

Introduction

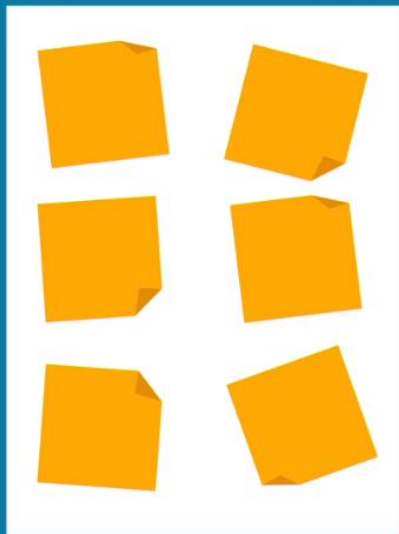
Welcome



A circular icon with a blue background and a yellow circle in the center. Inside the yellow circle is a white square containing a diagram of four wireframe screens connected by arrows, representing a user flow or storyboard.

## Scenarios and Storyboards

- Describe user and product behavior in detail
- Provide an understanding of design viability
- Cover end-to-end detail on interaction
- Serve as a great communication tool



LinkedIn



LinkedIn

Hello, I'm Chris Nodder, welcome to the fifth installment of the [UX Design Techniques](#) series. In this episode, we'll look at the way scenarios and storyboards help you ensure that there are no gaps in your design before you start building it. This is the fifth course in a series that describes a set of techniques you can use to make your development process more user-centered. In this course, I'll show you how to turn the ideas you generated in your ideation exercises into a more complete solution that you can subsequently prototype and build. After running design charrettes, or other similar ideation exercises, you have some great ideas for how you could design the interaction of your product to remove the pain points that users normally experience. However, at this point, you don't know how well those design ideas match with users' real-world tasks. To help flesh those designs out into full solutions, it helps to create scenarios that detail how users would interact with the feature. Then, turning those scenarios into storyboards means you can list all the areas that need to be built in order to make the solution useful and usable to your customers. It's time to take a look at the scenario and storyboard creation process. A well-

[thought-through storyboard helps you clearly articulate what you need to build to make users happy. So let's get started.](#)

---

## 1. Scenario and Storyboards in the User-Centered (UCD) Process

What are the uses of scenarios and storyboards?



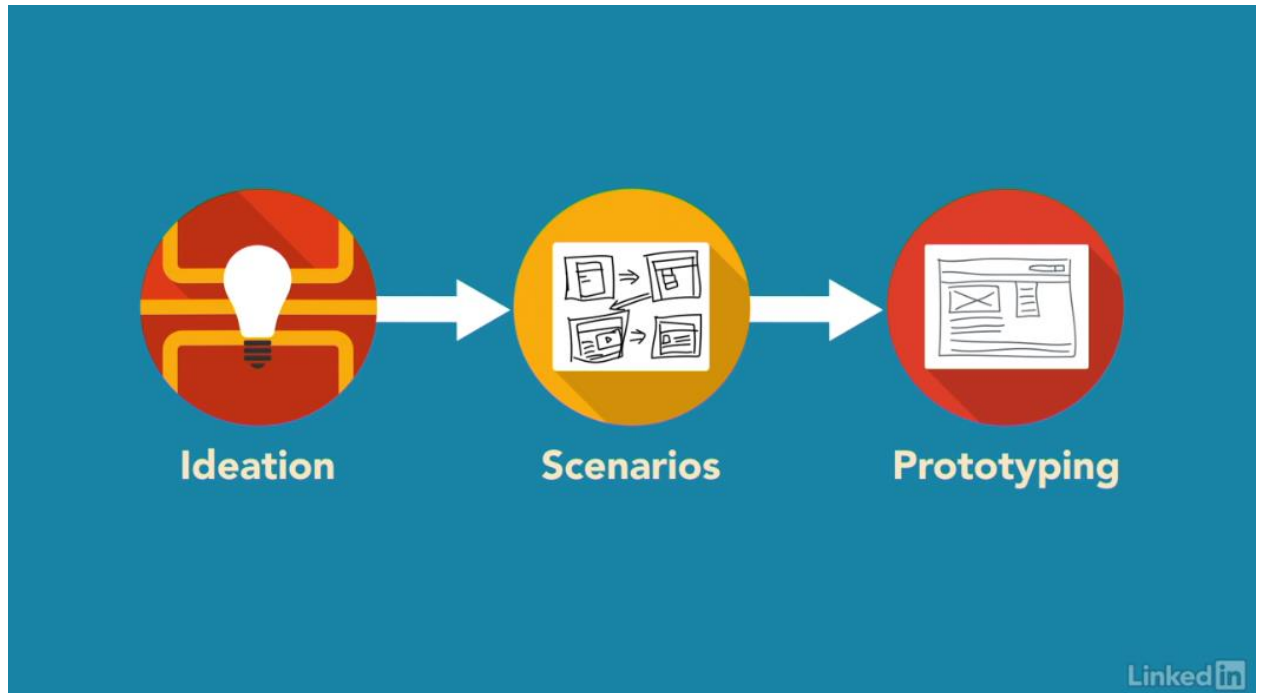
# Scenarios and Storyboards

- Describe user and product behavior in detail
- Provide an understanding of design viability
- Cover end-to-end detail on interaction
- Serve as a great communication tool

LinkedIn

[Scenarios and story boards provide a reality check for your designs. Helping you to consider how the interaction will play out in a real environment. Once you've used ideation techniques to create some concepts for how you might want your new interface to behave, it's time to create a story that incorporates that behavior, there are a couple of reasons to do this, one is that it lets you describe the behavior in more detail to other people. The other reason is that by walking through a user's typical interactions with the interface you'll end up with a deeper understanding of how well your design idea would work in reality if you rush straight from ideation into building a solution, you're likely to miss out on some important nuances that always shows in an interface when parts of it look like they were tacked on later. The truth is, they probably were tacked on, because the developers didn't take the time to consider their user's end to end interaction with the feature, that means they had to go back and add things or change the interface in an attempt to make it feel better. Instead, scenarios and storyboards ensure that you can tell a good story about how your proposed idea helps your personas to complete their task, and how it removes their pain points and meets your goals. Scenarios and story boards help you get closer to a true design solution, and they provide a great communication tool for keeping the whole team on track as you start development work.](#)

## Where these techniques fit in the UCD process



Be mindful of issues that stop the flow of these scenarios



# Involve the Whole Team

- ▶ User experience
- ▶ Business
- ▶ Marketing
- ▶ Development
- ▶ Project management

LinkedIn

Scenario and storyboard creation is most useful right after the ideation phase. To briefly recap, first you observe users, and then you create an experience map to extract pain points, goals, and personas. This gives you the information you need to conduct ideation exercises, such as design charrettes. Design charrettes give you a set of potential design concepts to work from. But to insure that the design charrette ideas will work in reality, you'll first create written scenarios, and then draw these out using storyboard techniques. Creating scenarios and storyboards gives us a head start on building a prototype that truly meets users' needs. Of course, the scenarios are written from the perspective of the personas you've created, so that you describe the task and your design solution in a way that would satisfy your personas' needs. The information contained in the storyboards maps almost directly to the elements that you'll need to create in order to build a suitable prototype. You might be wondering why I suggest taking this extra step between ideation and prototyping. What I've found is the creativity you use during the ideation phase needs to be tempered with reality before turning it into an interface. The process of describing how users would work with your designs allows you to explore many elements of that interaction, be mindful of issues, and develop coping strategies before you've started building the interface itself. That way, your final interface will flow smoothly through the interaction, rather than feeling like it wasn't designed for the users' full tasks. Scenario creation is the time when all the different disciplines' goals come together. There has to be a good user experience, the potential for a return on investment, a clear marketing value proposition, and the ability to deliver in a suitable time scale. The different team members involved in creating scenarios will bring the motivation from their various backgrounds to the session, and so we'll end up incorporating those elements into the scenarios you build. That means your scenarios will be believable, and everyone on the team can get behind them. Having this level of agreement between team members so early in the process means there will be less chance of hold-ups in subsequent stages of the development cycle.

## Benefits of Scenarios and Storyboards

- Ensuring a design is workable
- Describing required interactions
- Uncovering error conditions
- Predicting user behavior
- Preventing rework later on

LinkedIn

Storyboards keep the focus on meeting the user's needs and project goals.

LinkedIn

These techniques ensure that the design ideas you have are workable, show you what types of interactions will be required, help you to explore what error conditions might exist, and help to predict how users might interact with the interface. This all happens before you do any actual interface design. As with all the other user-centered design techniques that we've discussed, the idea behind scenarios and storyboards is to prevent you from having to do rework later in the process. Gathering user requirements early prevents rework you'd have to do when you finally learned about user's true behavior. Ideation prevents rework by identifying many potential design solutions before you get too engrossed in a single way forward. By stepping through how users might react to those design ideas. Scenarios and storyboards prevent rework by ensuring that there's a clear story to tell for your chosen design ideas. Storyboards are also an incredibly visual depiction of the solution you intend to produce. More than almost any other artifacts you create during the user-centered design process. Storyboards describe a combination of user



needs, design concepts, input conditions, and expected outputs. They do this in a compact and believable way that's immediately accessible to team members, management, and product sponsors. Anyone who's ever read a comic strip in the newspaper will be able to relate to storyboards. Although they depict a great deal of information about the interaction, storyboards do not dive into the interface itself. That prevents people from getting too carried away by specifics, instead it helps to keep the focus on the most important level at this point and that is ensuring that user's needs and the project's goals will be met by the proposed solution.

---

## 2. Creating Scenarios

### Prerequisites for scenario creation



# User data is important for writing believable stories and interactions.



[Before you can create scenarios, you need to know who you're building for, what those people's issues are, and what potential solutions you could employ. Luckily, you should have all this information readily available. If you've been following the user-centered design process we've described in previous courses in this series, you'll already have created personas that describe the specific user types that you're building the product for. You'll have extracted pain points from your experience map that correspond to users' biggest issues, and you'll have output from an ideation exercise, such as a design charrette, that describe the type of solutions you want to create. Although it's possible to create scenarios and storyboards without these previous steps, it will be harder for you to write believable stories, or to justify your interaction design decisions. The scenario process will also take longer because you won't have as many pieces of inspiration that your personas, pain points, and ideation exercise will provide. I truly believe that if you're watching this course without having first performed the other activities I've listed, you'll be better off starting at the beginning. Your scenarios will be more realistic and your storyboards will sell your design ideas much better that way. Plus, the process will give you more inspiration and ensure that you develop a much more user-centric solution.](#)

## Writing scenarios

[Writing scenarios is a fun activity. You get to put your personas into situations were they'd need to use your product, and then describe how the user experience that you want to create, will help them through that situation to a suitable outcome. The idea isn't to make life difficult for your personas, but just to describe how a regular day would be made better with the design ideas you've created. You might want to include a couple of the issues that you know will crop up regularly. So that you can describe how your design works around those issues. But the idea isn't to create an obstacle course for your persona to clear. The finished scenario should describe the different elements of the user experience that you aim to build, and demonstrate how that experience meets the goals you created. To do this work in pairs with another teammate. You'll need to have access to the experience map you created previously. See the course in this series on Analyzing User Data if you haven't done this already. Or whatever sources of user data you have available. You can have several pairs of people working on the same scenario at the same time. Because scenario writing is still a creative activity, getting multiple different takes on the same situation can help you come up with multiple creative ways of resolving the same issues. Choose one or](#)



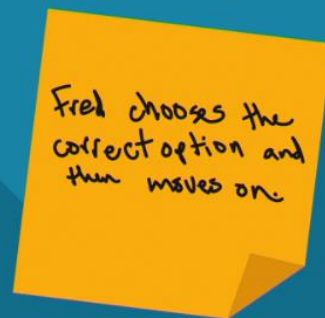
more of the user activity areas from your experience map to write a scenario around. Although the experience map describes the current situation, rather than your ideal situation, you'll still need to either accommodate or replace each of the user's current activities in order for your new design to be successful. Grab a stack of sticky notes and start writing out each step the user must take to reach their desired outcome. One per sticky note. I found this works best by just telling each other story fragments of what the persona might do to reach their goal. And then writing each step on a sticky note. As you go along, you'll probably find the need to embellish earlier steps with additional information. Or to add extra steps, that's where the sticky notes come in handy. It's easy to re-arrange or add notes, to add steps you forgot. You'll end up with a sequence of sticky notes, each describing a step in the interaction. Sometimes, for easier tasks this might only be five or six sticky notes. Other times, for longer and more complex interactions, you might end up with a couple of sheets of paper each containing multiple sticky notes. Now if you had several pairs of people working on the same scenario, it's time to compare the stories you each created. Go around the room taking it in turns to read out your scenario. And let other team members ask questions about how the persona might react in certain situations. After you've been around the room, take the time to combine the best elements from each group's scenario into one overall scenario. This combined scenario will become the point of reference for creating a storyboard and subsequently your actual interface. So take the time to make sure it covers all the points you care about. You'll need to repeat this scenario creation process for each part of the interaction for which you're designing a solution. You'll know you're done when your scenarios cover all of the user tasks you need to support and meet all of the goals you set.

#### **Tips for good scenarios**

Work in pairs

# Focus on interactions

LinkedIn



LinkedIn




Here are a couple of tips to help make your scenario creation exercises go smoothly. The first is to reiterate the need to do this exercise in pairs. Because everyone must work with another person, they have to say out loud what the user's actions should be. Having to verbally describe the stories to someone else ensures that you write it in a way that other people will understand. When you come back to this work in a week or a month, it still needs to make sense. By writing it coherently first time, it's harder for other people to misinterpret what you mean later on. My next tip is don't describe interfaces, instead describe interactions. By this, I mean that when your scenario talks about your persona's interaction with the product, describe what they put in and what they'll get out, rather than the actual method they use to achieve their goal, so rather than saying Fred uses a dropdown menu selector in a modal window to choose the correct option and then taps the Done button, just say Fred chooses the correct option and then moves on. It's easy to build interfaces later, so don't get tied down into a specific design approach yet. The interface should be determined by the interaction, rather than the interaction being determined by the interface. Finally, if you really want to make sure that you've created a good scenario, try acting it out. You can even have someone on the team play the role of the computer if that helps. When you act it out, does your scenario feel realistic? Does it seem achievable? Good scenarios should progress naturally and the act of walking through it verbally helps you to find any disconnects.

---


### 3. Storyboards to Visualize Scenarios

#### From scenario to storyboard



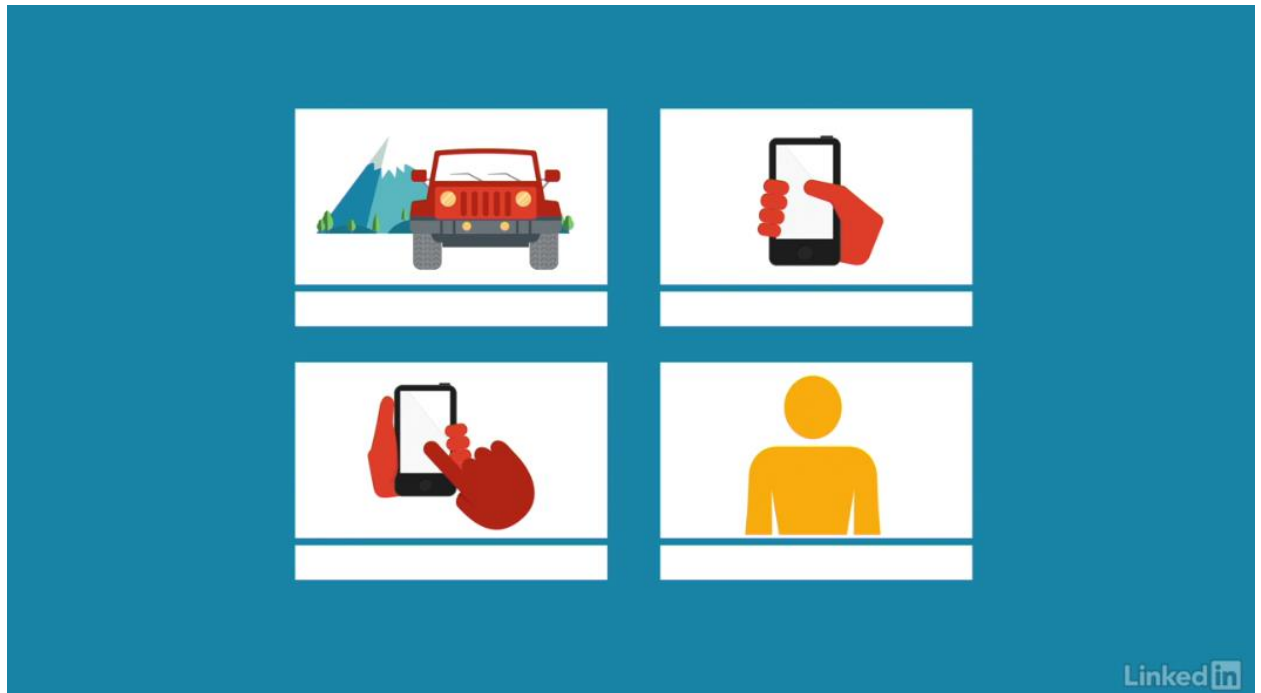
# Storyboards

- ▶ Depict emotion and action
- ▶ Draw attention to specific elements
- ▶ Show the user's progression through the interface
- ▶ Demonstrate workflow quickly and visually

Linked 

I often find it helpful to turn scenarios into visual storyboards. Storyboards were first used to help plan out cartoon animations, then later, to do the same thing for film. Storyboards help movie directors and producers to set out the scope of the interaction and make sure that the story hangs together in a believable way. That is all just as relevant in user interface design. Storyboard allows us to bring elements from the film world to UI design, the visual element allows for depiction of emotion and of action, both things that are hard to show in words alone. Visuals also allow the storyboard to pull our attention to a specific element of the UI, or interaction by drawing a zoomed in image, or to make us think about the bigger picture, by drawing something from a wider perspective. It's not essential, but many people react better to seeing things in a visual medium, than with words alone. Although storyboards still don't show the actual user interface, they show the user's progression through the interaction. This can help to firm up the design, remove unnecessary steps, and demonstrate where additional work is required. Storyboards are also really useful for demoing to executives who can't, or at least don't take the time to read. Best of all, most of the information you need to create your storyboards already exists in your scenarios. Although it's not a one on one mapping from the scenario to the storyboard, the same elements will need to be shown in both. If you first created a good scenario, it will be much easier to move on to building your storyboard.

## Storyboard creation



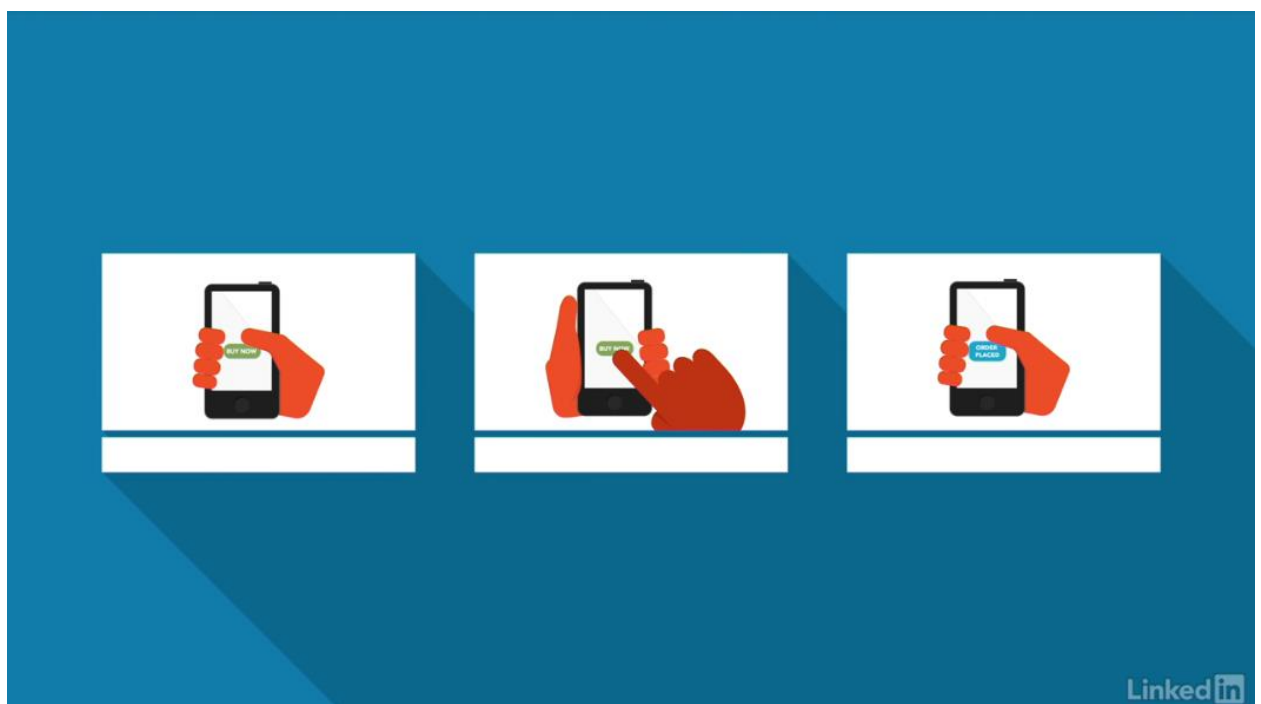
- Don't glue down your panels until your order is finalized
- Review the storyboard with your team



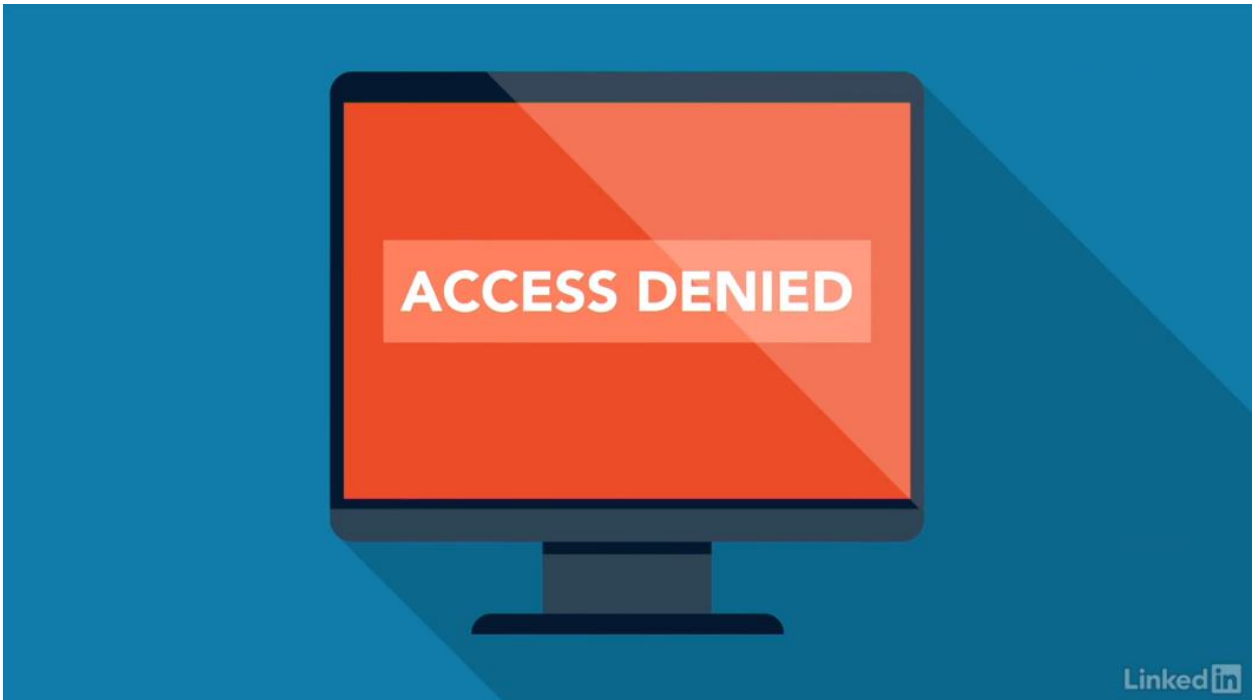
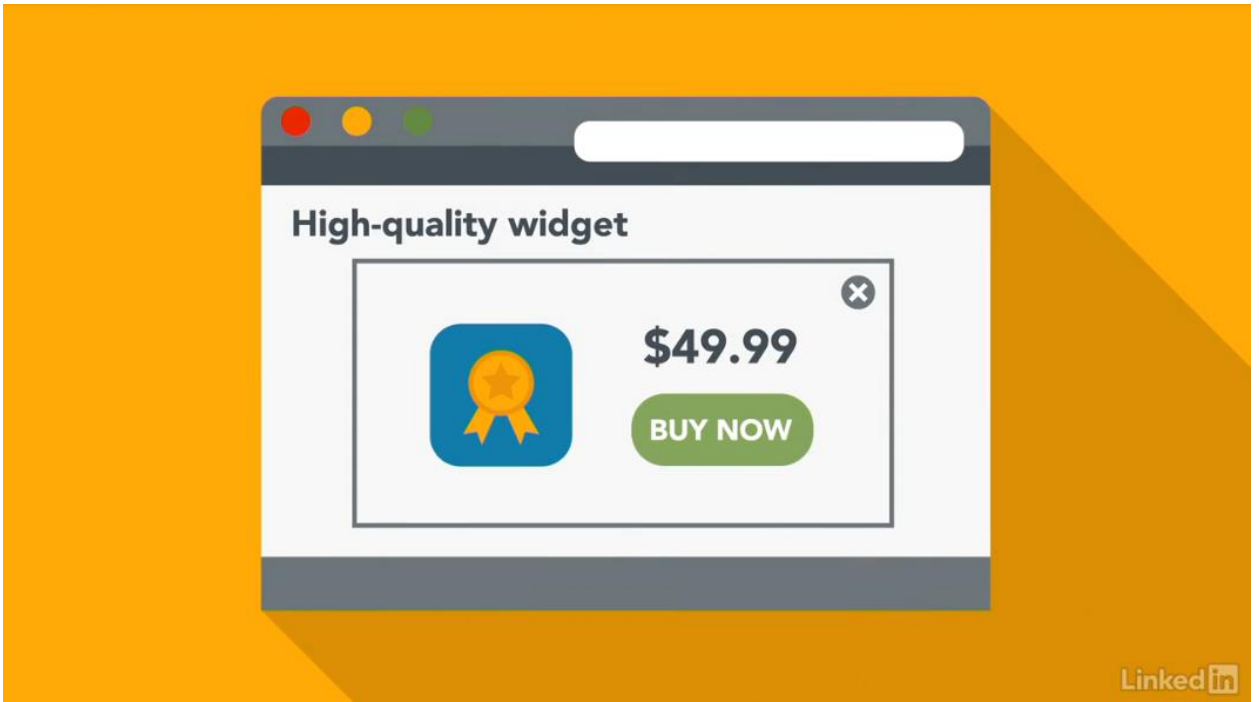
Storyboarding is one activity that's normally undertaken by just one or two individuals using the scenario input that was created by the whole team. The idea is to take the scenario text and create a visualization of each scene, showing how the users interact with the software and with their surroundings. You'll probably find an almost one-to-one relationship between the sticky notes from your scenario and the panels in your storyboard. But that isn't always the case, because often, a storyboard visual can cover several scenario steps. I find that it helps to establish a couple of shots that you'll repeat throughout the storyboard. For instance, the general scene, a closeup of the device, a depiction of the user interaction, an image that demonstrates the user's reaction to the interaction. Now, just use the most suitable of these shots for each scenario sticky and fill in the details that differentiate this panel of the storyboard from all the others. Like we already mentioned, storyboarding originated in the animation and movie industry. Directors would use storyboarding to test out different camera angles in order to help tell their

[story](#). You can use similar film-like camera angles as you draw your storyboards. For instance, you can establish the scene with a wide-angle view or use a close-up of the user's face to show an emotion. Or, you could draw a closeup of part of the interface to show a specific interaction. Sketch each panel on its own piece of paper. I find that three inch by five inch index cards are an ideal size, and then place them in order on a backing sheet of paper with room below each for a written description. The written description may be the same wording as the scenario, but it may be possible to edit that wording down now that there's an image that describes most of the action. Remember that your storyboard may still change as you progress through the sticky notes from the scenario, so don't glue any of the panels down until you're sure about the order. When you're done, review the storyboard with the team. Take their comments into consideration. Everyone on the immediate team needs to understand the storyboard in order for it to be a suitable artifact as you move through the process and to use it to create the actual interface.

### Tips for good storyboards







# You aren't designing the user interface.



Here are some tips for creating storyboards, to make them truly useful as you continue through the user-centered design process. The best advice I can give you is to display only the images that move the story forward. By that, I mean create panels for your storyboard if there's a change of scene, or if you need to show a specific piece of information, but don't worry about showing the interim states. You want to move the story forward, rather than getting bogged down in minute details. The only time you really need to show the interim states is if you think that there's an essential element of the interaction, or a system change, that you need to denote to help people who will be using the storyboards later to translate them into actual interfaces. If that isn't the case, or if the interaction will follow a very standard process, don't bother showing those interim steps. You may have noticed how the user interfaces in movies tend to be oversimplified, they often just show a couple of text entry boxes on the screen where in reality, there will be multiple form fields displayed. Or they display big red flashing access denied text when the hacker tries to get into the system, whereas in reality, the response would be muted and would not obscure the rest of the screen. Now, although you'd never build it that way, remember that you aren't designing user interface in these storyboards, instead, you're trying to express the tone of the interaction, so it's fine to just highlight certain parts of the interaction that are essential for telling the story, just like in those cheesy movies. Here, I'm showing the only parts of the product information page that are essential for the story, even though in reality, there might be other items such as a product description, specifications, customer reviews, and so on. This is one part of the user-centered design process that sometimes lends itself to being done on a computer. Because you'll be reusing template images, it might make sense to use a presentation tool like Powerpoint or Keynote to build the story panels. In fact, there are already public domain templates to help you do this. For instance, the ones created by Martin Hardy at [designcomics.org](http://designcomics.org), but there's no requirement to go digital, and I still like the very immediate and immersive feel of a hand-drawn storyboard. Ultimately, it's up to you and your comfort level with creating the storyboard images.

## Conclusion

Early feedback is much cheaper  
than writing the wrong code.