1.10: User Stories

#### **Introduction**

Hey there! Glad to have you back! You’re nearly done with the first Achievement. Time flies when you’re having fun, right?

In the previous Exercise, we stepped back from the world of UX to look at things from a high-level business perspective. We talked about BRDs (business requirements documents), as well as how they’re used to ensure all business stakeholders are aligned. Now, we’re going to take the information from our business requirements doc and translate it into user stories and job stories—small sentences that describe individual actions a user can take in your app.

You might be wondering why it’s important to convert business requirements into stories. After all, the requirements are already spelled out line by line in the BRD, and it is your project—you know what it takes, right? Even the best UX designers, however, can overlook things in their own projects. Writing stories can shed light on aspects of the business requirements you might not have noticed before. They can also help you simplify your processes while adding a human (user) voice to the specifications needed for the development of your project.

Once you’ve established the basic business requirements, you can start using semantic frameworks to bring them to life. These frameworks are a great way to communicate requirements to your team. Two such frameworks are **user stories** and **job stories**. While they both serve slightly different purposes, they can be used in conjunction with each other to determine what a persona wants to accomplish (the user story) and the context in which they want to complete this action (the job story).

Ready to tell your story?

#### **User Stories**



**User stories** are exactly what they sound like—stories that explain a specific action a certain type of user can take within your application. User stories are part of the Agile methodology (remember Exercise 1.8?). The goal of these stories is to distill your requirements down to small, readable and—most importantly—executable parts. Once you’ve exhausted your list of user stories, they’re individually assigned to team members, creating manageable chunks of work that all add up to a finished product.

Whether you subscribe to Agile or Lean methodologies, user stories are a great tool for simplifying product requirements.

There are a number of variations when it comes to writing user stories. Here, we’ll look at one of the most basic formats:

As a <role>, I want <a feature or action>, so that <desired outcome>.

A **feature** is a distinctive attribute or element of a website or app that helps a user achieve their goal. Features can be anything from autosaving when writing an email to allowing users to leave a review on a product they've purchased.

These user stories will ultimately be translated into visual user flows, which are diagrams that show how a user makes their way through a feature of your software. We’ll dig deeper into user flows in Exercise 2.6.

Referring back to the BRD for our travel guide app in Exercise 1.9, here are some examples of user stories we might derive from our travel guide app:

* As a <user>, I want <to download local guides for offline use>, so that <I can view them on the go without relying on Internet or cell phone data>.
* As a <guide creator>, I want <to see analytics on who’s viewed and downloaded my guides>, so that <I can better understand my users>.
* As an <administrator>, I want <to be able to delete accounts>, so that <I can keep the database organized>.

User stories usually come into play after you’ve conducted your user research, performed your competitive analyses, and defined your business requirements, which is why we’ll be looking at them again in Achievement 2. By then, you might find yourself wanting to revise quite a few of them. Don’t worry! Your user stories don’t have to be (and shouldn’t be) perfect the first time around. As you make your way through the product design process, whether in this course or in your actual career, you’ll uncover new learnings, requirements, and information, all of which could demand significant updates to your user stories. This is completely normal. Remember that everything revolves around the user. As you come to better understand the user and their needs, shifting things around becomes inevitable.

The main goal of user stories is to divide your work up into reasonable chunks, but they also give you a bird’s-eye view of your scope of work. Once you’ve exhausted all user stories and divvied them up amongst the team, it’s much easier to gauge how long a development cycle will take.

User stories are great for getting into the mindset of the persona. Once you've accomplished that, it's also helpful to zoom out and look at the context in which your personas complete the story. That's where job stories come in handy.



#### **Job Stories**

**Job stories** focus on the context in which a persona wants to complete a particular action. Oftentimes, it’s a shared situation that drives people to complete the same or similar action rather than shared personality traits. Job stories are most often useful when context is more important than the individual characteristics of your users and can help designers focus on answering the “Why” in regards to a particular situation or motivation.

At first glance, it might be difficult to see how this differs from user stories. Let’s take a look at the semantic structure and an example:

When <context or situation>, I want to <motivation>, so I can <expected outcome>.

Let’s see what happens when we emphasize a familiar context: Imagine someone at the airport, taking advantage of those last minutes of Wi-Fi at the airport before takeoff.

When <I’m at the airport about to take an international flight>, I want to <download travel guides to my mobile application>, so I can <use the local guides offline once I reach my destination>.

In this particular example, the job story restates the user story to focus on context and location. It brings to light a real life situation that we can all imagine: frantically conducting last-minute downloads at the airport before taking off into the data-less void that is the airplane and the foreign destination. This context can help us design for more-nuanced and specific situations.

At the time of writing this Exercise, user stories are more common and established than job stories, but it’s important to explore multiple semantic frameworks to help sharpen our thinking. Ultimately, the goal of both user and job stories is to develop a clear understanding of what the persona wants to accomplish as well as when they’re likely to use your product or service to accomplish it. Once you’ve created your user and job stories, it’s time to focus further on the problem at hand by refining problem and hypothesis statements.

#### **Writing User Stories and Job Stories**

Depending on your project and its requirements, user stories and job stories can be fairly general. Usually, however, granularity is helpful in dividing the work up into manageable bits. For example, instead of “As a guide creator, I want to upload a guide to a city so that travelers can download it,” you may be better off dividing this task into multiple user stories:

* As a <guide creator>, I want <to be able to add a header photo to my guide>, so that <it can be more personal and customized>.
* As a <guide creator>, I want <to be able to write text directly into the app one section at a time>, so that <it’s more manageable>.
* As a <guide creator>, I want <to be able to save my progress within the app as I write my guide>, so that <I can come back later to finish it>.

Each of these user stories is related to the larger task of posting a guide, so it might be helpful to keep them grouped together. This also helps manage the scope of work by breaking larger sections down into more manageable chunks.

##### **Keeping Your Stories Updated**

As mentioned earlier, it’s okay to update your user stories as you discover new insights into your users and project. User stories can easily go out of date as user needs become clearer and the scope of your project changes and evolves. You should review your user stories and job stories regularly to ensure they’re still relevant and accurate.

#### **Summary**

This Achievement was focused on the overarching concepts of UX as a field and the first steps you should take when beginning a new project, but over the course of the next few Achievements, we’ll be looking more in-depth at the many processes you’ll need to know like the back of your hand no matter the company or clients you end up designing for. User research, user testing, wireframing, prototyping—there are still many exciting things to look forward to!

In this Exercise, you learned how to personify business requirements by translating them into relatable, executable user stories. You also learned how the level of granularity in these user stories can help divide the work up into smaller, more manageable chunks. Now, you’re going to take a stab at creating user stories for your own project. Make sure you refer back to your BRD from Exercise 1.9 to help you come up with user stories. Keep in mind, however, the requirements from the doc are meant to be general, while your user stories are meant to be granular. The more specific your stories, the more manageable the work becomes.

#### **Resources**

* [10 Tips for Writing Good User Stories](http://www.romanpichler.com/blog/10-tips-writing-good-user-stories/)
* [Writing a Great User Story](https://help.rallydev.com/writing-great-user-story)
* [Replacing the User Story with the Job Story](https://jtbd.info/replacing-the-user-story-with-the-job-story-af7cdee10c27)

Take the quiz to test your knowledge on this Exercise.

Take Quiz

#### **Task**

* [DIRECTIONS](https://careerfoundry.com/en/course/become-a-ux-designer/exercise/user-stories#directions)
* [SUBMISSION HISTORY](https://careerfoundry.com/en/course/become-a-ux-designer/exercise/user-stories#step_submission_history)

 Estimated Task Time: 3 Hours.

Ready to inject some life into your app’s requirements? Review the functional requirements of your project (the ones you completed for Exercise 1.9 in the Business Requirements Document) and translate them into user stories.

**Directions**

1. Review your BRD from the previous Exercise to refresh your memory on the product you’re building.
2. For each high-level requirement listed in your BRD, write at least one user story. Keep in mind that some features require more than just one user story!
3. Export a PDF of your work and upload it here to discuss with your Tutor and Mentor. Feel free to share additional thoughts or ask questions in the submission box.

Be sure to follow the user story formula shared in the Exercise:

As a **role**, I want **a feature or action**, so that **desired outcome**.

Your user stories can be basic at this point. In Achievement 2, you'll spend more time analyzing your users, so your user identities will become more advanced.