App Design Journal

<Your App Name>

<Description of your app>

<Your Name>

Welcome

Your app design journal will help you keep track of your ideas and guide you as you cycle through the four stages of the app design process. You can play it through to see what's in it, but you'll work in slide view to add notes, images, shapes, and more. Feel free to add and duplicate slides, and refer back to it during current and future app projects.

Brainstorm

Purpose
Ideas
Audience
Focus
Reiterate

Plan

UI/UX
iOS features
Design

Prototype

Design
Flowchart
Build

Evaluate

Observation
Interview

Brainstorm

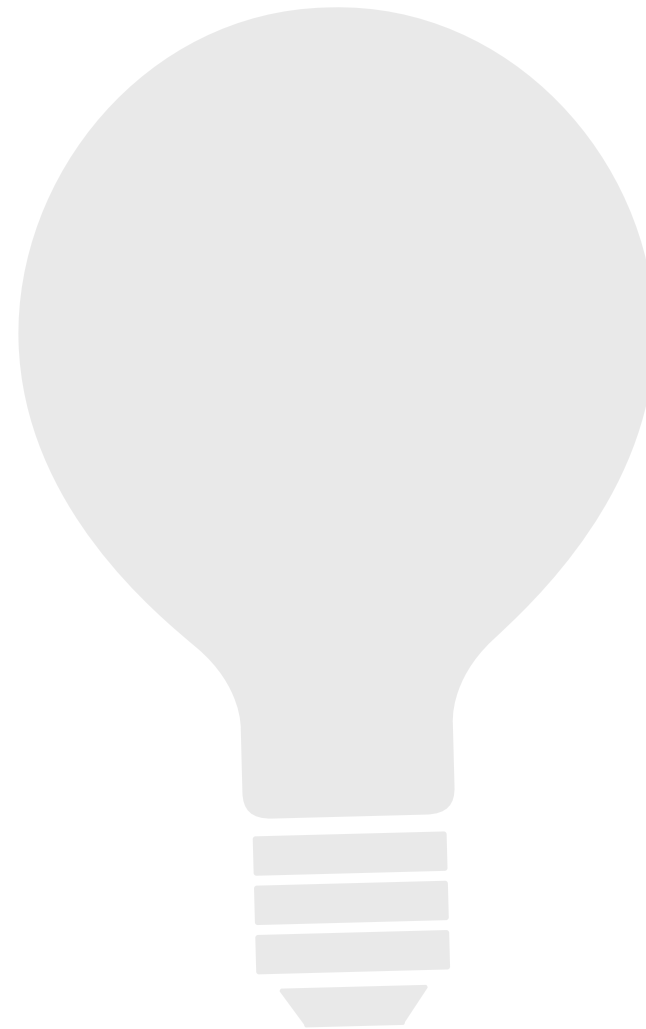**Purpose**
Ideas
Audience
Focus
Reiterate

Overview

The brainstorming stage allows you to identify problems and come up with possible solutions. This section includes a few key topics for you to think through. Some topics have optional Go Further activities if you're interested in exploring more. Jot down as many ideas, notes, and sketches that can help you design an app for solving a problem in your school or community.
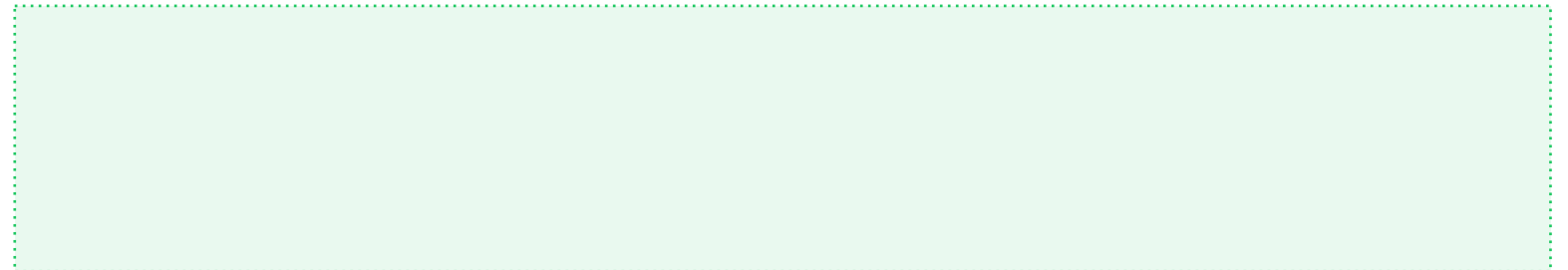
Brainstorm

Purpose
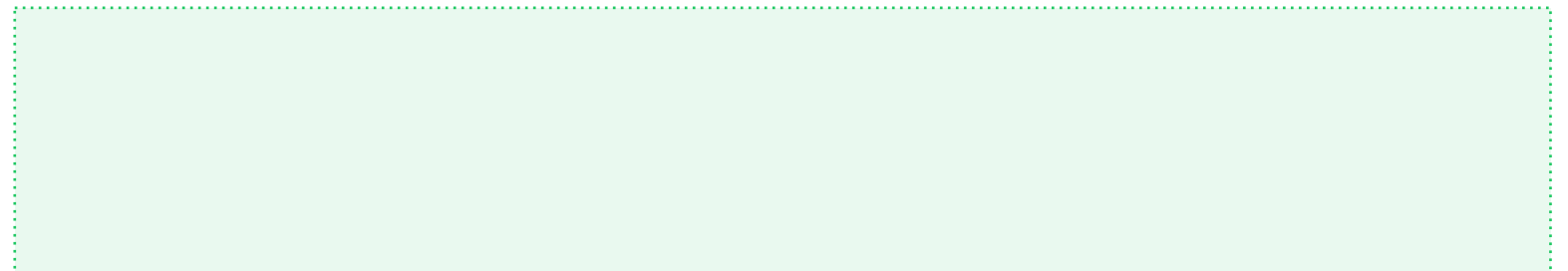Ideas
Audience
Focus
Reiterate

Define the opportunity, problem, or challenge

Before you can start to explore different options for your app, you need to be clear on what the opportunity, problem, or challenge is.
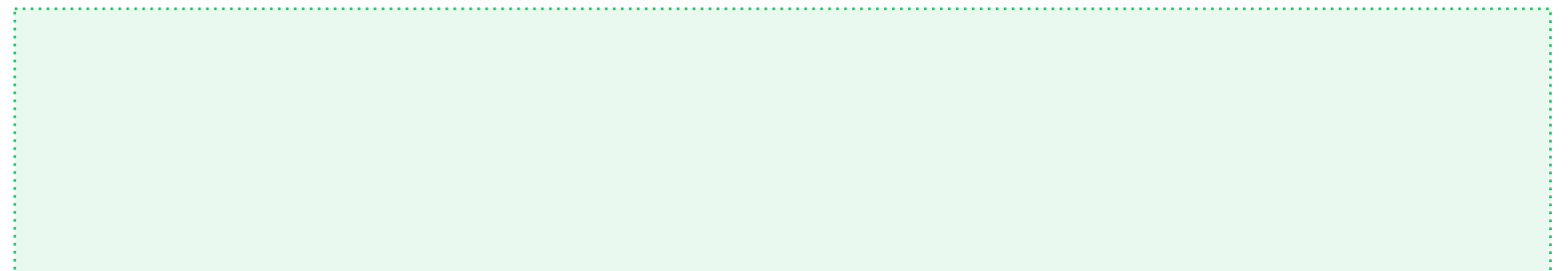
1. What do you know about the opportunity, problem, or challenge?

2. What questions do you need to find answers to?

3. Can you explain the opportunity, problem, or challenge in just one sentence?

Brainstorm

Purpose
Ideas
Audience
Focus
Reiterate

My favorite apps

Think about the apps that you use. Identify each app's purpose and why you use it. Which do you use most, and which did you stop using after just a few times? Why did you download them in the first place? Brainstorm a list of your favorite apps, and identify their purposes and the features that make them good.

App purpose:

I like this app because . . .

App purpose:

I like this app because . . .

App purpose:

I like this app because . . .

App purpose:

I like this app because . . .

App purpose:

I like this app because . . .

Brainstorm

Purpose
Ideas
Audience
Focus
Reiterate

My ideas

Brainstorm a list of apps you'd like to build. These could be new ideas, apps to solve specific problems, apps you think you can improve or personalize, or something silly! Browse the App Store for inspiration. Keep adding to this list and revisit it, as some ideas might become more or less interesting or crazy in the future.

Add your ideas.

Brainstorm

Purpose
Ideas
Audience
Focus
Reiterate

My app idea

From your brainstorming list, select one app idea to develop further and describe it below.

App name:

Write a description of what the app does.

Do a little research on your initial top ideas. Take a good look at the App Store. Do your app ideas already exist? Don't be discouraged if you find one or several that are like the app you've imagined. It just means you had a good idea. And maybe now that you've seen a bunch of similar apps, you can see ways to improve them.

For your favorite app idea, identify its top competitor on the App Store. If you can, download the app. Then check it out to answer the questions at right.

If there are any user reviews on the App Store, be sure to read those carefully, too. What are people finding difficult about the app? What else do they wish the app could do? Where are people getting confused? How would your app address each of these concerns?

My app's top competitor
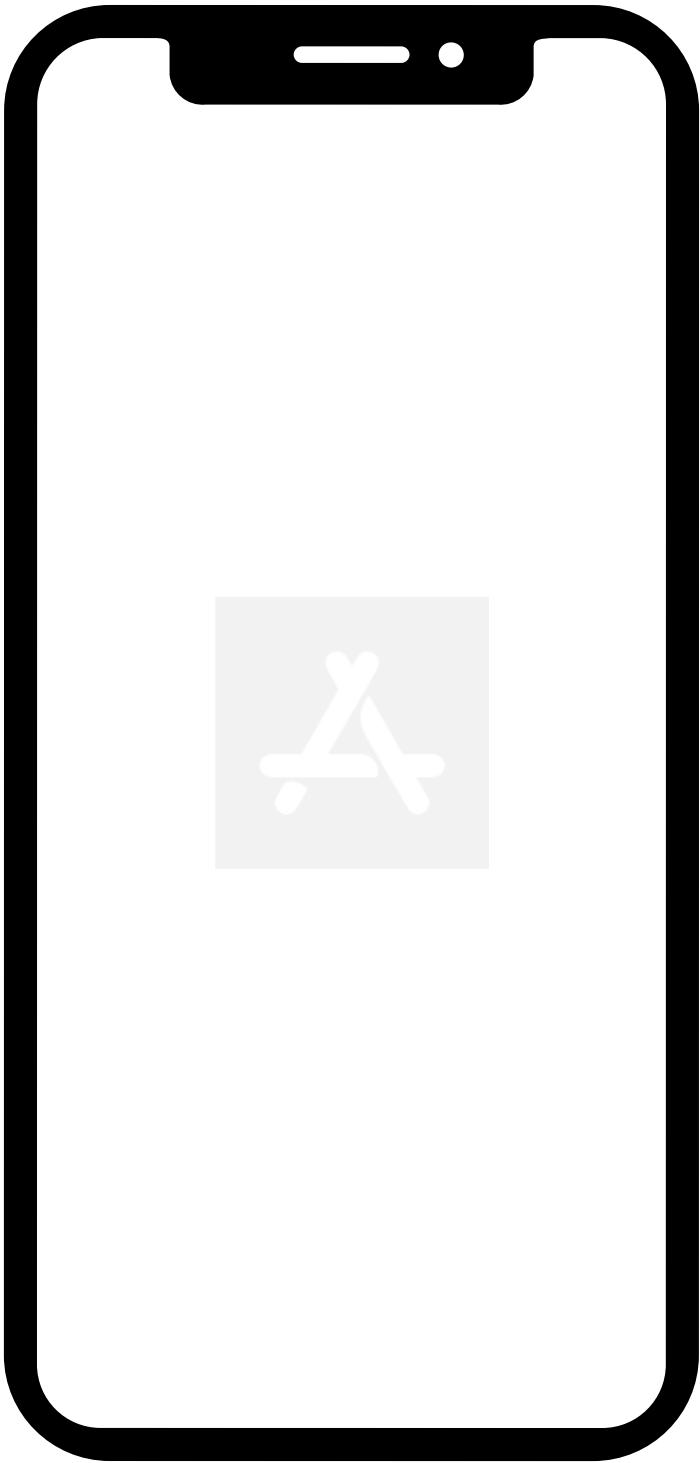
Add example images of your app's top competitor here.
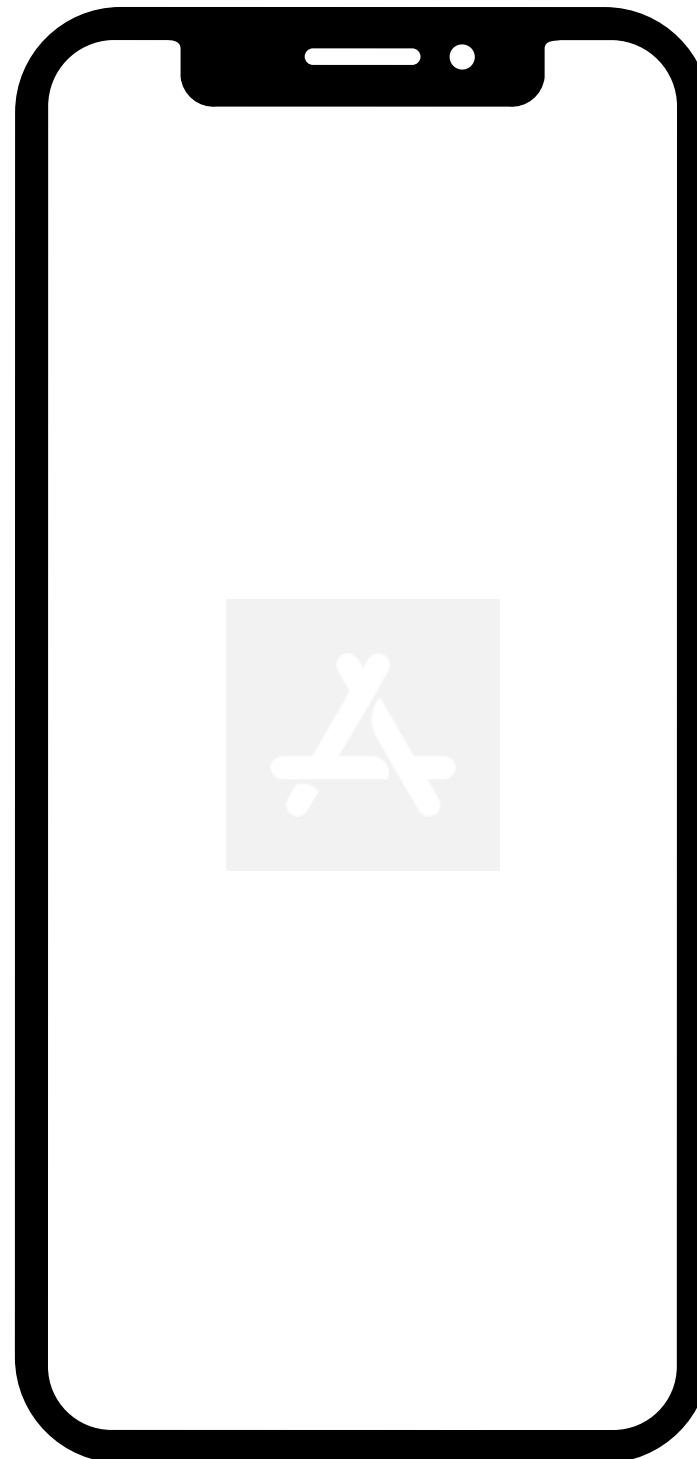
Is it easy to use? Why or why not?

Can you suggest improvements to the user interface?

How could it be designed better?

Brainstorm

Purpose
Ideas
Audience
Focus
Reiterate

Explore apps from different categories in the App Store. Take a look at the icons, screenshots, and descriptions. Add example images here to keep track of your research.

It's important to design with a target audience in mind. Who do you want using your app?

What have you learned about the audience those apps are meant for?

Did the developers do a good job communicating that?

Judging from the screenshots or preview video, do you think the apps are appropriate for their intended audiences?

Go Further
For your app idea, create a persona for each type of person who would use the app.

Duplicate this slide to outline each persona.

What does this person do?

How old is the person?

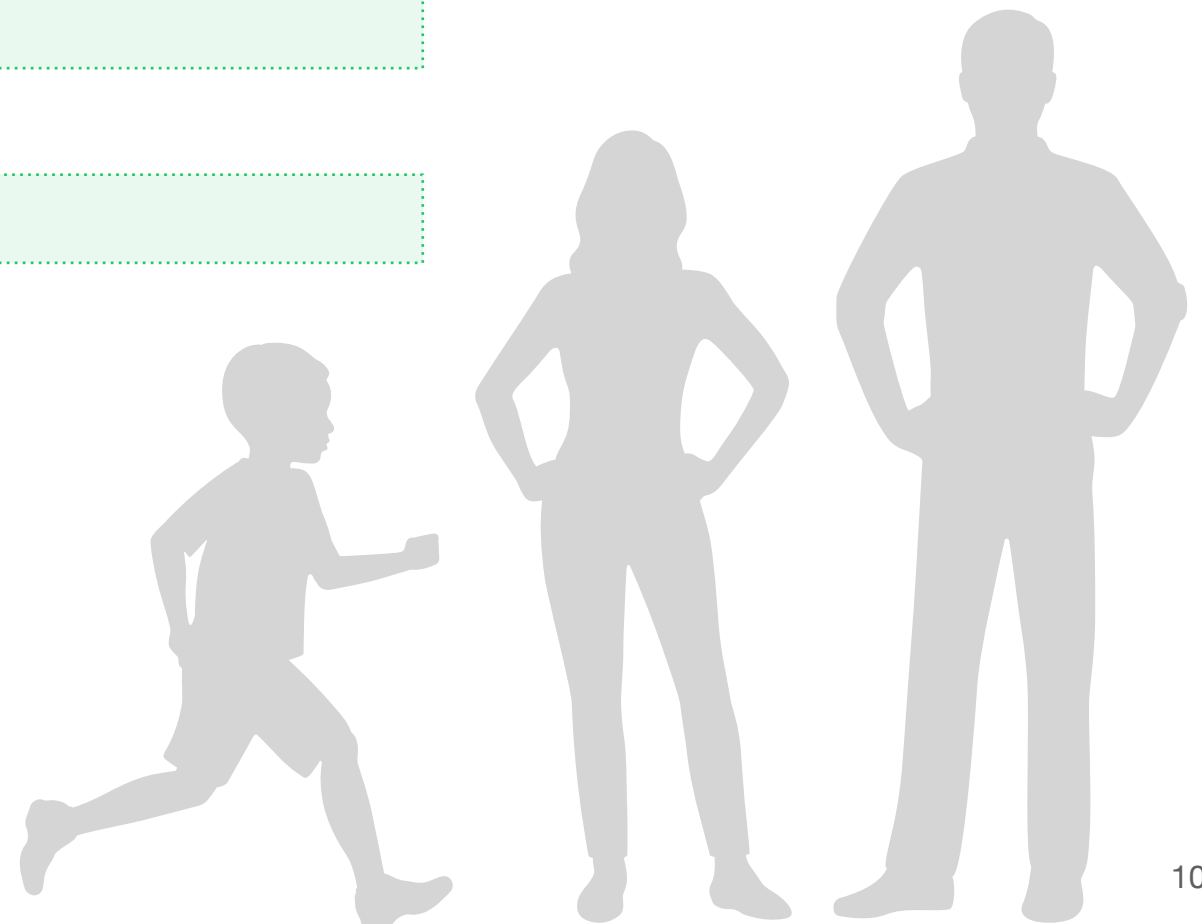Why is the person using the app?

Does the person prefer pictures or words?

How often does the person use their device?

Include other details.

Illustration or stock photo of the persona (optional)

Brainstorm

Focus

Before you commit to your app, go back and review your list of app ideas. Which ones seem most interesting? Focus on a few ideas for further brainstorming. What purposes do they serve and how do they solve issues? Who are the audiences? Write app statements to clearly define the apps' purposes. This can help you decide whether they're good ideas or not. Compare your new ideas to your favorite app idea. Is it still your favorite?

What will your app do?

> **My app will . . .**

Why does this need exist?

> **because . . .**

What will your app do?

> **My app will . . .**

Why does this need exist?

> **because . . .**

What will your app do?

> **My app will . . .**

Why does this need exist?

> **because . . .**

Plan

UI/UX
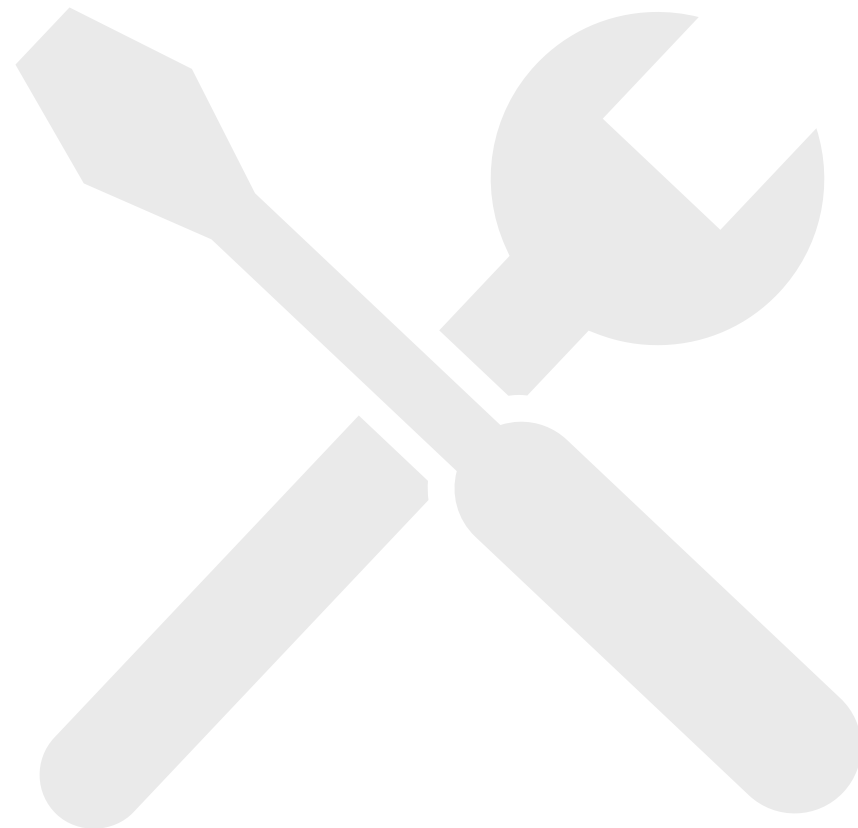**iOS features**
Design

Overview

The planning stage is when you figure out the details of your app and how it can achieve its goal.

Consider these three key areas as you develop your apps: UI/UX, iOS features, and design. The more you learn about each topic, the more advanced your app designs will be. The Apple Developer website is a good resource for you to learn more about these topics.

Plan

UI/UX
iOS features
Design

UI/UX

A good app should be easy to use. That's where the user interface (UI) design comes in. A well-designed UI makes for a good user experience (UX). Think back to the first times you used some of your apps, or try using a new app. What was the experience like? Did you get confused navigating them? Review elements like font size, icon shape and placement, and the navigation from screen to screen. Even the smallest element makes a difference in how someone experiences your app. Be sure to review the Human Interface Guidelines.

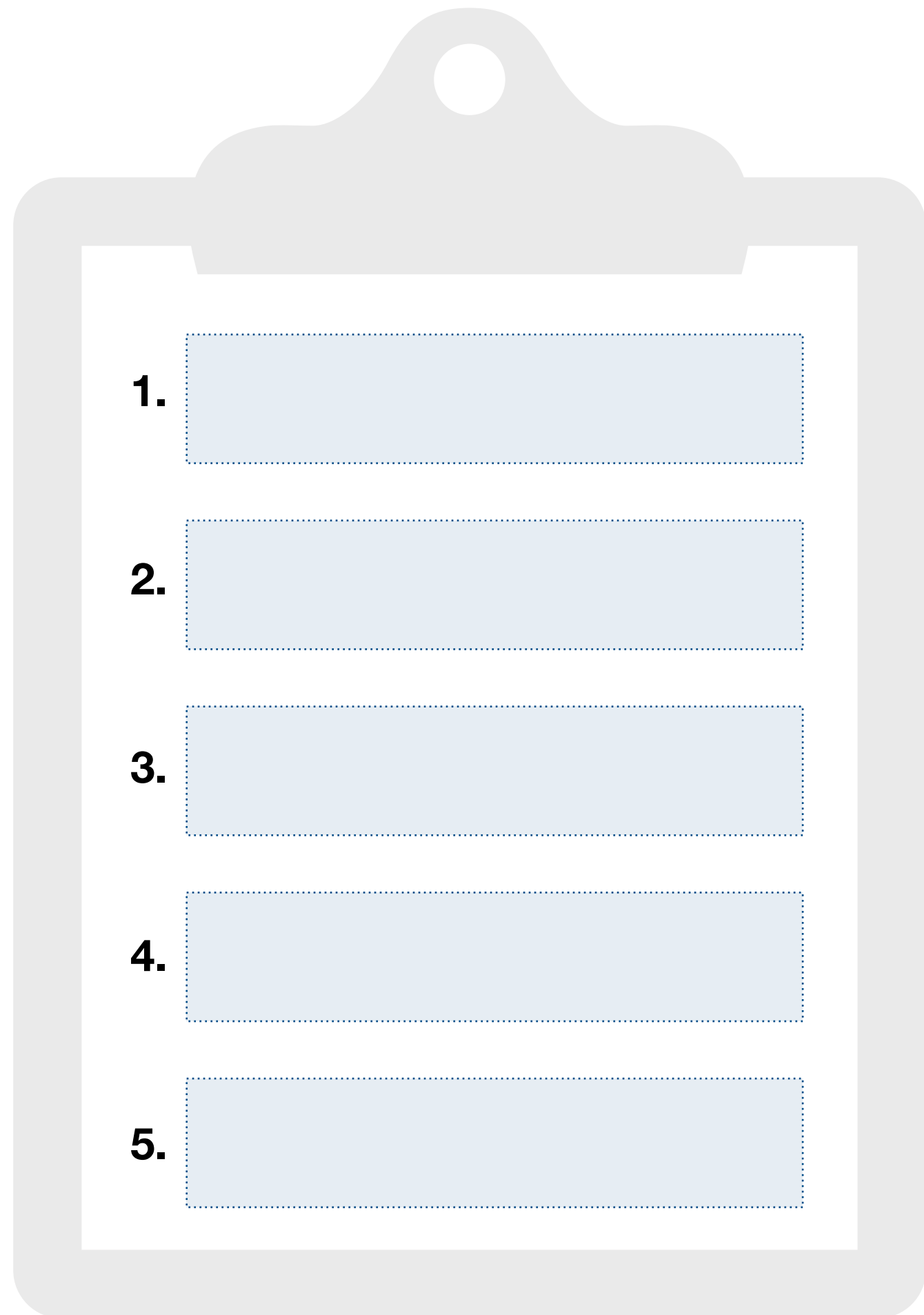Go back to your list of favorite apps and choose one to review. Think about the features that make it easy to use.

★★★★★

Review of my favorite app
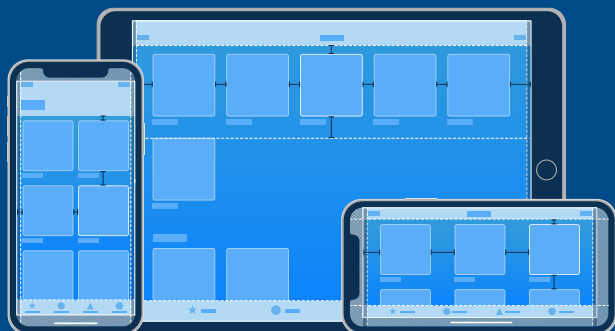
Go Further

Now consider the rest of the list of your favorite apps. Rank them in terms of their UI design. Which apps are easy to use and seem to just work? Write down the reasons that some apps are easier to use than others. Did you know what to do immediately? How many taps did it take to get going on the app? (The answer should be "very few.") First impressions count. Compare your notes with other students. Did you agree on the reasons?

1.

2.

3.

4.

5.

Plan

The basics

Keyboard

It's a very basic feature, but the keyboard is essential for many apps so users can input names, numeric data, and even emoji. This includes email, social media, or translation apps where you type a phrase and hear it in another language. What apps can you think of that use the keyboard? What kind of data do the apps take in?

Camera and microphone

Many apps use the camera and microphone to record sights and sounds. Think about apps that let you create movies, music, and photo albums. What about apps that let you communicate, like FaceTime and Messages? Or analysis apps, where you can overlay a graph onto a photo or mark it up for analysis? How many ways can you think of to use the camera and microphone?

Touchscreen

Both iPhone and iPad have a touch-sensitive screen. You can create apps that detect a user interaction, such as tapping the screen once, double-tapping, swiping, or dragging a button or an object. Think of the possibilities for games and other user interfaces that use touch as a natural interaction with elements on the screen.

How might you use these features in your app?

Plan

Get connected

Wi-Fi

Does your app need to connect to the internet to work? While most people may have access to Wi-Fi, think about what happens with your app when a user can't connect to a Wi-Fi network. Does having Wi-Fi access fit with your target audience persona?
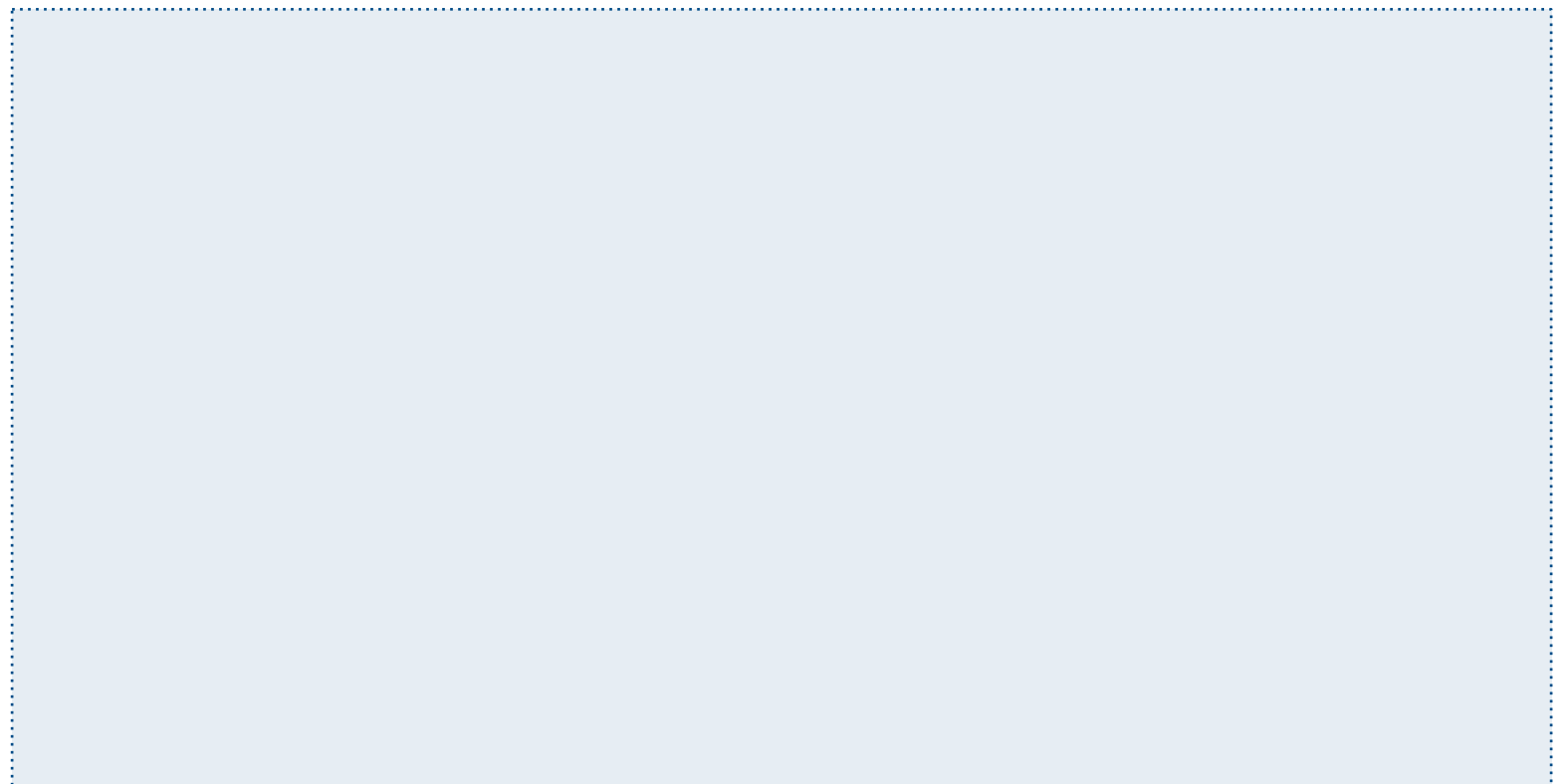
GPS

iOS devices have a built-in GPS (global positioning system) that shows where they're located on the earth within about 15 feet. It can also detect altitude (vertical distance from sea level). The Maps and Weather apps on your iPhone use GPS.

Bluetooth

This technology lets iOS devices connect with other nearby devices, such as a speaker to play music, a robot like Sphero that your device can control, or a digital thermometer.

How might you use these features in your app?

Plan

Get innovative

Speech recognition and machine learning

Try Siri out. Siri can recognize your speech. And if you continue to use Siri, you'll notice that it gets better at knowing what you want. That's machine learning. Which apps that typically use the keyboard to collect information can instead use speech recognition and machine learning? What types of audiences would benefit from these features?

Accelerometer and gyroscope

The accelerometer can detect whether a device is accelerating, decelerating, or in zero gravity. The gyroscope references direction, so it can measure the rotation of a device. Together, they can detect how iOS devices are being moved in three-dimensional space. Could you create an app that recognizes if the user is falling? Think about the Health app and the level tool in the Compass app on iPhone. How do they use the accelerometer and gyroscope?

Augmented reality

With augmented reality, you can blend digital objects and information with your real-world environment. Imagine seeing a life-size elephant right in your backyard or seeing the images in your favorite book come to life.

How might you use these features in your app?

Plan

Accessibility

iOS supports multiple ways to access onscreen content. People with physical impairments can use Siri or Dictation to interact with apps. People with visual impairments can increase the size and contrast of screen elements, or use the VoiceOver screen reader and navigate your app entirely by sound. But iOS accessibility features don't help only users with disabilities—they can help all users access your app in whatever ways they're comfortable with.

Try the iOS accessibility features so you know firsthand how they work.

How might you use these features in your app?
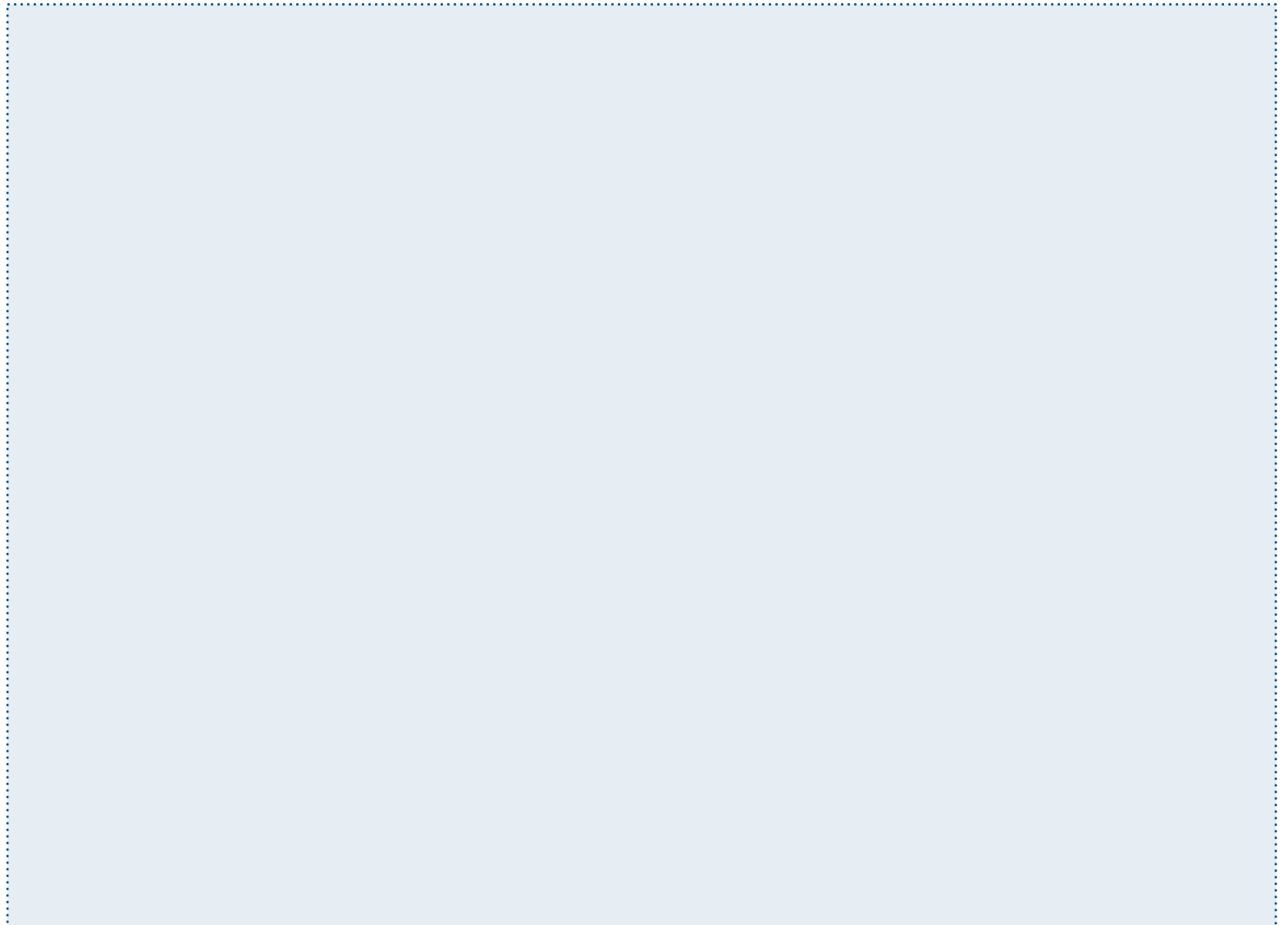
Plan

Feature smash

Now that you've learned about various features, it's time to see what combination works best for your app. First, stretch your imagination and try combining different features. Write down all the features on separate pieces of paper and fold them up. Take turns drawing at least two pieces of paper and coming up with ways to use those features together. For example, you could use the accelerometer and Bluetooth together to connect to a robot and use your device as the remote control.

Feature 1   Feature 2

Feature smash
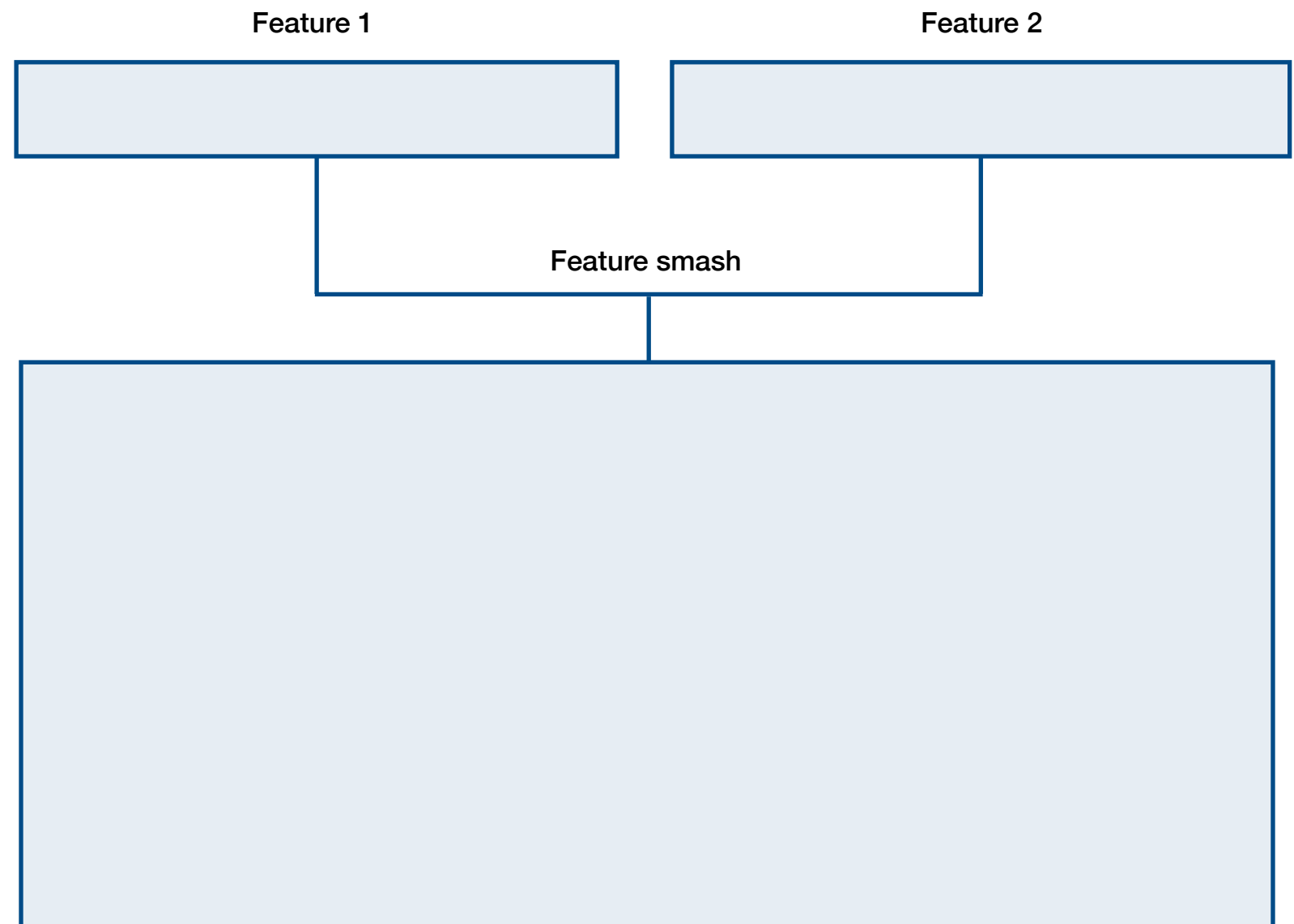
Now think about your app. Which features are essential to make your app state of the art? Which ones give it a wow factor? It could be, "Wow, that was so easy!" or "Wow, I've never seen that before!" Remember, you want your app to be unique, but also simple and easy to use.

Insert a new slide and map out what features you want to include in your app, along with how they help your app accomplish its goal.

Example

Trash Log
Keyboard: Text data table to categorize trash types

Marketplace
1. "Green products"
   • Ad revenue funds environmental restoration
2. Donation
3. Virtual forest items

Achievements
Medals to recognize achievements

Go Green App

Score
Algorithm that compares trash and recycling to similar households and awards points

Virtual Forest
Gamified element

Use points and rewards to add animals and plants

Challenges
Monthly challenges to help you live a greener lifestyle
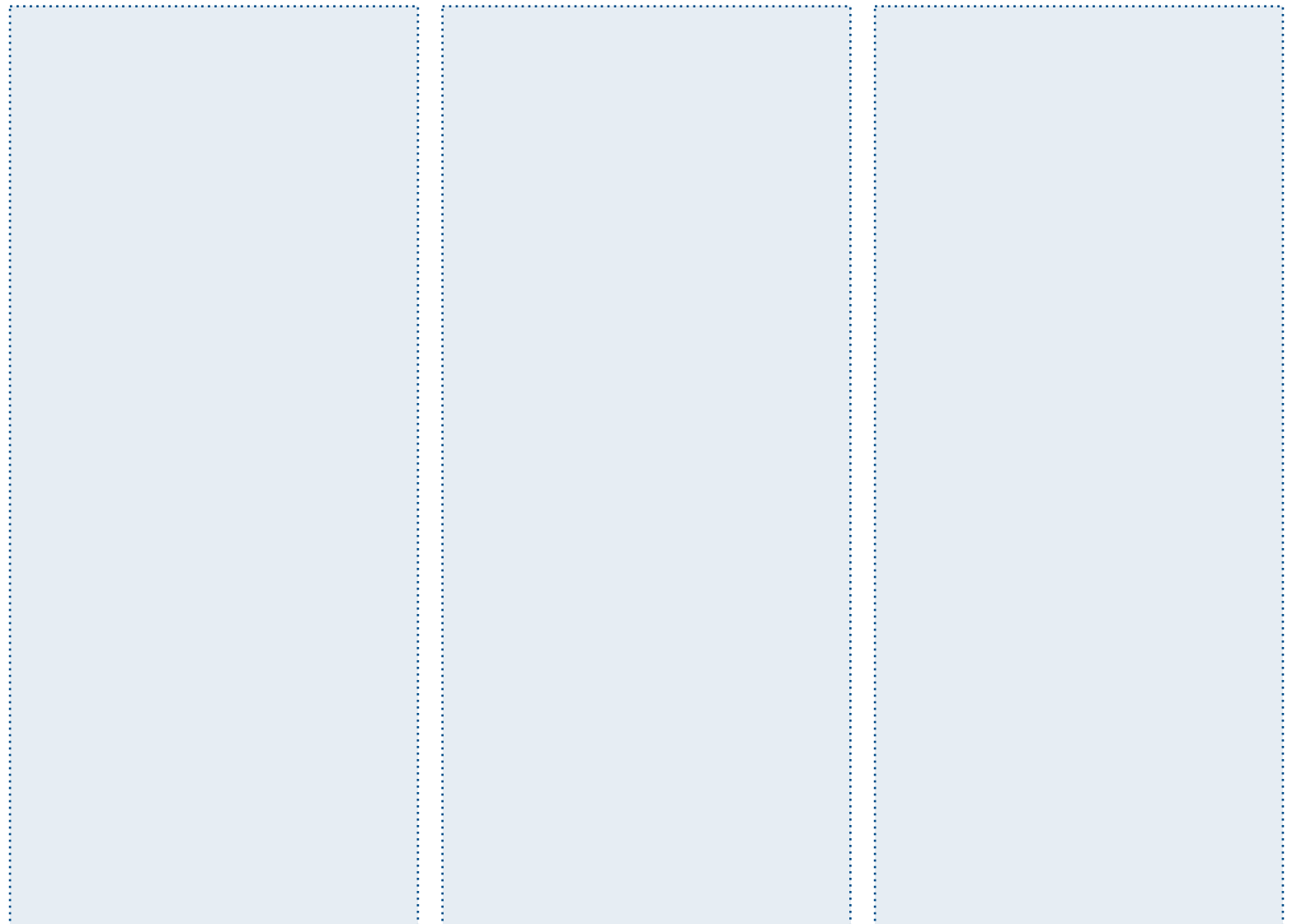
Share challenges with friends

Deeper Dive: The V in MVC

Model View Controllers (MVCs) are design patterns that organize an app's files, structures, and classes. Learn more in Unit 4, lesson 3.

Now that you've decided on the features, you need to think through exactly how, when, and where your app will use these features and how you can use code to achieve it. It's important not to get bogged down in the details of data modeling or how many controllers you'll need for a particular feature. For now, keep it simple: Make some simple sketches on paper, focusing on key views that are important to your app.

What views do you need to create a feature in your app? Draw or mock up the key views, thinking about these questions:

• How will you display the data of your app?

• How are you capturing input from the user?

• Do you want to show particular views on multiple screens?

Use your answers to these questions to explore alternative input methods, to determine if you're using too many different controls, and to locate views that you'll want to make reusable in code.

Plan

Be sure to review the Human Interface Guidelines again.

Design

Give your app some style and personality! But remember to keep it simple. You want the purpose of your app to shine through; you don't want too many colors or unnecessary elements to distract your users. Create a mood board to guide your app design.

Choose a color scheme.

Add example images of the types of visuals needed.

What fonts will your app use?

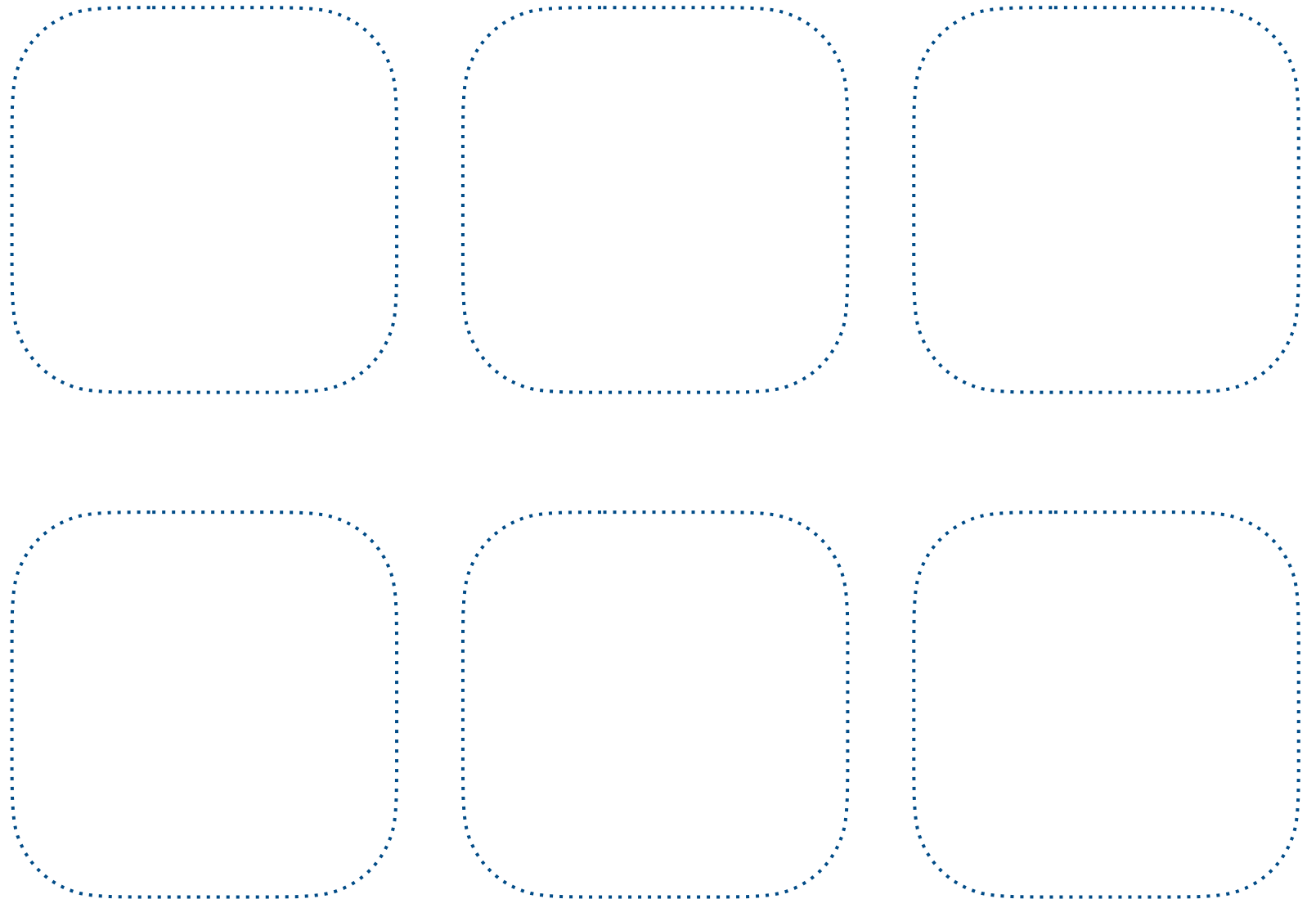Add phrases of design principles, such as "Keep it simple," "Modern," and so on.

Add sound files or describe the sounds your app will use to notify users of something, immerse them in a game atmosphere, or enhance the app mood.

An appealing icon matters to give a good first impression. When people are searching the App Store, they notice a good icon. You might have a brilliantly coded app with a slick user interface, but people will never download it if your app icon doesn't convey the right message.

Design a few different icons for your app. Show them to other students and ask them what they think your app does based on just the icon. Which one did they like the best? Pro tip: You can take a screenshot of your device's Home screen and add your icon to see if it stands out.

# Design a few different icons for your app.
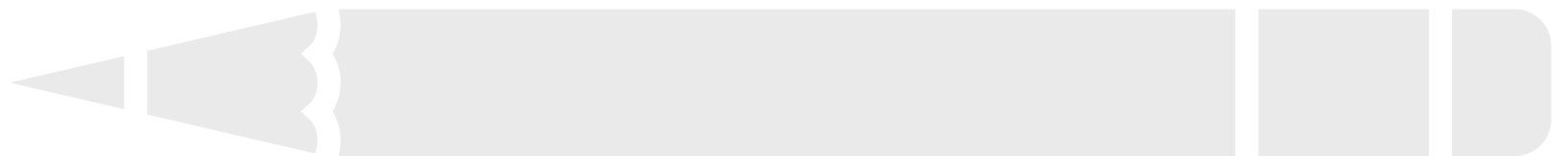
Prototype

Design
Storyboard
Build

Overview

Take a look at this video from WWDC on 60-second prototyping to find out how you can use Keynote to quickly test ideas. For a more in-depth exploration of app prototyping, watch this video from WWDC on iterative design. You don't need to watch the whole video, but it should give you a better idea of what to expect when prototyping with Keynote.
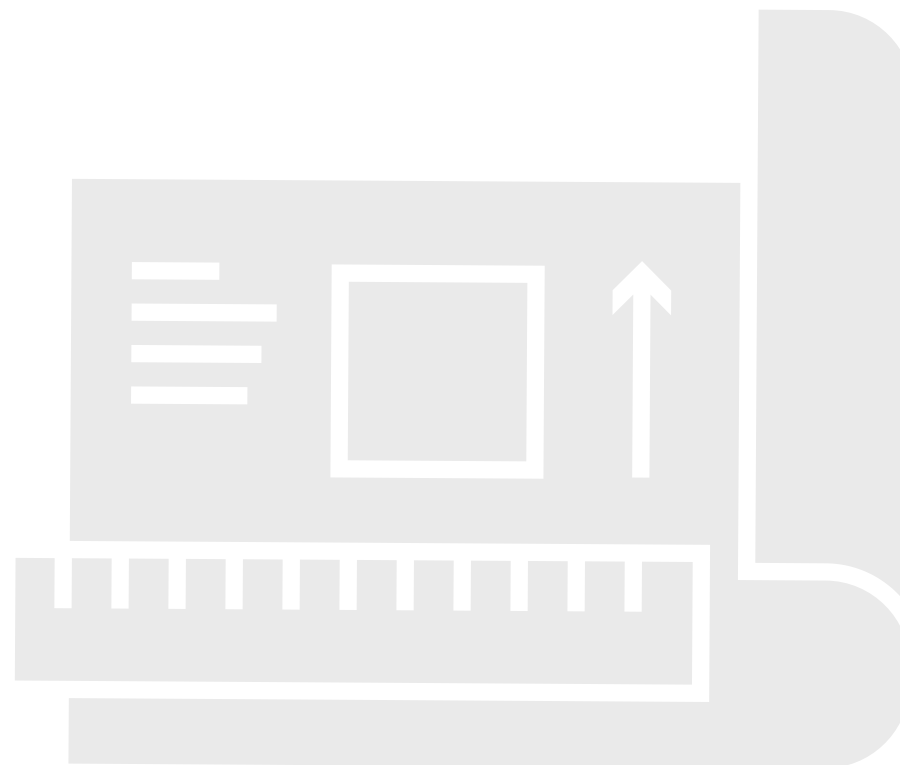
Prototype

Design

What do the main screens of your app look like? Which features appear where? Take a look at your design mood board again and mock up a few screens. You can use Keynote or a drawing app, or draw on index cards to create each screen. Take screenshots or photos of your images and add them to the app you'll build your prototype in.

Tip: Not feeling artsy? Use existing apps as inspiration. Take screenshots of app screens that have great features, add them as a layer or template, and customize them.

Home Screen
- Simple
- Welcoming
- Quick, cute animation for "Go Green"
- Clear entry point to app

(All "Get started" screens are green)

Data Setup
- Quick, easy data entry
- Big buttons
- Clear navigation

App Orientation
- Quick visual menu of app features
- Gallery scrolling is familiar
- Clear navigation

Virtual Forest
- Calming music
- Gentle colors
- Simple design
- Gentle animation
- Navigation menu is a constant; blue color indicates current screen

Trash Log
- Familiar calendar layout
- Large, clear pie chart
- Green is good
- + button opens more screens to add data
- "Business" end of the app—clear and efficient

Challenges
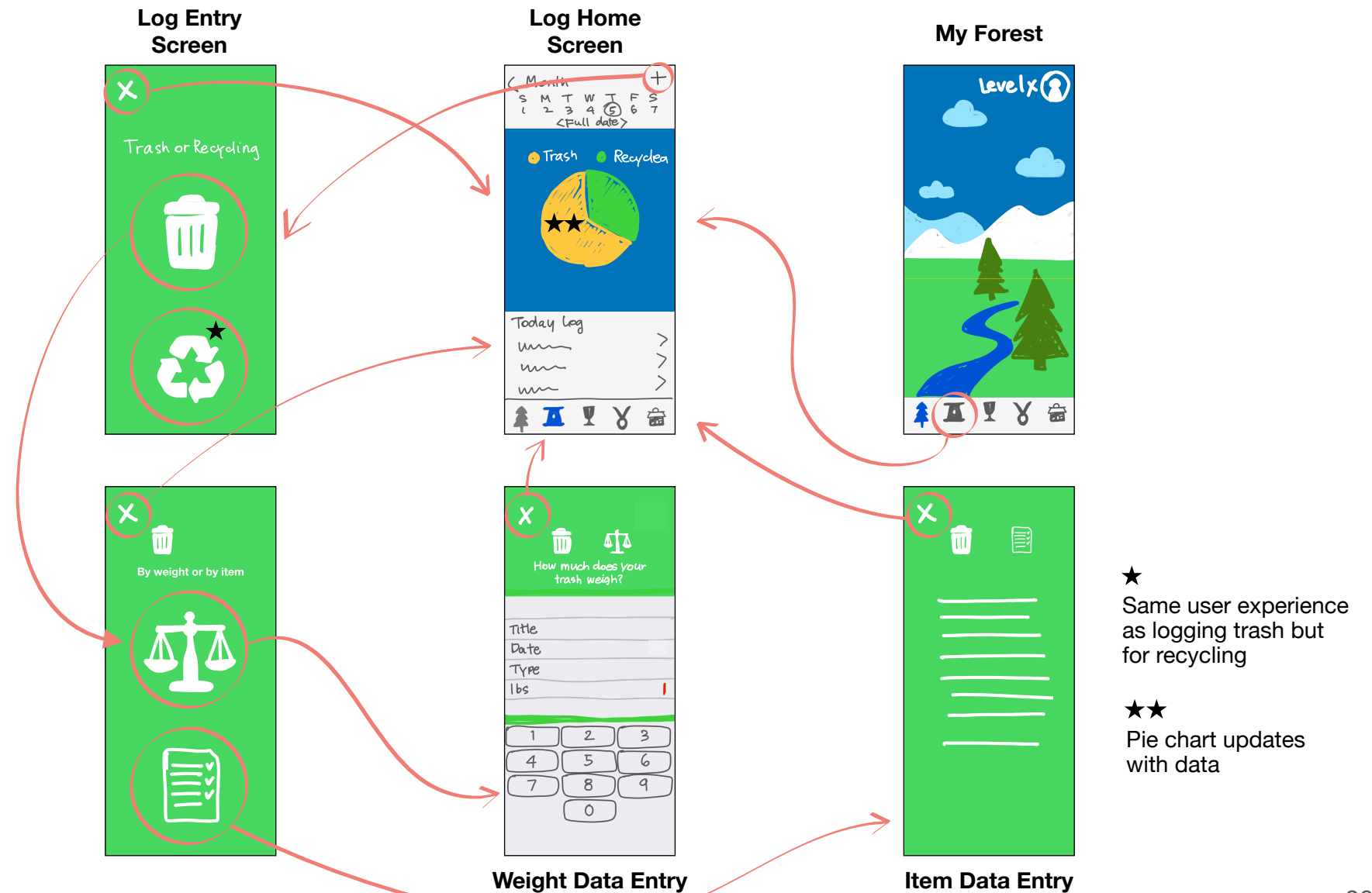- List view—scroll down to view most recent at top
- Simple illustrative graphic
- Separate screen for active and past challenges

Prototype

Design
Storyboard
Build

Storyboard

Insert a new slide, and create a flowchart like the sample below using the images you created. What are the key stages of the app, and how does the user get there? Be specific. For example, at what point is a certain feature available? Or what happens when a user taps *Yes* or *No*? What data does your app collect or provide and how is it represented? Map out the conditional statements needed for your app. For example, "If user taps *Yes*, then x; else, y." What other coding concepts would apply in your app? Do some parts repeat that would use a loop?

Log Entry Screen

Log Home Screen

My Forest

Weight Data Entry

Item Data Entry

★ Same user experience as logging trash but for recycling

★★ Pie chart updates with data

Prototype

Build

First take a look at these finished prototypes for inspiration. Next you'll use Keynote on a Mac or iPad or an iOS app like POP - Prototyping on Paper to build your prototype.

Tap or click to play the videos or download the iPad prototype and the iPhone prototype Keynote files for viewing.

Prototyping in Keynote

1. Set up the Keynote document to be the right size for your app prototype to run on your demo device. Click the Document tab in the Document sidebar, then in the Slide Size menu, choose Custom Slide Size. Enter one of these sets of dimensions:

- iPad: width=834 pts, height=1112 pts

- iPhone X: width=375 pts, height=812 pts

2. Decide on the colors and fonts you'll use, then design the navigation buttons. Park these design tools in working slides that you can delete later.

3. Build each screen on a different slide.

4. Create interactive links between the slides so that their buttons trigger touch events. Control-click the object you want to link, choose Add Link, then choose Slide.

5. To make sure that the presentation changes slides only when the user taps the navigation buttons, click Document, then in the Presentation Type menu, choose Links Only.

As you experiment with your own UI, you might also want to design apps that match the iOS design language. Download this Keynote document to access iOS UI elements.
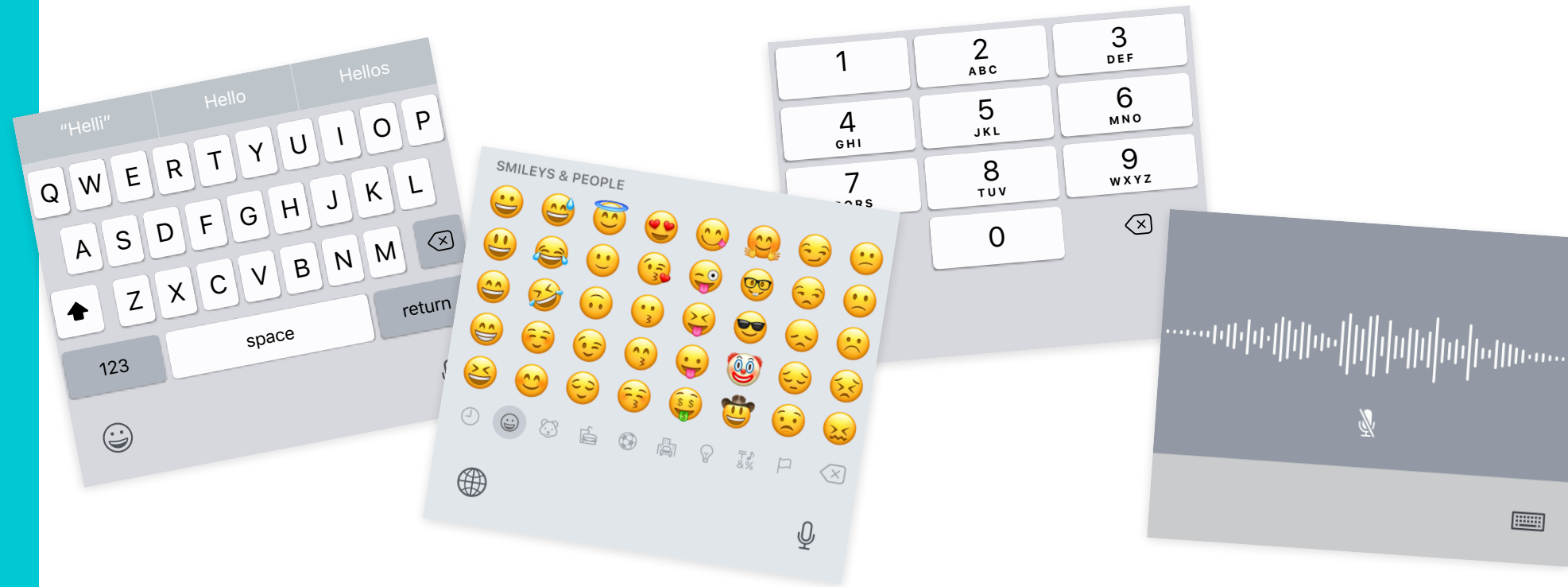
For each version of your prototype, think about the following:

- Can users choose to engage with the content in different ways?

- Can you provide different representations of the same data?

- What's the first screen (view) that the user sees? Which buttons are visible? Then what happens?

- Decide what kinds of graphics and icons your app will display and where they'll appear.

- How many taps will it take for users to find out what they need to know?

- How would users navigate between views?

- What are some simple ways to communicate the features of your app without using words?
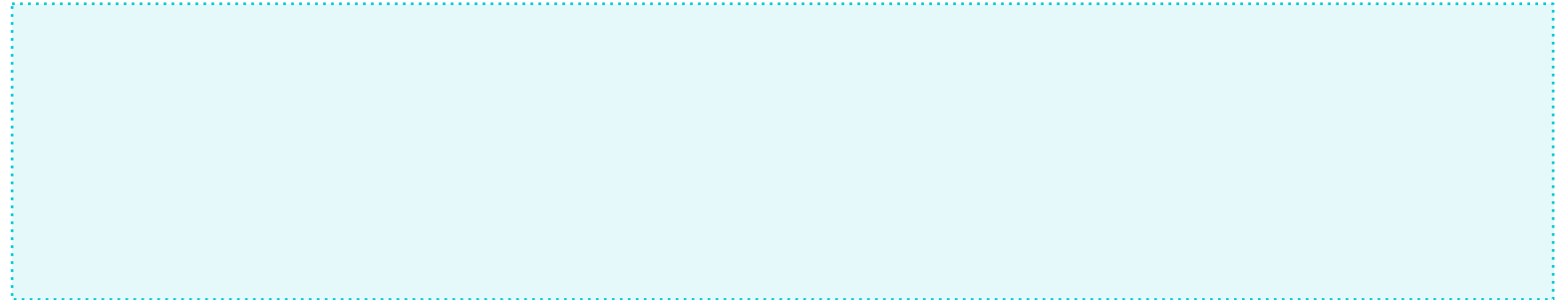
Deeper Dive: Detailing MVC

Once you've defined how your app will flow in your prototype, think about how to organize different parts of your code. The Model-View-Controller pattern helps accomplish this while keeping your code organized.

Use the following questions to help build a list of models, views, and controllers that your app would need if you began building the prototype with code.
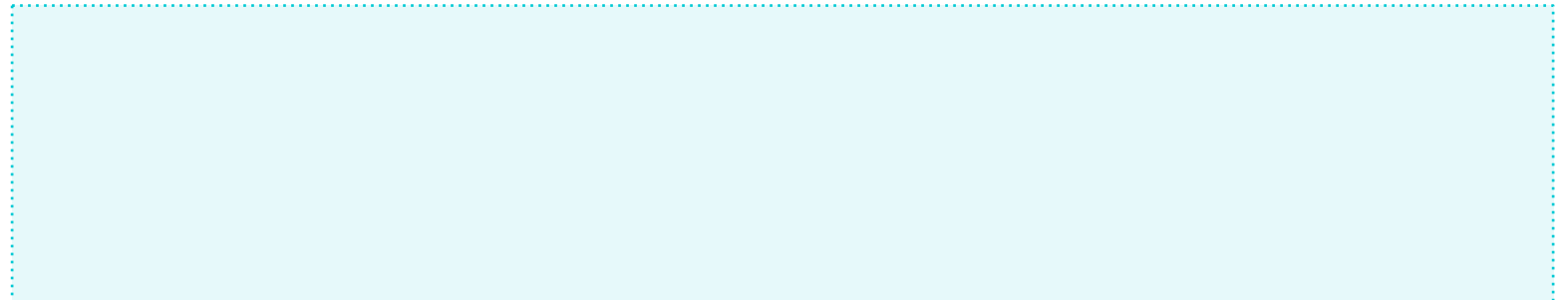
Models

What data do you need to build your features? Where does the data come from? Does the user supply it or does it come from a web service? Do you need to store the data on the device for offline access?
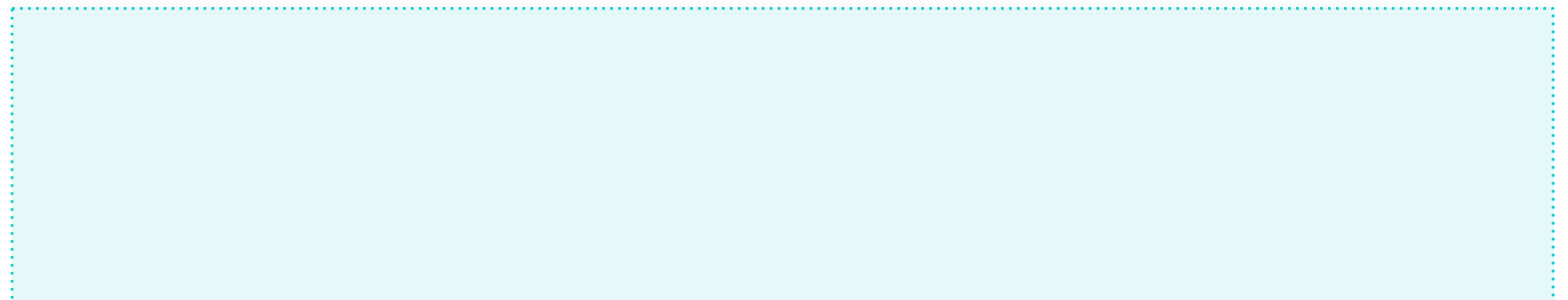
Views

Are there particular views you want to show on multiple screens? Did you include any custom gestures?

Controllers

How many view controllers does your app need? What controllers will help manage the data? Does your app have custom transitions that need a controller?
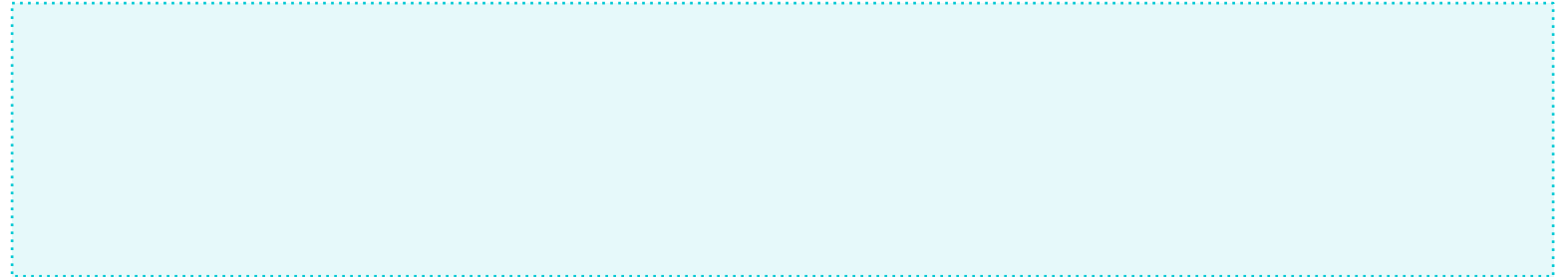
Deeper Dive: Using Xcode
One of the benefits of using simple prototyping tools, such as drawing and Keynote, is that it's very easy to make changes based on feedback. However, there's no rule that says you can't build portions of your prototype in an Xcode project, using Interface Builder and storyboards to piece together all the screens in your app. When should you consider reaching for an advanced developer tool for prototyping?
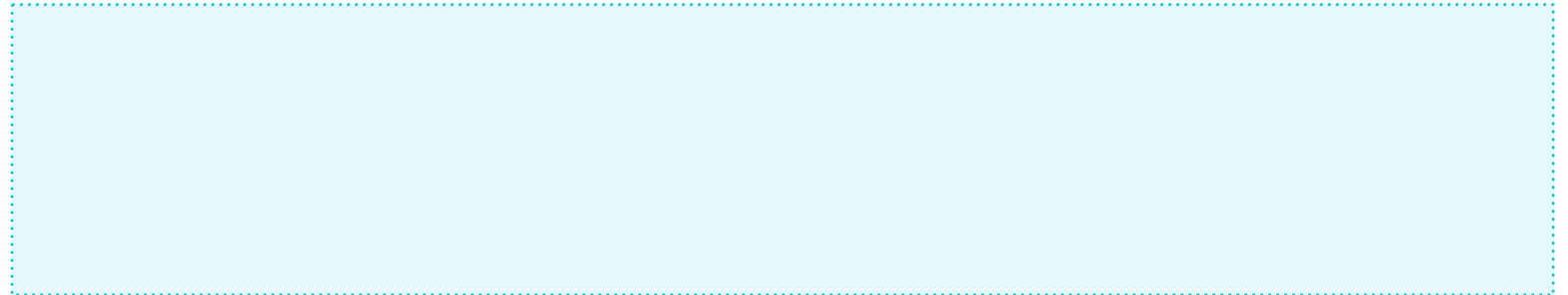
Keynote prototyping is great for communicating the UI of your app, but it's difficult to emulate all UX elements, especially app interactivity and feedback. Building parts of your app in Xcode enables you to demonstrate how the app responds to genuine user interactions. In your Xcode project, you might choose to focus on coding just one particular app function—you don't need to build the whole app or perfect the entire UI.

What sections of your app might you want to write code for first? Outline a plan for building your app. Look back at your storyboard.

Which views are most important and need to work first?

Which views might you want to save to build at the end?

Evaluate

Observation
Interview

Overview

Now it's time to test your prototype. You can have your classmates, family, and others try it. If possible, try to find testers who fit the target audience for your app. You should present your prototype, explain your new app idea, and tell the testers that you want them to try it. You can provide guidance if needed, but the objective is to observe the user and ask questions later.

Evaluate

Observation
Interview

Observation

Watch the tester explore your app, and use the questions below as guidelines for recording your observations.

Did the user know what buttons to tap?

Was the user ever confused? At what point?

Did the user enjoy the app?

Did the user smile or laugh at specific points?

Did you observe anything else?

Evaluate

Observation
Interview

Interview

Interview the user when they finish testing your app to better understand their experience. Here are a few questions to get you started:

✓ What did you like and not like about the app?

✓ Is the app useful? Would you use an app like this?

✓ What else might you want to see in this app?

Brainstorm

Purpose
Ideas
Audience
Focus
Reiterate

Reiterate

Remember, this is a design cycle and it's time to go back to the brainstorming stage. As you repeat the design cycle, think about what you learned from your evaluation. Did problems come up, and if so, how can you fix them? How can you improve your app?

Another important question to ask yourself is whether you're still excited about your app idea. If not, it might be time to go back to your list. Not all ideas pan out. One objective of the design cycle is to help you test concepts and figure out what's worth pursuing.

Do you still want to continue with your idea? If so, write the name of your app below, give it a star rating, and write an app review.

★★★★★

Review of my app

Go Further

Revisit the criteria you noted in the Purpose topic for what makes an app great, then answer the questions at right.

Continue to revisit the different topics throughout the design cycle. Revise your prototype accordingly, testing and retesting until you have the next great app.

Is your app innovative?

Does it do something that existing apps don't do?

Is it an app someone would use over and over again?

How can you improve your app?

35

App Pitch

You've tested and improved your app idea. Now it's time to polish it up and share it! Make a three-minute presentation or video of your pitch. A good pitch will tell a strong and clear story that makes people want your app!

Your pitch should include:

• **Why:** The problem your app is trying to solve

• **Who:** A description of who your app is for

• **What:** An overview of the app

• **How:** Details about the UX and UI, including:

  - The design

  - The features

  - The coding concepts it uses

  - The prototype and any visuals

  - Improvements made based on user testing