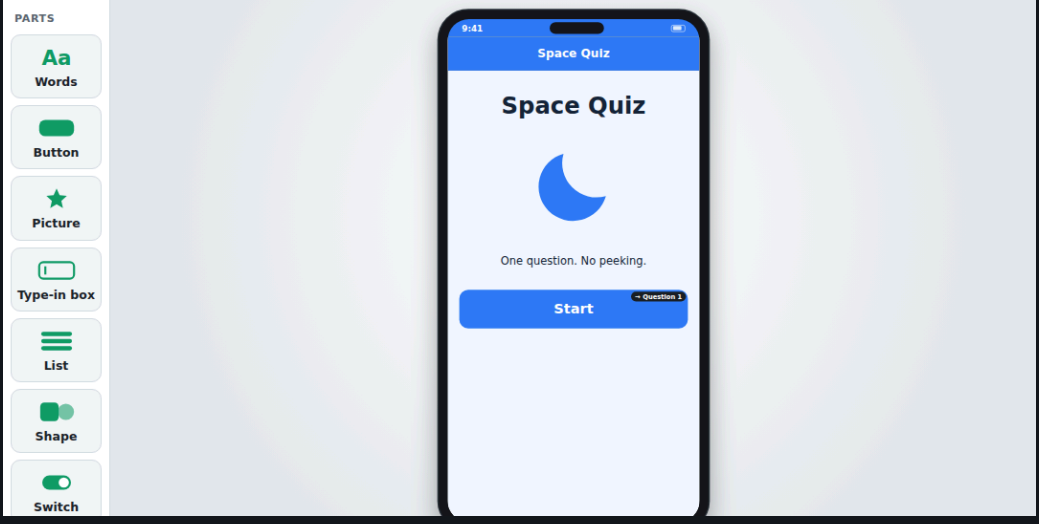
mytekDev

App Maker: Every Button Goes Somewhere

Grades 6-8 · 45 minutes

Computer science · Technology



Slides, rubric and answer key **Print and go** **No student accounts**

Chromebook friendly

What's included

- A one-page overview: objectives, essential question, vocabulary, standards and assessment
- A ten-minute walkthrough to rehearse before class, and background in plain words for the teacher
- A 45-minute lesson sequence: warm-up, direct instruction, guided practice, independent practice, closure
- Slides for the board: a PowerPoint file that also opens in Google Slides
- A student worksheet with a QR code and a table of buttons and a map to draw
- A 3-question exit ticket, two to a page, with an answer key
- A rubric: 3 criteria, 4 levels, scored out of 12
- Differentiation for English learners, students with IEPs or 504 plans, and students who need more support or more challenge
- Cross-curricular extensions, and notes for teaching it in grades 4 and 5 and in grades 9 and 10
- Standards: CSTA 2017

App Maker: Every Button Goes Somewhere

Grades 6-8 · 45 minutes · Computer science, Technology

Students open a small quiz app: four screens and six buttons. It looks finished. Two of its buttons go nowhere. Students use the app the way a person would, find the buttons that do nothing, and fix each one by telling it which screen to go to. Then they draw the app as a map, with a box for each screen and an arrow for each button, and add a second question of their own.

At a glance

ESSENTIAL QUESTION

What happens in an app when you tap a button, and who decided that?

OBJECTIVES

Students will be able to:

- describe an app as a set of screens joined by buttons
- explain that a tap is an event, and that nothing happens unless the app was told what to do when it happens
- find a bug by testing an app the way its user would
- fix a button by telling it which screen to go to, and test it again

VOCABULARY

Screen, Event, Navigation, Bug, Test

STANDARDS

2-AP-17, 1B-AP-15, 1B-AP-10

TIME AND MATERIALS

45 minutes, grades 6-8.

- One computer or Chromebook per student
- A projector
- The worksheet and exit ticket, one copy each per student
- The rubric, one copy per student if you grade the work

PREP

Make copies of the worksheet and exit ticket. Before class, open mytekdev.com/try/appmaker?lesson=every-button-goes-somewhere&ref=tpt-appmaker-1 once on a student computer: it opens on a quiz called Space Quiz, and checks that your school's filter allows the app. Run it yourself, about ten minutes: press Try it, tap Start, then Venus (nothing happens); press Keep designing, click Venus and pick Not that one under Then go to; then find the second dead button (Play again, on the Right! screen) and point it at Space Quiz.

ASSESSMENT

- Formative: the questions under What to look for, asked while students work, and the exit ticket (answer key included).
- Summative: the worksheet, scored with the rubric.

TECHNOLOGY AND PRIVACY

Runs in any browser, Chromebooks included, with nothing to install. Students need no account and no email: the link works signed out, and nothing they do is stored on our server unless they choose to save. The app they build runs on a phone drawn on the page, so no phones are needed. Making a link for a real phone needs a free account and is not part of this lesson.

Lesson sequence

MINUTES	STAGE	WHAT YOU DO	WHAT STUDENTS DO
0 to 5	**Warm-up** Hook	Show the title, essential question and Today I can slides (about 1 minute). Then the warm-up: when you tap a button, how does the app know what to do? Write two answers on the board.	Answer the warm-up on the worksheet.
5 to 12	**Direct instruction (I do)** Model it	Project the app with the starter open. Point at the four cards under the phone and click each one: an app is screens, and the phone shows one at a time. Click the Start button on the phone and read its panel aloud: Then go to, Question 1. Say that a tap is an event, and this line is what the app does when it happens. Do not fix anything yet. **It's working when:** In Design, clicking a button picks it. Only in Try it does a button do its job. Press the same big button to go back and forth.	Watch, and fill in the vocabulary on the worksheet.
12 to 22	**Guided practice (We do)** Find it, fix it	Students open the link or scan the QR code. A short guide walks them through finding the Venus button, fixing it, and hunting for the second one. When most have found it, ask: how did you find it? **It's working when:** Try it opens the app on the screen being designed, which is handy for testing one screen. From the start, beside Keep designing, goes to the app's first screen. **Watch for:** Students who look for the second bug by reading the panels in Design. That works here with six buttons. Ask how they would find it in an app with sixty: by using it.	Follow the guide. Fix Venus, then find and fix the second button.
22 to 38	**Independent practice (You do)** The table and the map	Students fill in the table: every button, the screen it is on, and where it goes. Then they draw the map. Early finishers add a second question: Ready-made under the phone, then Question, and point Play again at it. Walk the room with the What to look for questions. **If they ask:** "Which screen has the most arrows coming into it? Which button takes you back without naming a screen?" The map	Fill in the table, draw the map, and add a question if there is time.
38 to 45	**Closure** Showcase and exit ticket	Show the Showcase slide. Two students say how they found the second button. Go back to the warm-up answers. Hand out the exit ticket.	Answer the exit ticket.

Before you teach: a ten-minute walkthrough

Open mytekdev.com/try/appmaker?lesson=every-button-goes-somewhere&ref=tpt-appmaker-1. A short guide opens over the app. The phone shows a quiz called Space Quiz, and four cards sit under it: Space Quiz, Question 1, Right!

and Not that one.

1. **Use it.** Press Try it. Tap Start, then Mars: the app says Not that one, and Try again goes back to the question. Tap Saturn: Yes! Saturn. Tap Play again: nothing happens. Reload the page to start over.
2. **Find the first one.** Press Try it, tap Start, tap Venus. Nothing happens. Press Keep designing: you are still on Question 1.
3. **Fix it.** Click Venus on the phone. The panel on the right shows Then go to, with Nowhere chosen. Click Not that one. A small dark tag appears on the button.
4. **The second one.** Click the Right! card under the phone, click Play again, and pick Space Quiz under Then go to.
5. **The table.** Start goes to Question 1. Mars goes to Not that one. Saturn goes to Right!. Venus goes to Not that one. Play again goes to Space Quiz. Try again goes Back one screen.

> The moment that lands is tapping a button that does nothing. Students expect an app to know what a button is for. It only knows what it was told.

Background for the teacher

Programs that people use sit and wait. Nothing runs until something happens: a tap, a key, a timer. That something is called an event, and the code that runs when it happens is the event's handler. In App Maker the handler of a button is its panel: the steps under When tapped, first, and the screen under Then go to.

A button with Then go to set to Nowhere is not broken in any way the app can see. It does exactly what it was told, which is nothing. That is what most bugs are: the program doing what was written and not what was meant. The only way to find one is to use the program and compare what happens with what should.

Try again uses Back one screen and names no screen. It goes wherever the person came from. On this quiz that is the same as naming Question 1. In a quiz with two questions sharing one Not that one screen it is not, which is why it is there.

App Maker is not a text programming language. Students build each rule as a sentence and the app writes the code, which they can read and cannot edit. It has no loops and no lists, a button holds one But if, and numbers are the only kind of data an app remembers. What it does have is the real thing in small: events, a sequence of steps, variables and a conditional, in an app that runs.

What to look for

Ask these while students work. A student who has understood the lesson can answer them without help:

- What does this app do when nobody taps anything?
- The Venus button looked fine. What was wrong with it?
- How did you find the second broken button?

- Which button goes back without naming a screen? Where does it go?
- If you add a second question, which buttons have to change?

> The third question is the real test: a student who says "I used the app and tapped everything" has the idea of testing. One who says "I looked at every button's panel" found it too, and should be asked how that would go in a bigger app.

Cross-curricular extensions

- English: a choose-your-path story is a map of the same kind. Draw the map of a short story with two choices before writing it.
- Math: an app has 5 screens and every screen has a button to each of the others. How many buttons is that?
- Science: write a three-question quiz about the unit you are studying, with a screen that says why each wrong answer is wrong.
- At home: pick an app on a family phone and draw the map of its first three screens.

Vocabulary

Screen	One page of an app. An app shows one screen at a time.
Event	Something that happens that an app can react to, such as a tap on a button.
Navigation	Moving from one screen of an app to another.
Bug	A mistake in a program that makes it do something its maker did not mean.
Test	Using a program on purpose to find out whether it does what it should.

Differentiation

ENGLISH LEARNERS

- Keep the vocabulary slide up, and point at the screen as you say each word: the cards under the phone and the sentences in the panel carry the meaning.
- Pair each learner with a partner; the table is words copied from the buttons and the cards.
- Accept arrows drawn on the map in place of the last column of the table.

STUDENTS WITH IEPS OR 504 PLANS

- Pair students for the table, one at the app and one recording.
- Zoom the browser to 125%; the phone and the panel stay readable.
- Give the map with the four boxes already drawn, where the IEP allows a modified assignment.

STUDENTS WHO NEED MORE SUPPORT

- A sentence starter: "When I tap ___, the app goes to ___."

STUDENTS READY FOR MORE

- Add a second question with Ready-made, then Question. Which buttons have to change so the quiz

- A check-in after the guide: every student has pressed Try it and Keep designing at least twice before starting the table.
- Give the first row of the table filled in: Start, on Space Quiz, goes to Question 1.

asks both?

- Make a wrong answer on your new question go back with Back one screen. How is that different from naming the screen?
- Change the quiz to a subject of your own: the question, the three answers and the picture.

Adapting for other grades

Younger students: In grades 4 and 5, do the guide together on the projector, with pairs sharing a computer, one clicking and one recording. Skip the third rubric row; total 8, and 6 meets the objectives. Draw the map together on the board before students draw their own.

Older students: In grades 9 and 10, have students press See the code after fixing a button and find it under "What is on it": the button carries the name of the screen it goes to. Ask what the app would do with a button that names a screen that was deleted, then try it.

Student worksheet

Name: _____ Date: _____

Open the app: scan the QR code, or go to mytekdev.com/try/appmaker?lesson=every-button-goes-somewhere

Essential question: What happens in an app when you tap a button, and who decided that?

Warm-up: When you tap a button in an app, how does the app know what to do?

_____

1. **Vocabulary.** Screen, event, navigation, bug, test: write each in your own words.

2. **The button table.** For each of the six buttons write: the words on it, the screen it is on, and where it goes when tapped. Put a star by the two you fixed.

3. **The map.** On the back, draw a box for each screen and an arrow for each button.

1. The Venus button did nothing when it was tapped. Was the app broken, or doing what it was told? Explain.

2. How did you find the second button that went nowhere? Why is using the app a good way to find a bug?

Exit ticket

Name: _____

1. An event is (A) a screen in an app (B) something that happens that an app can react to, such as a tap (C) a mistake in a program

2. A button does nothing when it is tapped. The most likely reason is (A) the phone is too slow (B) nobody told the button where to go or what to do (C) the button is the wrong color

3. Name one way to find a bug in an app.

App Maker: Every Button Goes Somewhere

Exit ticket

Name: _____

1. An event is (A) a screen in an app (B) something that happens that an app can react to, such as a tap (C) a mistake in a program

2. A button does nothing when it is tapped. The most likely reason is (A) the phone is too slow (B) nobody told the button where to go or what to do (C) the button is the wrong color

3. Name one way to find a bug in an app.

Answer key: exit ticket

QUESTION	ANSWER	WHY
1	B	A tap is an event. The app does nothing until one happens.
2	B	An app does only what it was told. Then go to was set to Nowhere.
3	Use the app and tap everything	Accept: test it, try every button, have someone else use it. "Look at the code" is fine with a reason.

Rubric

Score each row from 1 to 4.

Name: _____

CRITERIA	4 EXCEEDS	3 MEETS	2 APPROACHING	1 BEGINNING
The button table	All six buttons, each with its screen and where it goes after the fixes.	At least five buttons recorded correctly.	Three or four buttons recorded.	Fewer than three.
Finding and fixing	Both dead buttons fixed, and says how the second was found by testing.	Both dead buttons fixed.	One dead button fixed.	Neither fixed.
The map	A box for every screen and an arrow for every button, each arrow pointing the way the button goes.	Every screen, with most arrows right.	Some screens and arrows.	No map.

Total: ____ / 12. A score of 9 or more meets the objectives.

Standards

Algorithms and programming standards from CSTA 2017. Students test a small program, find two errors in it and fix them. The grade 3 to 5 standard is listed because many grade 6 students meet these ideas for the first time.

FRAMEWORK	CODE	GRADES	STANDARD	HOW THE LESSON MEETS IT
CSTA 2017	**2-AP-17**	6-8	Systematically test and refine programs using a range of test cases.	Students test every button in an app, record where each goes, find the two that go nowhere and test again after fixing them. The test cases are the six buttons.
CSTA 2017	**1B-AP-15**	3-5	Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended.	Students find two errors by using the app and fix each one.
CSTA 2017	**1B-AP-10**	3-5	Create programs that include sequences, events, loops, and conditionals.	Students work with events (a tap) and a sequence of screens. This lesson has no loops and no conditionals.

Checked November 2026 against: CSTA 2017 K-12 standards progression chart.

Optional: save student work in a free class

The whole lesson runs at the link on the worksheet, with no accounts and no sign-up.

To keep students' work, and see it all in one place, make a free class. It takes about ten minutes, once, and covers every lesson in this series: mytekdev.com/tools/class-setup

Students in a class open mytekdev.com/tools/app-maker/lessons/every-button-goes-somewhere/start instead of the worksheet link. Each one gets their own copy of the starter, and the same link reopens it next time.

Terms of use

Thank you for downloading this lesson. The plan, the handouts and the art were made by mytekDev.

You may:

- print and copy it for your students
- change it for your own classes
- post it on your class website or learning management system
- share this file with other teachers, whole and unchanged, with this page included

You may not:

- sell it, or put it in a product you sell
- share a changed copy
- claim it as your own

The app this lesson uses is free and runs in the browser. Students need no account. It is at mytekdev.com.

Questions, or found a mistake? Ask on the Teachers Pay Teachers Q&A for this resource, or write to us at mytekdev.com/contact.

© 2026 mytekDev. All rights reserved.