U4 Practice PT - Decision Maker App Planning Guide

Project Description

For this project you will create an app that helps a user make a decision. Your app must take in at least one number and one string from the user that will help to make the decision. All of this information will be used as part of the decision making process. In addition, your code must include at least one function used to update the screen.

You will submit
- Your final app
- A video that shows your program running
- This completed project-planning guide

App Requirements
- At least one number and one string used to make and report a decision with a conditional statement
- A function which updates the screen and is called at least twice in the program
- Conditional statement includes at least one logical operator (&&, || or !)
- There are at least three different possible output answers (i.e. "Yes, you can adopt a cat!", "No, you can't adopt a cat", and "Congratulations, you can adopt a kitten!).
- Every function contains a comment explaining purpose and functionality
- Clear and easy to navigate user interface
- Cleanly written code which is free of errors

Steps
- Brainstorm an app idea for making a decision
- Interview classmates for ideas on what information would be needed to make the decision
- Draft a flowchart of the decision making process
- Design your app's user interface
- Design and program your app in App Lab
- Collect feedback from your classmates and update your app
- Record a screen capture of your app being used ●   Submit your final app

Investigate

Step 1. Brainstorm App Ideas: Your app should be designed to help a user make a decision. For this project your user is your classmate. The decision can be small or big, like what to eat for lunch or where to apply for a job. Keep in mind how your idea might help solve a problem for your user.

Idea 1:
What shop Cap should go to if he wants to buy meat?

Idea 2:
How to determine if someone is ready to launch a TVC(Thrust Vector Controlled) rocket?

Idea 3:
What potential problems user might be face in particular situations?

Step 2. Choose One Idea: Talk through your ideas with a classmate. Pick the one that you are most interested in.

App Idea: The app that makes decision for people who wants to launch a rocket, I put together several questions/ problems might be cause trouble launching the rocket in the app.

Step 3. Survey Your Classmates: To design your app you'll need to understand your users. For this project your user is your classmate, and you'll need to understand what information will be needed to make the decision.

Find two classmates and talk to them about your topic for a couple minutes. Then fill in this table.

Name	What information is needed to make this decision?
Cap	If the rocket is stable enough.
Jaymin	If the software is good enough for the rocket to control itself.

Step 4: Storing information: What variables will be used to store information?

Name	Information Stored	Variable Type (string, number, Boolean)
Answer1	If the user wants to start the app	number
Answer2	The answer for question 1	number
Answer3	The answer for question 2	number
Answer4	The answer for question 3	number
Answer5	The answer for question 4	number
Switch	If the user wants to start the app	Boolean
Software	If the software is ready	Boolean
Hardware	If the hardware is ready	Boolean
stability	If the rocket is table	Boolean
trajectory	If the trajectory is predicted	Boolean
output	The final output	Boolean
void result()	Output function	function
void responseCollection ()	The final output	function
void init()	App initialization	function
struct Decision	the structure of the overall app, easier to declare in the int main(){}	structure

Step 5: Flowchart: Draft a flowchart to show the decision making process

Example:
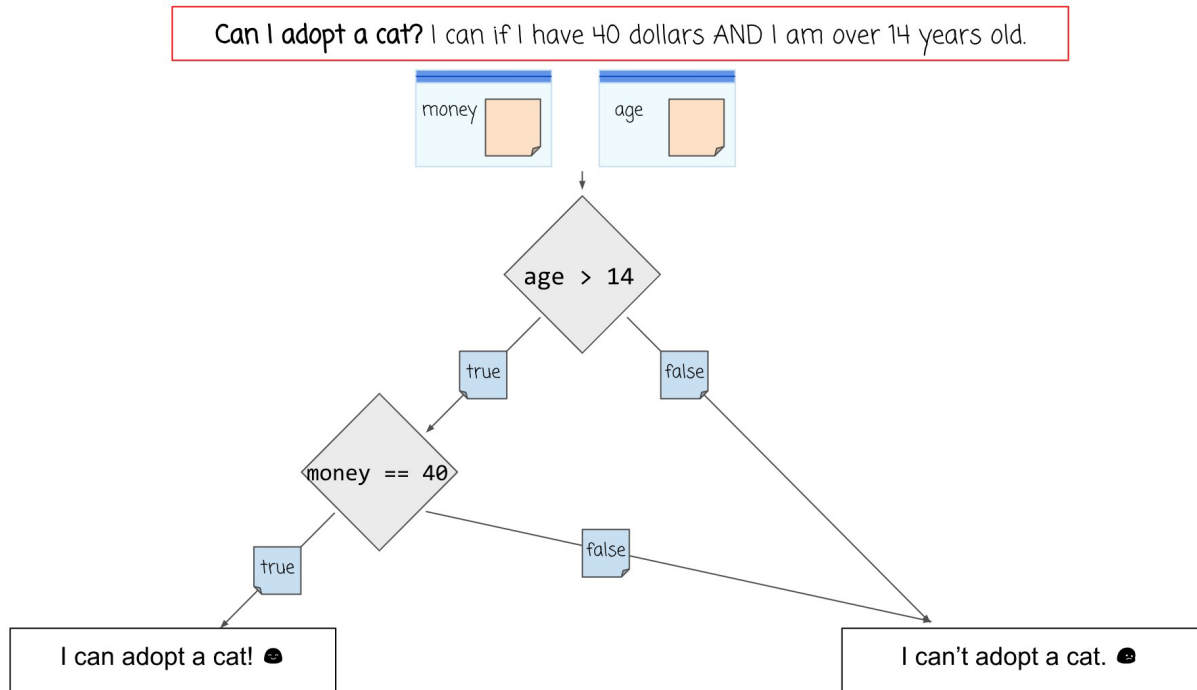
Component	Purpose
Can I adopt a cat? I can if I have 40 dollars AND I am over 14 years old.	Start your flowchart with a question. What decision are you trying to make?
money age	Baggies are used to represent the variables which store information. In your flowchart, draw a small rectangle for the variables.
age > 14	A diamond represents a decision point, based on the original question. Write the Boolean expression that will be used to determine the answer.
true	True and False arrows designate the paths taken, based on the result of a decision (diamond). Note that every decision may have only 2 possible paths that result from it, one for true and one for false.
I can adopt a cat! 😊	A rectangle at the end of a decision path represents a possible result.

↓	A **simple arrow** indicates that we are moving from one action to the next without considering any decision. These will generally be used to link a set of actions to be completed one after the other.
Flowchart	

Design

Step 6. Design User Interface: In the space below draw a rough sketch of your user interface. This means you should include all the buttons, text, and images that the user will be able to use. Write notes or draw arrows showing how different user interface elements should work.

Note: There are no screen requirements for this app - you may use one or more screens.

Prototype

Step 7. Start Building Your App: Build your app. Along the way make sure you:
- Use the design you drew as a starting point, but it's OK to update as you go.
- Reference the flow chart when setting up your conditional statements
- Use your debugging skills to check that your app is working
- Comment all functions explaining purpose (what does it do) and functionality (how does it work)

Test

Step 8. Testing: You will need to test your app to make sure it works as expected. To do that find at least two classmates to use your app. While they use the app watch them to see if anything is broken or confusing. Afterwards ask them to share any specific improvements they'd like to see.

Name	Things that could be improved based on watching them use the app	Improvements this person recommends
Mr. Compton	Not enough comments	

Cap	Too little explanations, and too technical.	

Step 9. Pick Improvements: Pick at least one improvement you plan to make to your app based on feedback you collected from your classmate.

Improvement 1:
Made more comments and explain the questions better.

Improvement 2 (Optional):
Step 10: Complete Your App: Finish your app!

Step 11: Record Video. Record a video that demonstrates the running of your program as described below. Your

video may NOT contain voice narration and must be no more than 1 minute in length Your video must

demonstrate your program running, including:

- ☐ Input to your program
- ☐ At least one aspect of the functionality of your program
- ☐ Output produced by your program

Question 1: Provide a written response that:
- describes the overall purpose of the program
- describes the functionality of your app
- describes the input and outputs of your app

The overall purpose of the app is to help user decide if today is a good day to launch a rocket. The app is fault tolerant, meaning if the user put in answers system doesn't like, the app will not crash. The app is consisted of 4 questions with a main "switch"—the first question—to determine if to execute the quiz or not. The overall feedback was good. The app automatically translates user response into Booleans that will be out into the final equation which equates all the Booleans together. If the answer is true for all the answers, the system will output yes. On the other hand if there is one false, the system will say no because rocket launch cannot tolerate a single mistake made. The app consist of Booleans, strings, numbers, conditional functions for fault tolerance, structures and logistical equations.

Question 2: This project was created using a development process that required you to incorporate the ideas of your partner and feedback from your classmates. Provide a written response that describes one part of your app that was improved through input from EITHER your partner or feedback you received from classmates. Include:
- Who specifically provided the idea or recommendation
- What their idea or recommendation was
- The specific change you made to your app's user interface or functionality in response to the recommendation
- How you believe this change improved your app (Approx 150 words)

After I finish the app and showed Mr. Compton, he said the app needs more comments as explanations so it would make the app clearer to the user. I add comment for all the variables and many other import part of the function. I also created a Github Repository. In the README.md, I explained the overall function and

use of the app so it is clear to the user. Everyone is free to edit the software, and the app is open for the public to use. I also attached to .cpp file to the repository to people can download my code to make changes when needed. They can even turn my code into their own codes that is for their own projects.

Rubric

Category	Convincing Evidence	Approaching Evidence	Limited Evidence	No Evidence
App Development Planning Guide:	Planning guide is fully completed.	Planning guide is mostly completed.	Planning guide is somewhat complete.	Planning guide is not complete.
Video	Video shows the program running including input, program functionality, and output.	Video shows the program running and two of the following: input, program functionality, and output.	Video shows the program running and one of the following: input, program functionality, or output.	No video was made.
Written Response 1:	Response accurately describes the purpose, functionality, and inputs/outputs of the app.	Response describes the purpose and functionality, or the inputs/outputs of the app.	Response partially describes the purpose and functionality, or the inputs/outputs of the app.	Response does not describe the purpose, functionality, and inputs/outputs of the app.
Written Response 2:	Response clearly describes an idea or recommendation provided by a partner / peer and how it improved the app.	Response describes an idea or recommendation provided by a partner / peer and how it improved the app, lacking clarity.	Response describes an idea or recommendation provided by a partner, but does not explain how it improved the app.	Response does not describe an idea or recommendation provided by a partner.
User Interface:	The User Interface is easy to navigate and it's clear how the app is designed to be used. All text is readable.	The User Interface is mostly easy to navigate and it's clear how the app is designed to be used. All text is readable.	The User Interface is lacking in some readability or it's not clear how to use the app.	The User Interface is difficult to navigate and it's not clear how the app is designed to be used. Text is unreadable.
Code: Warnings & Error Messages	No warnings or error messages appear when the app is run.	A few warnings or error messages appear when the app is run..	Many warnings or error messages appear when the app is run.	The app does not run at all.
Code: Variables	At least one number and one String are each stored in a variable and used to make a decision.	One data type (numbers or Strings) is stored in at least two variables and used to make a decision.	One variable stores either a number or String and is used to make a decision.	No variables are set up or used to make a decision.
Code: Function	A function is used to update the screen. The function is called at least two times in the program.	A function is used to update the screen. The function is called one time in the program.	A function is created to update the screen but is not called in the program.	A function was not created to update the screen.
Code: Conditional	A conditional is used inside of the function to make a decision based on information stored in variables. The conditional correctly uses a logical operator (&&, ||, or !) in the Boolean expression. The decision is displayed on the screen. There are at least three different responses that could be displayed.	A conditional is used inside of the function to make a decision based on information stored in variables. The conditional does not correctly use a logical operator (&&, ||, or !) in the Boolean expression. The decision is displayed on the screen. There are at least two different responses that could be displayed.	A conditional is created inside of the function, but does not use information stored in variables to make a decision or display it on the screen.	No conditionals are present in the function.

Code: Comments	The update screen function has a comment which clearly explains its purpose and functionality.	The update screen function has a comment which clearly explains its purpose or functionality.	A comment is present, but it does not clearly explain anything about the function.	No comments are present.