.

Now that you see how it works, it is important to know that you don't need to calculate it the hard way. You can find several online calculators, including this one from SurveyMonkey: www.surveymonkey.com/mp/nps-calculator.

Google's HEART Framework

In this context, HEART stands for Happiness, Engagement, Adoption, Retention, and Task Success (see Figure 1.8). This is one of Google's tools for measuring how their solutions are performing. This approach combines some of the other methods we have reviewed with new methods, resulting in a comprehensive framework. This is another one of those systems that could be difficult to set up within your organization, but it can be worth it if you have a team that is committed to monitoring and maintaining it. In the organization I was working at when I was introduced to this method, it was set up with a dashboard for various teams to monitor. I believe members of the executive team were very excited by this dashboard because it was the first time they could actually see on one page a relatively real-time view of how their system was performing. There were flaws in our execution, and keeping the data current and complete proved to be too much of a challenge for the team over time, but what I saw while it was running was enough to convince me of the value.

What I like about this framework is how closely it ties to the overall approach outlined in this book. There is a strong focus on defined goals and measuring your progress, so adapting this framework is pretty straightforward.

Similar to a lot of what you read in this book, any effort in the right direction is better than no effort. If your team can only manage to accurately and consistently report on a small piece of the information that makes up the HEART framework, you are still better off than if you don't measure any of it. Adoption is an easy place to start, followed by engagement and retention. Every situation is different, so I'm going to summarize all of them and provide a direct link to Google's info on the subject so you can dig deeper if you are interested.

Happiness

You can measure happiness in a few different ways. You can ask questions about satisfaction via a survey. You can rely on the Net Promoter Score that we discussed previously. You can also track social media and user forums to gather sentiment information. Whatever methods you choose, gathering information about happiness will allow you to better understand the impact of issues you may uncover elsewhere. If you have a technical issue and your happiness indicators don't change all that much, then you can gauge future responses to that type of issue more accurately. If, on the other hand, something in your system that your team believed was trivial turns up on a user forum or Twitter as being a big problem and you see that other users are agreeing, sharing, or creating a hashtag about it, then you know your team needs to reprioritize that issue.

I've personally used the information gathered via the ForeSee pop-up feedback prompt to identify and raise issues for prioritization, so I know that direct information about customer sentiment can be very helpful.

Engagement

Engagement is an interesting thing to track. If you ask most product owners today, you'll likely hear that they want their product to be addictive. What this usually means is that they want users to engage with their product every day and ideally multiple times a day.

That actually doesn't always make much sense. If a user is an executive who is checking in on how the operations are going across the country, she might not really need to do that every day. It's possible that checking in at this level twice a week might make more sense for her. It's also possible that 30 seconds of reviewing a dashboard twice a week under normal circumstances represents a wonderful user experience for the executive in question.

I understand that your business wants to be able to illustrate strong engagement, but I don't recommend forcing the issue. If you have a compelling user need that you can address on a daily basis, by all means work toward that. If not, stating daily usage as a

goal can distract from goals that could be achieved and can discourage your teams by needlessly directing them to work on something that is likely unattainable.

My suggestion is to base your initial engagement goals on anticipated user needs and then work from there to add value that might drive more engagement over time.

Adoption

As I mentioned before, adoption should be relatively simple to monitor. You'll be looking to track new signups, new trials, etc., as a way of understanding how many new customers you have in the system week to week, month over month, and year over year. Keeping track of adoption is a great indicator of how well your product is fitting within your market.

Retention

This is all about keeping the users you already have because they are usually the most valuable to your organization. This often gets measured on a spectrum. You might report on the number of users who log in daily as part of engagement and use that same method to report on those users who have logged in at least once in the past month as being retained. However your business decides to measure it, retention is all about tracking your success at keeping users and is an essential part of understanding how changes to your offerings are impacting the user experience.

Task Success

Task success broadly relates to what most people think of as a conversion. Conversions are thought of as the end goal of a user's interactions with a system. An example of a conversion might be a completed checkout on an ecommerce site. Another example would be a contact form submission on a marketing site. Defining your key tasks and what the outcome should be is an essential part of measuring and improving on your user experience.

To read a lot more about Google's HEART framework, visit this URL:

ai.google/research/pubs/pub36299

Feature Matrix

Another tool that is essential to understanding how your product or service compares with the competition is a feature comparison matrix. This is usually a simple spreadsheet that starts off with a comprehensive list of features offered in the market in one

column on the far left and then a list including your product and all of your competitors each in their own cell in the top row of the spreadsheet. Below each product name is usually a check for each feature that is offered in that product. If a feature isn't offered, that cell is left empty. Once you have completed the matrix, it is very easy to see how all the products in the market relate to one another in terms of features as shown in Figure 1.9.

A more complex version of this takes into account the relative priority each feature has in relation to your most valuable user's scenarios. To calculate this, you first need to rank your personas by their relative value to your organization. This is most often tied to revenue, but for some companies influencers are more important early on because they are more focused on attracting and building audience, etc. Either way it's important to rank your personas so that you can make design decisions based on their value. We'll discuss creating personas and prioritizing them in detail later in the book, but for the sake of this conversation just know that it's important to have them because they provide a context for understanding your position in the market.

Once you have the personas ranked, you'll need to have a list of the features that are important to your most valuable users. You can gather that list of features by itemizing what features are required to complete each scenario that is associated with that persona.

If your most important users need a way to find a ride on demand, such as when using an app like Uber, they will need at least the following features to satisfy that scenario:

- 1 Geolocation so the app knows where to send the car for the pickup
- 2 An account associated with payment information
- 3 The ability for users to enter a start and ending location
- 4 A time and cost estimate
- 5 A ride type selector
- 6 A confirmation and pickup time estimation

Once you have that information, you can assign a higher value to the features on the list so that when you complete your matrix you'll have a clearer picture of how your solution compares to your competition when it comes to what your most valuable users need.

You can find a link to a feature matrix template at www.chaostoconcept.com/feature-matrix.

Value

In my experience, value is very rarely calculated because many think that it's too subjective or inaccurate. I believe this is going to be an essential metric as UX and

Competition Comparison Document ☆  Share					
File Edit View Insert Format Data Tools Add-ons Help All changes saved in Drive					
100% 11 Calibri Undo Redo Bold Italic Text Color Background Color					
	A	B	C	D	E
1	Competition Comparison Document				
2	Features	Acme Product #1	Acme Product #2	Competitor Product #1	
3	Feature #1 (25 pts)	X		X	
4	Feature #2 (25 pts)	X	X	X	
5	Feature #3 (20 pts)	X	X	X	
6	Feature #4 (15 pts)	X	X		
7	Feature #5 (15 pts)	X			
8	Score	100	60	70	

Figure 1.9: A feature matrix is a great way to document how your product compares to others in the market. It is also a great way to visualize the specifics of what makes the market leaders stand out.

product teams mature. In hopes of spearheading the conversation on this topic, I offer the following formula:

I'm about to geek out here for a bit and what I'm about to describe isn't established best practice so if you aren't interested in learning about this at the moment, skip ahead to page 28.

Value = The prioritized and weighted product features divided by the Normalized Price (yearly, monthly, etc.) across the market

This sounds more complicated than it really is. If you completed the advanced feature matrix reviewed previously, you already have the prioritized and weighted features. The normalized price is calculated by looking at all the competition's pricing models and converting each so that they are all being measured in the same way.

If your product offers only a monthly subscription, you would start off by saying that the normalized price is \$15.00 per month.

If Competitor A has a monthly and yearly pricing plan, you would use the one that adds up to the cheapest monthly payment. For this example, we will say that their price is \$20.00 per month.

If Competitor B offers a one-time purchase price, we'll assume a lifetime for this product to be four years before a major upgrade that requires another payment. Let's say that their one-time price is \$199.00. You would then divide that by 48 months and get a normalized price of \$4.14 a month. Obviously this price would seem to be a big advantage but a lump sum payment of \$199.00 can be a substantial barrier to entry, especially for a consumer product, so that needs to be factored in.

This is where you'll likely need to make some adjustments based on the relative price points, but for this example I would say any product with a one-time price greater than \$25.00 and less than \$50.00 would get a \$5.00 a month penalty with an additional \$5.00 a month penalty added for each \$50.00 above that. For this example, the \$199.00 one-time price would normalize to \$4.14 a month and would receive a penalty of \$15.00 a month, making the total adjusted monthly cost \$19.14 per month.

The adjusted price breakdown looks like this:

$\$199.00 / 48 = \4.14

One-time price between \$25.00 and \$50.00 = \$5.00 penalty

One-time price additional penalty for the next \$50.00 = an additional \$5.00 penalty

One-time price additional penalty for the next \$50.00 = an additional \$5.00 penalty

Since \$199.00 is less than \$200.00 we stop there and add the penalty to the normalized price and end up with the \$19.14 mentioned previously.

With this calculation done, our market normalized price range is:

Your product: \$15.00 per month

Competitor A: \$20.00 per month

Competitor B: \$19.14 per month

Now that we have the normalized prices, we need the values for the prioritized and weighted features. As mentioned previously, in the advanced feature matrix process we prioritize and weight our features by defining those features that are most important to our most valuable personas or user segments. We'll discuss creating personas and prioritizing them in detail later in the book, but for the sake of this conversation just know that it's important to have them because they provide a context for understanding the value of various features.

For this example, let's say we added up the feature value points for your product and found that it scored 17. Each feature your product has is worth 1 point. Your product offers a total of 12 of the 16 features available in your market so you start off with 12 points. Five of those features are essential to your most valuable users, so each of those gets an extra point.

With the weighted features accounted for, those 5 features add up to 10 points.

The remaining 7 standard features add up to 7 points.

Adding these together we end up with a total of 17 feature points for your product.

Let's assume we completed the same process for the other two products and ended up with 20 feature points for Competitor A and 12 feature points for Competitor B.

With all that done, we are ready to plug the numbers into the value formula to better understand our product's relative value. The higher the value index, the better.

Your Product Value:

$$17 / 15.00 = 1.13$$

Competitor A's Product Value:

$$20 / 20.00 = 1.00$$

Competitor B's Product Value:

$$12 / 19.14 = .62$$

I believe this type of value scoring is important because it provides a clear way to understand your position in the market and also establishes a clear link between price and functionality in relation to what your most valuable customers need.

Benchmarking, KPIs, and OKRs

Benchmarking, internal and external, is pretty simple. It is also essential to the iterative design process because it is the foundation for how we'll track our progress. In order to establish a benchmark, you'll be tracking how your competition compares to your product in relation to very specific conversion goals.

Documenting the number of clicks a user must initiate to complete the checkout process in an ecommerce site is a great example of a competitive benchmark. It would be relatively easy to visit each of your competitors, add a product to the cart, and then start counting the number of clicks it takes to complete that checkout. You would then do the same thing with your other competitor sites. Once you have all those clicks to complete benchmark numbers you can compare them to your own product to understand and track your relative performance. There are other ways to track this sort of thing, such as measuring the time it takes a user to complete a task or measuring the number of mistakes a user makes while completing a use case.

KPIs (Key Performance Indicators)

Monitoring the number of successful task completions (conversion rate) or the number of customers who added at least one item to the cart (micro conversion) is an example of tracking your KPIs. For non-ecommerce systems, you could track the end state of any meaningful workflow as a conversion along with important substeps in a workflow as micro conversion.

Tracking all the micro conversion and the final conversion in a single workflow can be thought of as a *conversion funnel*. All this means is that you are tracking each step in a workflow and monitoring the number of users who successfully complete each step. Because these are usually linear processes, the fall off rate (number of users who drop out of the process at any given step) usually resembles a funnel if you were to illustrate it. To picture this, imagine that you have 100 users visit your site (the wide end of the funnel) and that 50 of them visited a product page. In this case the product page visit micro conversion rate would be 50%. Now imagine that of those 50 product page visits 10 of those users added a product to their cart. That would mean you have a 10% add to cart micro conversion rate. Finally, imagine that 1 of those 10 that added a product to the cart went on to finish the checkout process. That would mean that your overall conversion rate is 1% and that represents the narrow end of the funnel as seen in Figure 1.10.

Tracking these types of indicators will help your team understand the overall performance of your system and will also provide vision into whether the changes you are making along the way are helping or hurting your business. This is very important because your iterations will rely on an accurate flow or performance data to provide direction.



Figure 1.10: Conversion funnels are an industry-standard way of visualizing how well your system is moving users through the process.

OKRs (Objectives and Key Results)

Another term you will hear is OKR, or Objectives and Key Results. This process is essentially the same as what I described when introducing Goals, Strategies, Objectives, and Tactics. The idea is to define your objectives and make sure you have a way of measuring success or failure. One addition that I'll make based on the following article from Google Ventures (library.gv.com/how-google-sets-goals-okrs-a1f69b0b72c7) is that there is a sweet spot in your success rate. If you are 60% to 70% successful, you're doing great. That means that you have moved the needle in the right direction and that your objectives were sufficiently challenging. If you are always scoring 90% to 100% you can be happy about the progress, but it also likely indicates that your objectives aren't ambitious enough. This isn't a new system, but in many ways it feels like it is just now starting to gain widespread acceptance.

