

Understand information architecture

Create paper wireframes

Create digital wireframes

Use Gestalt Principles in wireframes

▶ **Video:** Use Gestalt Principles when creating wireframes
4 min

📖 **Reading:** Learn more about using Gestalt Principles in wireframes
20 min

📝 **Practice Quiz:** Test your knowledge on Gestalt Principles
4 questions

Week 2 review

Learn more about using Gestalt Principles in wireframes

There is a belief among some people that, like a piece of art, the design of an app or website is subjective and open to a user's personal interpretations, thoughts, or feelings. Although a user's reaction to a design might be subjective, the design itself isn't. Your designs will give users solid solutions to problems they experience. Think about what the user wants to do when they visit your app or website. What is their goal? How will you meet that need with your design? How will the user react?

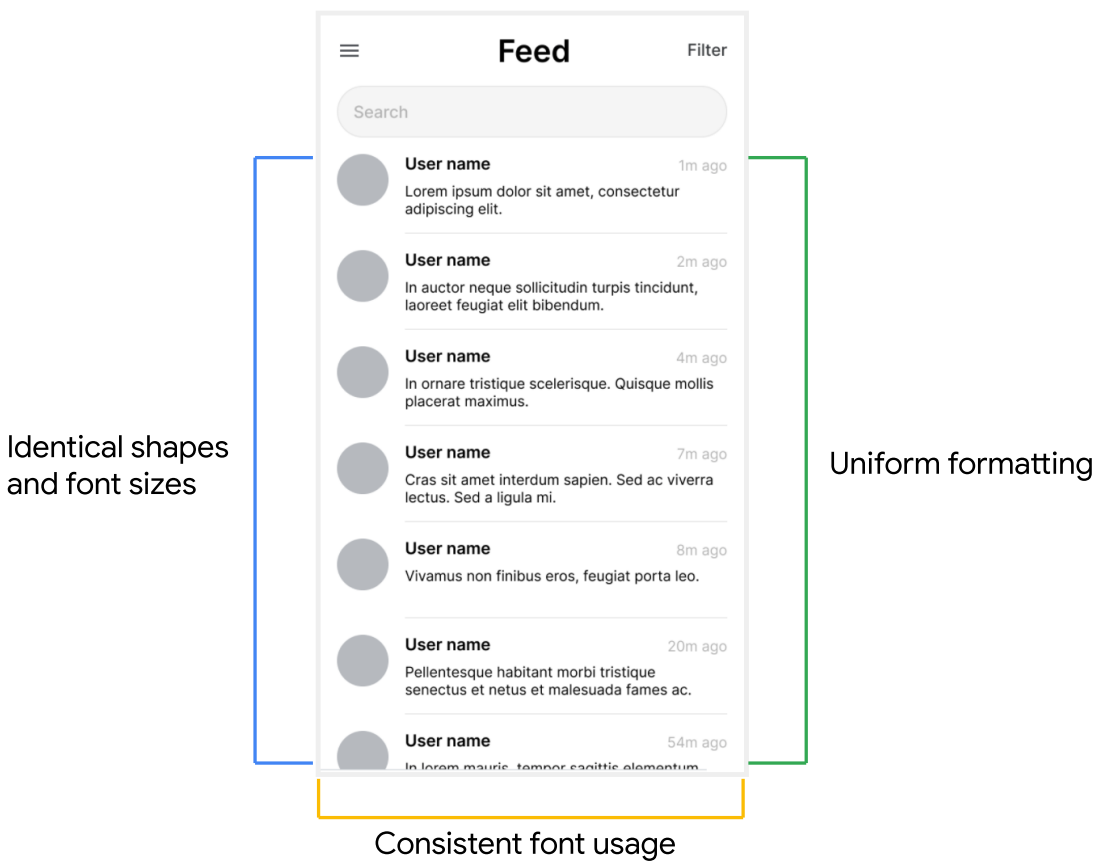
The human brain makes sense of the world around it by building logic, patterns, and structure. **Gestalt Principles** describe how humans group similar elements, recognize patterns, and simplify complex images when we perceive objects. These principles guide how humans view the designs around them in everyday life. In the wireframes you're creating, you can apply Gestalt Principles to organize content so it is visually pleasing and easier to understand.

In the video, you learned about three of the Gestalt Principles: similarity, proximity, and common region. As a reminder:

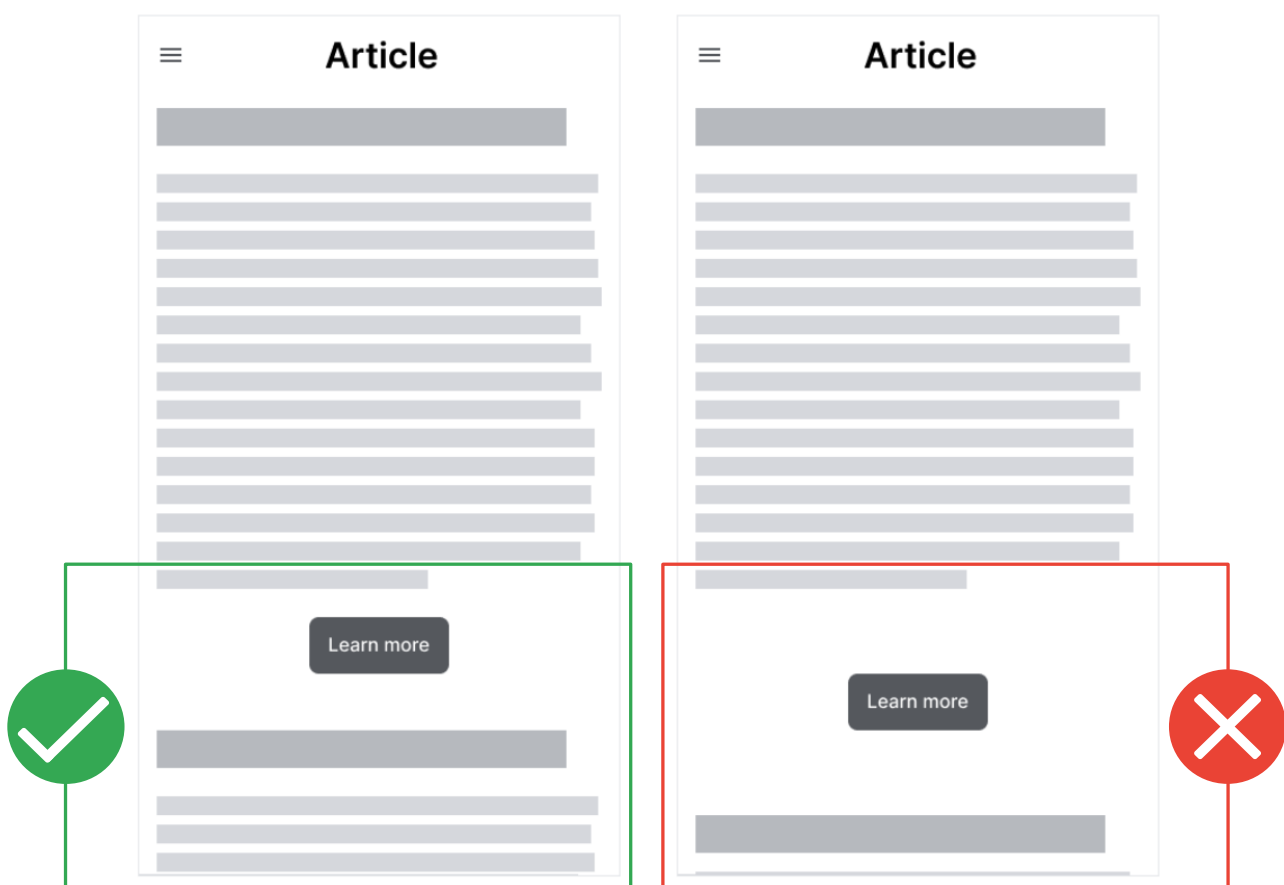
- **Similarity** means that elements that look alike (in shape, size, or color, for instance) are perceived to have the same function.
- **Proximity** means that elements that are close together appear to be more related than things that are spaced farther apart.
- **Common region** means that elements located within the same closed area are perceived to be grouped together.

So, how do you apply these Gestalt Principles to your own work? Check out an example of each of these three Gestalt Principles in actual wireframes.

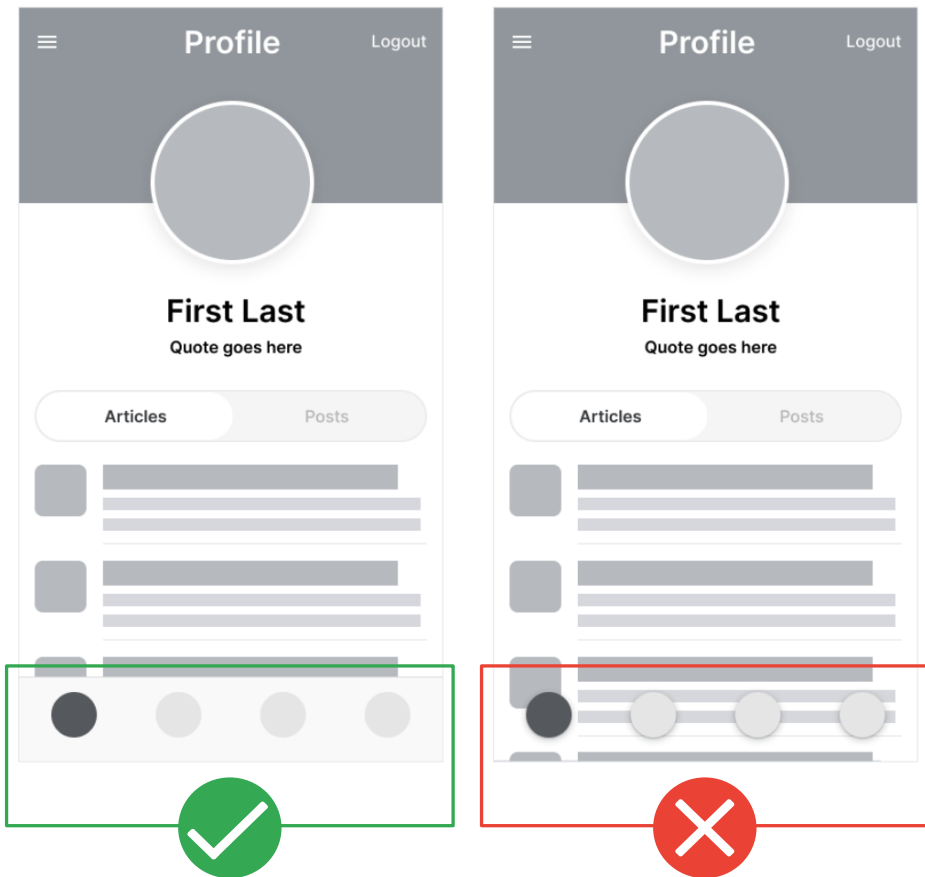
First, **similarity** is key when it comes to your wireframes. To help users navigate your app quickly, you want to keep "like with like" for consistency. For example, if you're including avatars or images of users in the design of your app, you'll want to make sure that the placeholders for those avatars are the same shape and size in your wireframes. This similarly indicates to users that the elements have the same function — in this case, linking to profiles of users.



Next, imagine you draw a button labeled "learn more" close to the text of an article. If a user presses the button, they'll assume they can learn more about the content of that article. This kind of design uses the Gestalt Principle of **proximity** to your advantage! Alternatively, if you leave a lot of space between the text of the article and the "learn more" button, users might not understand the connection between the two elements. A few seconds of confusion or hesitation can mean the difference between a user continuing to the next page of your app or closing it to find something more useful.



Finally, have you ever noticed that navigation bars at the top or bottom of a mobile phone screen are usually in a different color or grouped with a border? That's the Gestalt Principle of **common region** in the real world. By creating a border and adding a block of color to a navigation bar, you can anchor the icons for the main pages of your app, instead of having the icons float on the screen. This design tip helps users recognize that the icons in the navigation bar are more important than others and are perceived to be grouped together.



There are countless ways to apply Gestalt Principles in your wireframes that improve the usability of your app. If you want to learn even more about Gestalt Principles, check out this [article from User Testing](#) that describes additional Gestalt Principles and this [article from UX Collective](#) with examples of the Gestalt Principles in wireframes.

Mark as completed

