



Slide to unlock



UI

Overhaul a UI Design Without Upsetting Current Users

by Scott Plewes, Didier Thizy



Companies with successful software products often know when their user interface could be improved, but are concerned that any sort of major change might upset their user base.

“Not only would a redesign cost me a small fortune” one executive told us, “but look at the backlash against Google and Facebook’s changes. I’m not willing to risk losing customers over design.”

INDEED, GOOGLE, TWITTER AND FACEBOOK caused quite a stir recently by making dramatic changes to their user interfaces. In less than 24 hours, hundreds of users formed a coalition against the new Gmail, with a mission to “urge Google to fix the horrid new Gmail interface and take us back to productivity”. Facebook users began posting “If it Ain’t Broke, Don’t Fix It” banners to express their dissatisfaction.

But for every example of design gone wrong, there is a success story of a company blasting ahead of its competitors by reinventing its product design. Salesforce.com’s dominance in the CRM market, for example, is largely attributed to its innovative design updates.

We strongly believe that the risks associated with not updating your User Interface (UI) outweigh those associated with an update. But you absolutely need to use the right product management and User Experience (UX) design techniques to evolve your product carefully and avoid a user revolt.

In this paper, we focus on key lessons for how to approach a major design update. Do it right and you’re on the path to capture new markets, increase customer satisfaction and retention, and pull ahead of the competition.



Focus on Form AND Function

HERE'S A CRAZY EXPERIMENT. You are the owner of the Green Bay Packers. Currently the best team in football. You don't like paying Aaron Rodgers, Greg Jennings and the rest of the team all that money. So you replace them with guys who closely resemble in looks the current team, and you field them. They have the same jerseys, line up correctly and, to the untrained eye, look like they are the Packers.

We all know this experiment would fool people until the ball is snapped on the first play from scrimmage; and then we'd have to do a body count.

No one would expect a team that just looks like the Packers to play like the Packers. But companies try this when they are designing products all the time. They look at the iPhone or some other product they want to emulate, decide they'll make their product look as much like this as possible and are surprised when, on product release, they are sacked 20 yards behind the line of scrimmage.

The point about the Packers and software products is the same. 90% of the value of the experience comes in how they work, not how they look. Steve Jobs would emphasize this point all the time at Apple. Of course, it's not that how it looks isn't important. It absolutely is. It's just not nearly enough.

This is one of the major complaints about Gmail's new interface. "Gmail's new interface is form over function. Strange because the reason everyone went to Gmail was function" says one Twitter user. "Gmail's new look is uncluttered, which is a good thing until you try to navigate and then, user beware".

It is a myth that the essence of a fantastic experience and product success comes from how it looks. The importance of looks may hold true for a second or two. And it might be a very important second or two in

purchasing decisions. However, if you are looking to have your product be endorsed by the market, you better spend time on much more than how its looks. Especially in a world where there are so many ways for product information and assessments to travel quickly. If someone actually ran that experiment with the Packers, the whole world would know in a matter of minutes.

A product redesign that looks good, but doesn't deliver, faces the same challenge; rapid spread of its failure to deliver the experience promised by how good it looked.

Remember, anyone can be an armchair quarterback, but to drop back and thread a pass through a zone defence while 280 pound giants are bearing down on you requires more than just looking good in your uniform. Never make the mistake that what a product looks like is the same as what a product experience actually is.



Redesign Based on Evidence

IN ORDER TO MAKE FUNCTIONAL (not just aesthetic) improvements, product managers and teams need to perform product and design research with real-world users. Most people agree to this in principle, but vary wildly in how they execute.

One of the most common mistake companies make is to implement UI changes based on what users "say" they want. Unfortunately, research shows that what people say they will (or won't) like doesn't always match reality. Moreover, a user's suggestion on how to solve a UI problem may be unworkable, unreasonable, or both.

While it's important to solicit and listen to user feedback, users aren't designers or usability experts so their complaints and suggestions should not be taken verbatim. Don't confuse a design suggestion with a design solution.

To identify the real UI problems and solutions, you need to perform careful UX research that takes facts – not opinions – into account.

There is a variety of ways to perform correct design research. Our website offers a [quick reference](#) for some of the most time-effective methods, including protocol-based requirements gather, stakeholder interviews and paper prototyping. But two fundamental techniques that should be used in any major user interface update are focus groups and usability testing.

Focus Groups

Focus groups are a powerful exploratory tool that can provide rich information on the opinions and attitudes of the target audience towards an existing product.

“

...what we say we will do (intent) and what we actually do (action or behavior) often vary because we can't predict contextual factors which may alter our behavior.”

They are best used in the early Discovery and Assessment stages of a design update project in order to:

- get a feel for the range of opinions on different aspects of the interface;
- understand the reasons for change requests, new interface ideas, and underlying preferences;
- build a raw list of user requirements.

This information can provide direction to a development team in the early stages of development, so that the development team can prepare and even start development in key areas in parallel with the design effort.

Focus groups however are not a good source of behavioral data. What people say they will do, compared to what they actually do, is notoriously unreliable.

There are several reasons for this. First, what we say we will do (intent) and what we actually do (action or behavior) often vary because we can't predict *contextual factors* which may alter our behavior in a given situation. This is why we all like test drives, or 30 day return policies.

Second, we are not fully aware of all of our behaviors—some everyday actions are so commonplace that we sometimes forget we even do them. For instance, if you ask people something as simple as “what features on a telephone do you use and approximately how often?” and compare the results to direct observation data on the same question, the results are usually staggeringly different.

That's where observational techniques like usability testing come in.

Usability Testing

Usability testing is a technique that designers use to determine the effectiveness of a design by observing users walk through a pre-determined series of tasks in the software, and assessing their performance. For example, if users consistently stumble on a particular part of the workflow, this is a strong indicator that the design is flawed.

The designer usually follows up each usability test by interviewing the user and asking more qualitative questions around their experience.

Usability metrics—like time to complete task, number of errors, number of clicks, success rate—are collected along with ease of use ratings.

Note that usability testing is very different from beta testing. In beta testing, users will typically only report usability problems that make it nearly impossible for them to accomplish a task, or, in other words, things that are very clearly a bug. They typically won't report that they found something difficult or unintuitive. People don't always like to admit that they failed at something. Also, beta testers, or at least the ones who take time

to report issues, are often fans of the product, and by nature, power users. They may have learned to work around or ignore usability issues.

Hopefully, by the time you release to beta, you have solved most of the obvious usability issues, because the most expensive time to fix a usability issue (or any bug for that matter) is in beta.

Combining Focus Groups And Usability Testing

While Focus Groups are best used early on, usability testing is should be scheduled throughout development to provide metrics on how effectively the target audience can use the new design as it evolves.

Using each technique appropriately (as summarized by the table below) provides lightweight but thorough validation that the design will meet the needs of your current users and those you wish to attract with the redesign.

| | FOCUS GROUPS | USABILITY TESTING |
|----------------|--|---|
| METHOD | Group discussion with 6-8 participants | Individual testing of 6-8 participants |
| WHEN TO USE | Pre-development, early in program | Throughout development (concept, detailed design, verification) |
| KEY MOTIVATION | What are customers' wants and desires? | How well can end users use the functionality a product provides? |
| KEY FOCUS | What will motivate potential customers to buy the product? | What will make the product easy for end users to learn and use? |
| OUTPUT | Opinions, attitudes, preferences | Behavioral measures and preferences based on usage, ease of use ratings |

When undertaking a major design update, there are hundreds of different directions in which you could go. But what are the most important areas of the product that need improvement, and how can you ensure the changes you make will be well accepted by users?

The key is to ensure that these choices are based on evidence, not gut feelings or personal opinions. User research techniques such as focus groups and usability testing will identify the most important tasks and UI areas to address, and help you update those areas for maximum ROI.



Stop the Endless UI Debates

RESEARCH CAN BE VERY EFFECTIVE for identifying useful UI ideas, but simply having a good idea is not enough. For example, your company might discover it would be advantageous to include GPS functionality, a calendar app, and game section in your smartphone design. If the design and implementation of those features makes them difficult to access, however, you will lose much of the value you set out to provide.

Don't leave the implementation of a UI design idea to chance! Involve the development team early in the process to ensure the idea is translated properly into a final product. Our related paper, [How To Get Amazing Software Out The Door Fast](#), details how product managers, designers and developers can work as an integrated team to deliver software that exceeds the expectations set out at the initial concept stage.

In this paper however, we'd like to focus on one key issue that can bog down any team during development, no matter what process they use or how integrated they are – UI design debates.

We've all been there – what starts as a UI review meeting becomes an endless argument about where a button should be positioned, or whether the text should be two sizes larger or smaller. Everyone has an opinion, based on past experience, a book they've read, or simply personal preference. How do you get past the arguing, make a decision, and continue?

Be Clear About The Value You Are Bringing In The New Version

Take the team back to first principles. When arguing over a particular feature or screen, ensure the team can all answer the following questions:

- Why will a user want to use this feature?

- What will be their most common tasks?
- Under what conditions, or in what context, will this feature be used?
- Who is it that will primarily use this particular feature?"

If you don't have at least the beginnings of answers to these questions up front, before you start to design, then what is it that is directing the design of the product? Reflect on these questions for a moment. They are much more than just a list of functionality. In fact, one of the things they will help the design and development team do is prioritize functionality.

Taking just a bit of time to answer these questions up front, and to make sure everyone on the team knows the answers to these questions, will really speed things up. If you don't have these questions answered they will inevitably come up again and again in the design and development process. Often we see teams debating whether to include a certain feature. What you're really debating in this case is "who is our primary user and what would they value?" If you considered this question earlier, the answer would be clear, and you'll breeze through those discussions.

Take It To The User

I've been involved in UI design for almost 20 years, and one thing I see time and again is—no matter how smart or talented the people involved in the design and development of the product (and I've met a lot of smart and talented people)—they inevitably miss something about how a user will behave with a certain part of the product design, and sometimes it's a critical aspect of the design.

A common mistake is to iterate, and iterate, and iterate again using internal feedback to try and get it right. There is no substitute for feedback from real users.

Take the team back to the product management and design research data. As Steve Johnson from Pragmatic Marketing would say "Your opinion, while interesting, is irrelevant."

Try A/B Testing

Often the debate is between 2 proposed solutions. This could be a good candidate for A/B testing, a fast form of user research where you launch two designs to two separate user groups and compare the results.

A/B testing can help you measure which of two or more designs produces the most conversions, fewest clicks, fastest time, or whatever metric you've decided to measure. Online testing tools like [verifyapp.com](#) take advantage of crowdsourcing to make this process even more convenient.

There are also drawbacks to A/B testing. In particular, you have to rely on your best guess as to the real reason why design A performed better than design B. You don't get any feedback on whether or not the user "gets" the system.

As well, you are likely to test yourself to a non-optimal solution. Incremental A/B testing will find a solution that, relative to the other solutions that were presented, produced the best results. However, because it is always relative testing, you run the risk of getting stuck in a **local maximum** and ultimately cheating yourself out of the best solution.

A/B testing should be used as a quick cheat, a complement to more qualitative forms of research like the focus groups and usability testing we described earlier.

Attain Excellence, Not Perfection

We all know from school that attaining perfection, as opposed to attaining excellence, might not be worth the effort. If getting from 98% to 100% takes twice as long, most of us will be happy with 98%. In user experience design there's another reason beyond this to not try for perfection.

Once you've hit the targets you set for the user experience design, and the overall UI design has been demonstrably shown to be excellent through usability testing or other formal testing techniques, trying to "fix up" small aspects of the UI design can actually undermine the whole design.

One of the reasons we keep our UI designers involved all through development and right up to release is to ensure that the development team follows the overall direction set for the UI. For instance, a developer may have an idea that makes a single screen better by changing a control, in search of the "perfect" screen. However, it may be inconsistent with the rest of the UI design and therefore it may actually set the design backwards (a phenomenon known as "Design Drift")

As you read the above you might have said to yourself "tell me something I don't know". However, we find most experienced product teams know these techniques in principle, but find it tough to put into practice with discipline. We're all human, and when it comes to design we love to debate and put in our two cents. This takes time—time that could be spent creating and delivering great design instead of debating it.



Handling Design Complaints— Interpret and Prioritize

NOTHING IS UNIVERSALLY LOVED. When you're evaluating user complaints about your UI (there will always be some!) don't react to them in a knee-jerk fashion.

Experienced product managers and user experience designers will gather qualitative and quantitative data from users in order to determine:

- how widespread each design issue is;
- the impact of an issue on users;
- the relative importance of one issue over another.

This data and analysis allows teams to make sense of all the user reactions, so that they can swiftly address issues that will most affect their product (and ultimately business) success and put aside all the red herrings.

How Widespread is the Complaint?

The complaints you get may be identifying a widespread problem, but they could also simply be gripes from a very small but vocal group of users.

This is where quantitative data becomes very useful. When Google changed its user interface, it added a pop-up bar in the corner of the screen that asks users for feedback on the new design at any time.



Figure 1: Google prompts for feedback on their new design

“

When evaluating user reactions, teams need to analyze the *impact* of the feedback rather than the emotion behind it.”

This is an effective mechanism for determining how widespread design problems are, and as a bonus allows users a sense of control as they “vent” their frustrations in direct communication to the software maker.

However, this method can be costly as it requires staff to sort through the wide variety of comments. The quality of the data provided is also dubious—users will often offer solutions rather than explaining the “why” behind their frustration.

For more cost-effective and equally powerful data gathering, consider conducting a random survey of your targeted audience through user interviews or ethnographic study. This provides the quality and quantity of feedback necessary to ascertain whether you’re dealing with a common complaint that should be immediately addressed or a minor issue that should be lower on the priority list.

Who Is Affected?

Product Managers familiar with the Pragmatic Marketing program know the value of establishing Personas upfront—Buyer Personas typically represent the typical decision makers who buy the product, and User Personas represent the major user types.

User Experience designers take this concept a step further and prioritize Primary, Secondary and even Tertiary personas. These indicate, among other things, how common each type of user is and how *important* they are to your business.

For example, a healthcare software application’s primary users might be cardiologists, with secondary users defined as physicians in other specialties as well as hospital support staff.

If a complaint is coming frequently from individuals in the Primary audience, this is a strong indicator to prioritize ahead of complaints from the Secondary audience.

What Is The Actual Impact?

Design changes typically impact two main areas: the UI design and the user workflow. You’ll find users will react very differently to a UI change that affects the visual design versus one that affects their workflow.

Visual UI changes tend to provoke instant, emotional reactions such as “That’s gorgeous!” or “That’s hideous!” Reactions to workflow changes, however, take slightly longer to emerge and normally take longer to adapt to.

When evaluating user reactions, teams need to analyze the *impact* of the feedback rather than the emotion behind it. For example, does “That’s hideous!” mean “Now I won’t renew my subscription,” or does it simply mean “I wouldn’t have chosen those colors.”

Some of the techniques outlined in this paper don’t necessarily need to be performed by a user experience professional. However, judging the

impact of user reactions is something that is dramatically enhanced by experience.

Weigh The Short-Term & Long-Term Impacts

Sometimes a design change will result in short term user pain but long-term gain. This is particularly true when it comes to workflow changes.

Let’s say product management identifies that an automated data entry feature would help users complete a key task more quickly. When it’s first rolled out, your users may object to it. They’re used to entering data in a certain way (even if that way is time consuming and inefficient) and may be upset that they now need to “re-learn” how to do something they were already familiar with. After two or three days, however, they will realize they can complete a form in half the time and begin to appreciate the change.

Facebook may have had this principle in mind when they changed their user interface. Yes there was an initial backlash, but how much are you hearing about it now? You may have even been one of the people initially upset, are you still?

Unfortunately, quickly determining whether the issue reported is superficial or a sign of a deeper problem is a complex mix of analysis and instinct that user experience designers train for years to master.



Right Expertise, Right Process

UP UNTIL NOW YOU MAY BE NODDING, agreeing in principle with the recommendations we've presented. Then you look at the progress your team has made so far and notice a gap.

Unfortunately, as the demand for improved design and usability increases, we see companies jumping head first into a train wreck—spending top dollar to hire designers and build a UX team, but making frustratingly little progress.

There are a whole host of reasons why this happens, but we've found the root cause tends to have something to do with having the wrong skills, or someone too junior, on the job. More specifically, it tends to be one of the following 3 reasons:

Hiring Visual Designers To Be Interaction Designers

Great visual designers are essential. The good ones know how to present visual information in a way your users immediately find compelling and simple. They help make the potentially confounding analytics graphs in your product easy to understand. However, they're not superheroes that can move effortlessly from being an expert in visual language, to an expert in interaction design—just by jumping in a phone booth. These are two completely different disciplines.

Generally speaking, visual designers do not study information design and workflow design, nor do they specialize in taking user research and translating it into the appropriate interaction mechanisms. If you're staffing your team with only visual designers, I can safely bet you're frustrated because your product still has usability issues, and users are still confounded by certain features and/or workflows.

The opposite would be true if you hired only interaction designers—you would likely have a very usable product with dated icons and no visual cohesiveness. But that scenario is far less common. Visual designers and Interaction designers go together like Batman and Robin—you can't have one without the other if you want an unbeatable dynamic duo.

Putting An Engineer (Or A Marketer, Or Anyone Else Without A Design Background) In Charge Of UX Design

It's hard to coach a football team if you've never played the game. You can be an expert strategist, a great motivator, and maybe even an expert participant in another sport. But there's a reason football teams don't hire rugby coaches when they want to win championships.

The same is true for succeeding at any function within a company. You don't hire a VP of Sales to be your VP of Engineering. UX design is no different from any other key function. If you want to successfully improve UX at a company, or division level, you need someone with experience in charge driving design at a strategic level, and looking out for all the integration points—or it will constantly be an afterthought.

Designers and Developers That Aren't Used to Working Together

Sometimes there is a disconnect between design and development teams, which starts in the way their roles are defined.

Designers are looking to design the best possible experience for the user. This group is measured on design success and, to some degree, product success in the market.

Developers are measured on speed and quality. They need to deliver on time and on-budget and their code needs to be bug free (which only adds to the on-time problem, since bug-free code takes time).

Perhaps the most challenging aspect of modern software creation is having multiple team members—designers, developers, architects—working iteratively and in parallel. Wouldn't it just be easier if the Product Manager completed their requirements definition, then handed it off to designers to create the full design, who then handed it off to development to build and test?

Easier? Maybe. But certainly not more successful.

Understanding the main differences between roles is key. Developers are not pushing back on feature requests because they're stubborn, but because they're responsible for maintaining timeline and quality. A designer doesn't look to change the design "yet again" because he gets a giddy thrill each time he pushes out the timeline—it is because he needs to make sure that users will love the product.

To ensure your teams understand each other, consider investing in integrated design and development training. Our related paper, [How To](#)

[Get Amazing Software Out The Door Fast](#), provides a lot of detail on how designers and developers can work together, as one team.

If any of these sound familiar, you're in good company. In the rush to use design as a competitive edge, many great companies are diving in and making one or more of these missteps. The good news is they are very fixable with deliberate action:

1. Hire or contract interaction designers to work with your visual designers. How many of each you need really depends on the nature of your product, but you need a mix.
2. Make sure whoever has the reins has the training and experience to lead the team, and represent design at the decision table.
3. Ensure your designers and developers are working together as an integrated team. Don't underestimate the value of training, process and communication here.

With products like the Apple iPad and Salesforce.com setting a new bar for software, vendors are recognizing an opportunity to update their product's user interface as a means of furthering their success. But jumping into a design update without the proper expertise can be very costly, in terms of the design process and, even worse, if the updated product causes a user backlash.

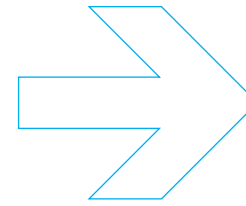
We'd like to invite readers to consider what will happen if you DON'T make a change.

Sometimes, when a product is successful, the risk of "messing with a good thing" is judged to be too high. This thinking however, can lull companies into complacency and cause products to fall behind in the marketplace. While there is always a certain amount of risk in a UI update, there is now a larger risk that your product will stagnate and become outdated without a change. If you don't take the risk of improving your user experience, chances are a competitor will.

In this paper, we've outlined key points for ensuring a successful design update that resonates with existing users and helps capture new ones:

- Focus on updating form and function, in other words the aesthetic aspects as well as the usability of the product
- Ensure every decision is driven by the right research techniques with real users
- Break up design debates quickly so that your team can spend more time implementing a polished product
- Interpret feedback and complaints correctly, so that you can swiftly address real issues and put aside the red herrings
- Ensure your team is equipped with the right expertise and process to carry out the project

It's easy to take the conservative approach and maintain things the way they are, ensuring you don't hurt customer goodwill with a bad UI decision. The reality, however, is that with proper product management and UX techniques, a UI update can *reliably* produce results that keep your product ahead of the curve.



Contact Us

For questions or comments about this white paper, or for more information on a consultation, please contact:

Didier Thizy
Director, Marketing
didier@macadamian.com
+ 1 877-779-6336 x136

About Macadamian

Macadamian is a global UI design and software innovation studio that provides a complete range of highest quality usability, design, and software engineering services to industry leaders across North America. Our experience, and proven ability to work seamlessly with product management executives and software teams is why companies turn to Macadamian to develop product strategies, design compelling user experiences, and build quality software.

Whether you're a small start-up or a corporate giant, we can help you transform ideas into market-ready features or products that will stand out from your competition and delight customers.

Additional information can be found at www.macadamian.com.

