Name(s) Brent Lee Period 2 Date 9/20/20

App Development Planning Guide

Project Description

For this project you will work with a partner(s). Together you will create an app that teaches your classmates about any topic you find interesting. Along the way you'll learn how to use many of the features of App Lab as well as skills that will help you when building more apps throughout this class.

You will submit

- Your final app
- This completed project-planning guide

App Requirements

- Uses at least three screens
- Includes examples of images, audio, and text
- A clear and easy to navigate user interface
- Clearly communicates information about your topic
- Screens are cleanly written and free of errors

Steps

- Collaborate with your partner(s) to pick a topic you are both interested in
- Design your app's user interface using this planning guide
- Design and program your app in App Lab
- Collect feedback from your classmates or family and update your app
- Share your final app with the class

Investigate and Reflect Phase

Step 1. Brainstorm Topic Ideas: Your app can teach your classmates about any topic you and your partner agree on. Your topics could be a hobby, something you've always been interested in, a piece of your personal history, or just something you think your classmates should learn more about.

Write down three ideas for a topic that you brainstorm individually.

Idea 1: Computer Quiz

Idea 2: Minecraft Quiz

Idea 3: Valorant Quiz

Step 2. Choose One Topic: Now talk through your ideas with your partner. Together pick a topic both of you are interested in teaching your classmates about. Explain in a few sentences what would be covered. For example, if your topic is Basketball, you would write a few sentences explaining that you would cover the rules and the origin of the sport.

Our Topic: Minecraft

The information that will be covered in our quiz is some less known facts about minecraft. A lot of people already know

the common facts about Minecraft and I want to cover the less known information to make the app more useful.

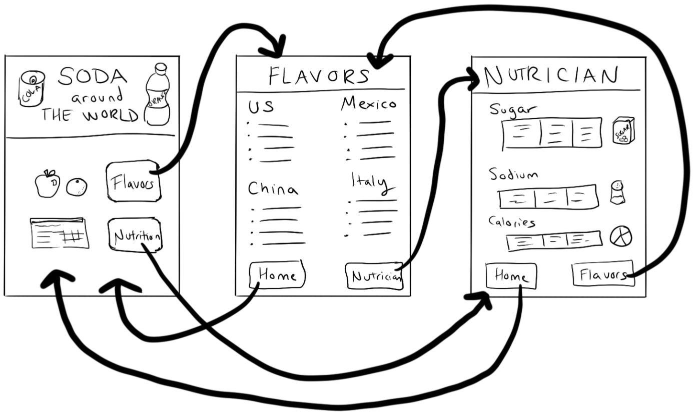
Step 3. Interview Your Classmates, Friends and or Family: To design your app you'll need to understand your users. For this project your user is your classmates, and you'll need to understand what they already know about your topic.

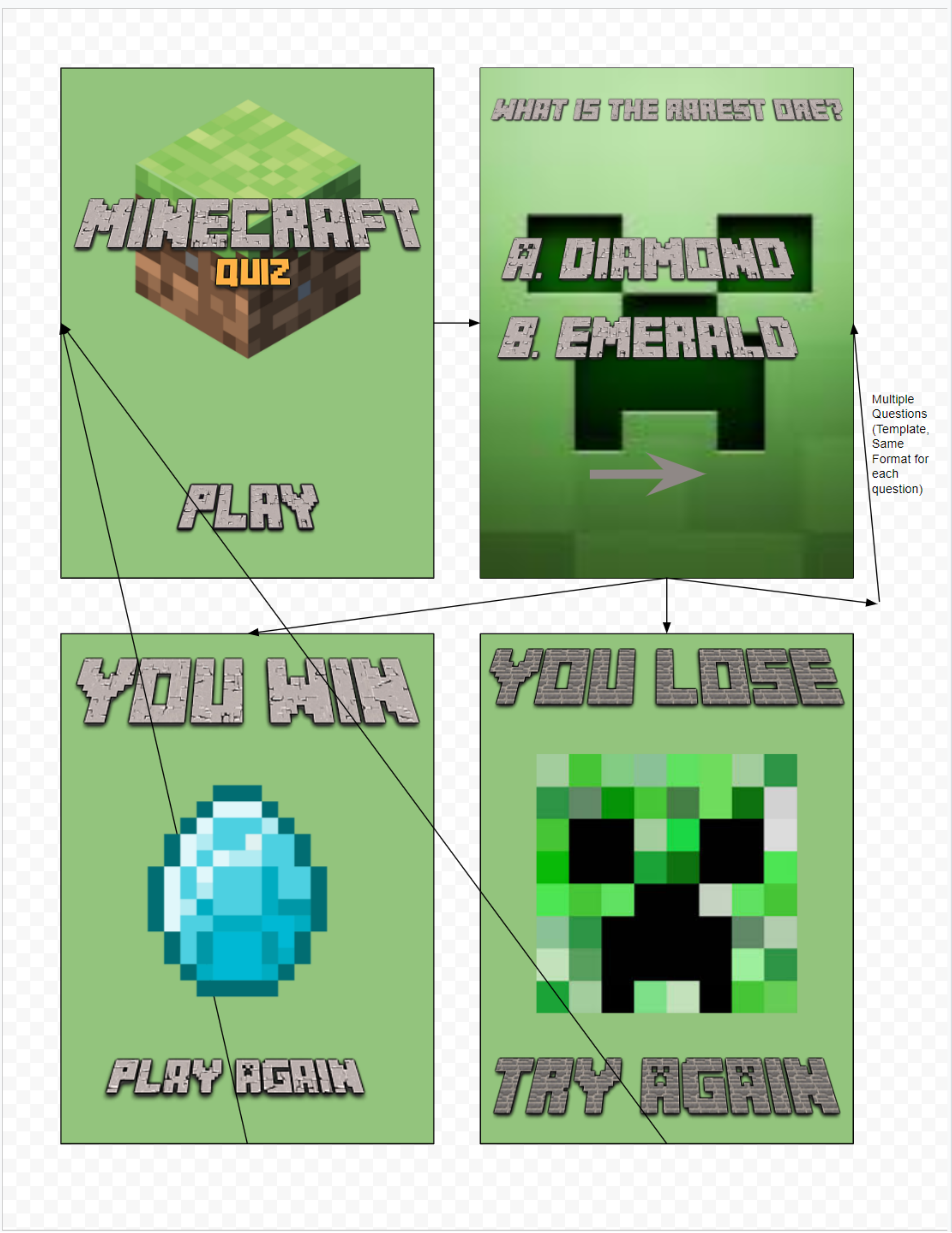
Find two classmates and talk to them about your topic for a couple minutes. Then fill in this table

Name	What do they already know about your topic?	What do they need or want to learn about your topic?
Jarrett	They have played the game before and thus have previous experience which will help the creation of the app.	They want to learn some new and interesting facts regarding the game.
Ariya	They enjoy the game recreationally and have previous experience with the game.	They are interested in learning more about the game and remembering old facts and tricks that were forgotten.

Design Phase

Step 4. Create a Program Specification: Based on your research you identified **requirements** for what your app must teach your classmates. On the next page you should draw a **specification** that shows how your app will actually run to meet those requirements. 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. For example, if clicking a button takes me to another screen, I should draw an arrow from that button to the drawing of the screen.





Multiple Questions (Template, Same Format for each question)

Prototyping Phase

Step 5. Start Building Your App: Work with your partner(s) to build your app.. Along the way make sure you:

- Use the program specification you drew as a starting point, but it's OK to update as you go.
- Use your debugging skills to check that your app is working

Before you begin to code, fill out the chart below for any Event Handlers in your program:

Element ID	Action	What happens?
"dogButton"	*"click"*	*A picture of a dog appears* *The background of the screen changes to green*
menuPlay	click	Screen changes to the screen "question1"
question1a1	click	Screen changes to the screen "question2"
question1a2	click	Screen changes to "loseScreen"
question1a3	click	Screen changes to "loseScreen"
question2a1	click	Screen changes to "loseScreen"
question2a2	click	Screen changes to the screen "question3"
question2a3	click	Screen changes to "loseScreen"
question3a1	click	Screen changes to "loseScreen"
question3a2	click	Screen changes to the screen "question4"
question3a3	click	Screen changes to "loseScreen"
question4a1	click	Screen changes to "loseScreen"

question4a2	click	Screen changes to “loseScreen”
question4a3	click	Screen changes to the screen “winScreen”
tryAgainButton	click	Screen changes to “menuScreen”
playAgainButton	click	Screen changes to “menuScreen”

Use the chart to guide you in adding programming statements to your program.

Testing Phase

Step 6. Testing & Feedback: You will need to test your app to make sure it works as expected. To do that find at least two classmates or family members to use your app.

- Ask students to read through your program specification and requirements
- Ask them to use the app and test out the different behaviors included in your specification
- Write down anything you noticed them finding confusing or broken
- Ask them to share anything they recommend improving

Name	Things that could be improved based on watching them use the app	Improvements this person recommends
Beck Robinson	Couldn’t find anything that they had trouble with using the app. They used it flawlessly.	Add sound to make the app more interesting
Ryan Luong	Couldn’t find anything that they had trouble with using the app. They used it flawlessly.	Add more images

Step 7. Pick Improvements: Pick at least one improvement you plan to make to your app based on feedback you collected from your classmate.

Improvement 1: Add more images to the app

Improvement 2 (Optional): Add more sound

Step 8: Complete Your App: Finish your app!

Reflection

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

(Approx 150 words)

Write your reflection here

The overall purpose of my app is to quiz and inform the user on Minecraft facts. Currently, as the app has not been coded yet, it only functions as a design. After it is coded, the app will function as a fun quiz to inform the user on some interesting Minecraft facts. After pressing play, the user will answer some questions about the game. If they get one question wrong, they lose and they have to restart. If they get every single question correct, then they win and continue to the win screen. If they want to, they can play again. When the start button is pressed, that information is processed and it is coded that it will proceed to the first question screen. If the user inputs the wrong answer, then the user loses and can choose to try again. If the user inputs the correct answer, then the user wins and can also play again.

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)

Write your reflection here

The people that specifically provided the idea or recommendation were both of my partners, Jarrett and Ariya. They said that they had played the game before and had prior experience with the game. They recommended to me that I include harder questions or use information that not everyone already knows. The specific change that I made to my app's user interface or functionality in response to the recommendation was that I decided to make harder questions. Also, to make it more difficult, I decided to provide three answer options instead of two. For example, instead of including an easy question like "What happens if you try to sleep in the nether," I added a question like "What mobs drop the most experienced orbs?" I believe these changes improved my app because most people do not know this information. Therefore, my app is more fun for them and they learn more from it.

Rubric

Category	**Extensive Evidence**	**Convincing Evidence**	**Limited Evidence**	**No Evidence**
User Interface Screens	User interface includes at least three screens	User interface includes two screens.	User interface is on a single screen.	The screen is blank.
User Interface Navigation	The user can easily navigate between all screens.	The user can easily navigate between most screens.	The user can easily navigate between some screens.	The user cannot navigate between screens.
User Interface Elements	The app includes at least one example each of: - Text - Image - and audio.	The app includes at least one example of two of the following: - Text - Image - And audio	The app includes at least one example of one of the following: - Text - Image - And audio	The app includes no text, images, or audio.
Code	Code runs without errors	Code runs with a few errors.	Code does not run or has a lot of errors.	Code is blank.
Element IDs	Screen elements all use meaningful IDs.	Screen elements mostly use meaningful IDs	Some screen elements use meaningful IDs	Screen elements do not use meaningful IDs.
App Topic	Topic is clearly communicated and explained.	Topic is somewhat clearly communicated and explained.	Topic is not communicated well.	App appears to be a random collection of elements with no clear topic.
App Development Planning Guide:	Planning guide is fully completed.	Planning guide is mostly completed.	Planning guide has a few parts completed..	Planning guide is empty.
Written Response 1:	Response accurately describes the purpose, functionality, and inputs/outputs of the app.	Response mostly describes the purpose, functionality, and inputs/outputs of the app.	Response is not complete, but does describe the purpose, functionality, or inputs/outputs of the app.	Response does not address the prompt in any way or is blank.
Written Response 2:.	Response fully describes an idea or recommendation provided by a partner / peer and how it improved the app.	Response mostly describes an idea or recommendation provided by a partner / peer and how it improved the app.	Response is not complete, but does describe some of the work with a partner.	Response does not address the prompt in any way or is blank.