Name(s)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Period \_\_\_\_\_\_ Date \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |
| --- | --- | --- |
|  | **App Development Planning Guide** |  |

## 

## Project Description

For this project you will work with a partner. Together you will create an app that teaches your classmates about any topic you both 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
* Code is cleanly written and free of errors

**Steps**

* Collaborate with your partner to pick a topic you are both interested in
* Interview classmates to identify what they already know about the topic
* Design your app’s user interface using this planning guide
* Design and program your app in App Lab
* Collect feedback from your classmates 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: How to do woodturning. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Idea 2: How to code some basic C. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Idea 3: A quiz on computer science topics from Unit 3. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Step 2. Choose One Topic:** Now talk through your ideas with your partner. Together pick a topic both of your 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: How to cde some basic C.

The team (I) will talk about how to set up a basic main.c file. I will explain what header files are and what functions are. This wont go into much detail since I only have a few pages and little time to complete this, but at the end a working basic program will be made. People will be able to follow along and learn the very basic of coding in C.

**Step 3. Interview Your Classmates:** 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

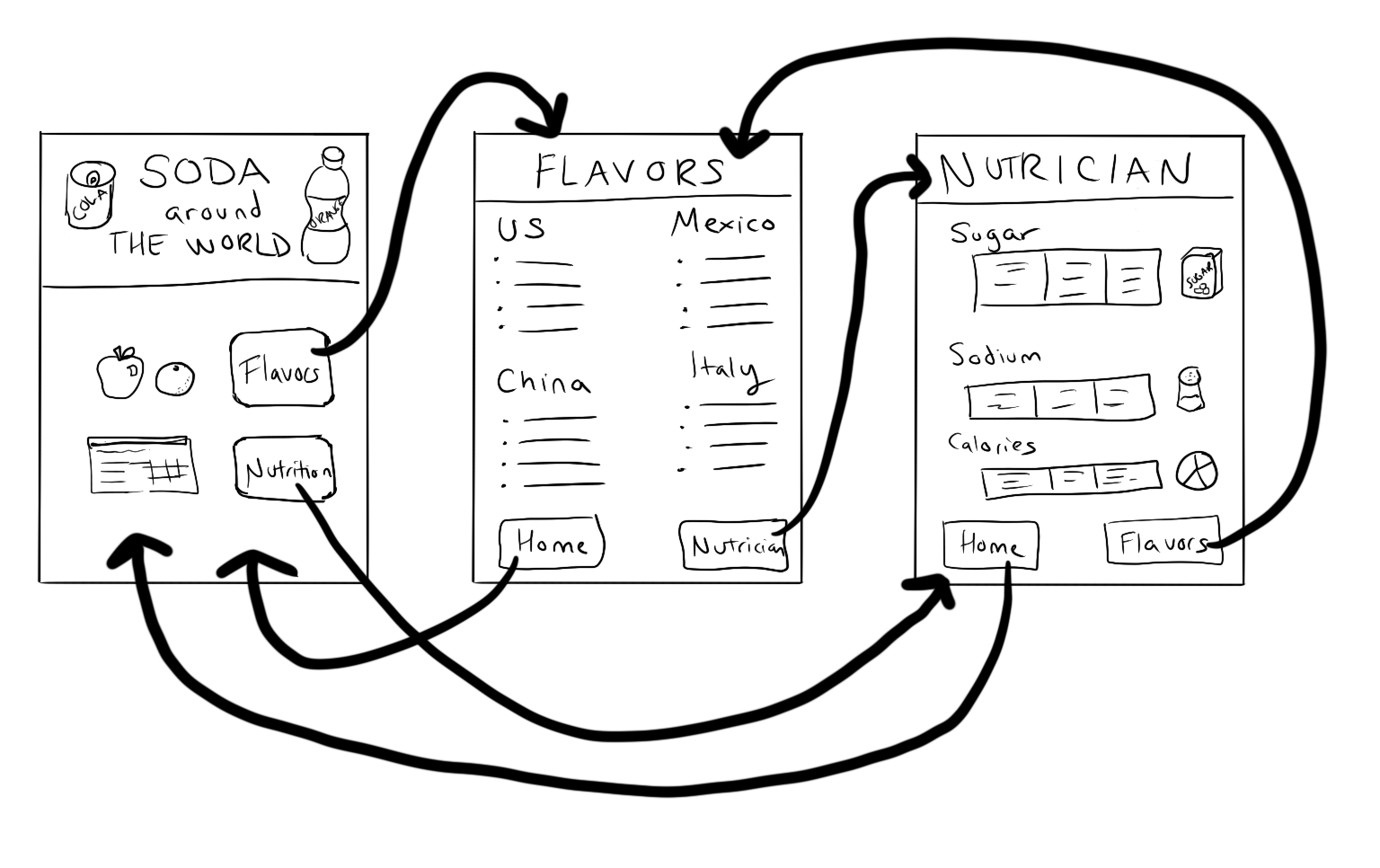
\* I used my parents

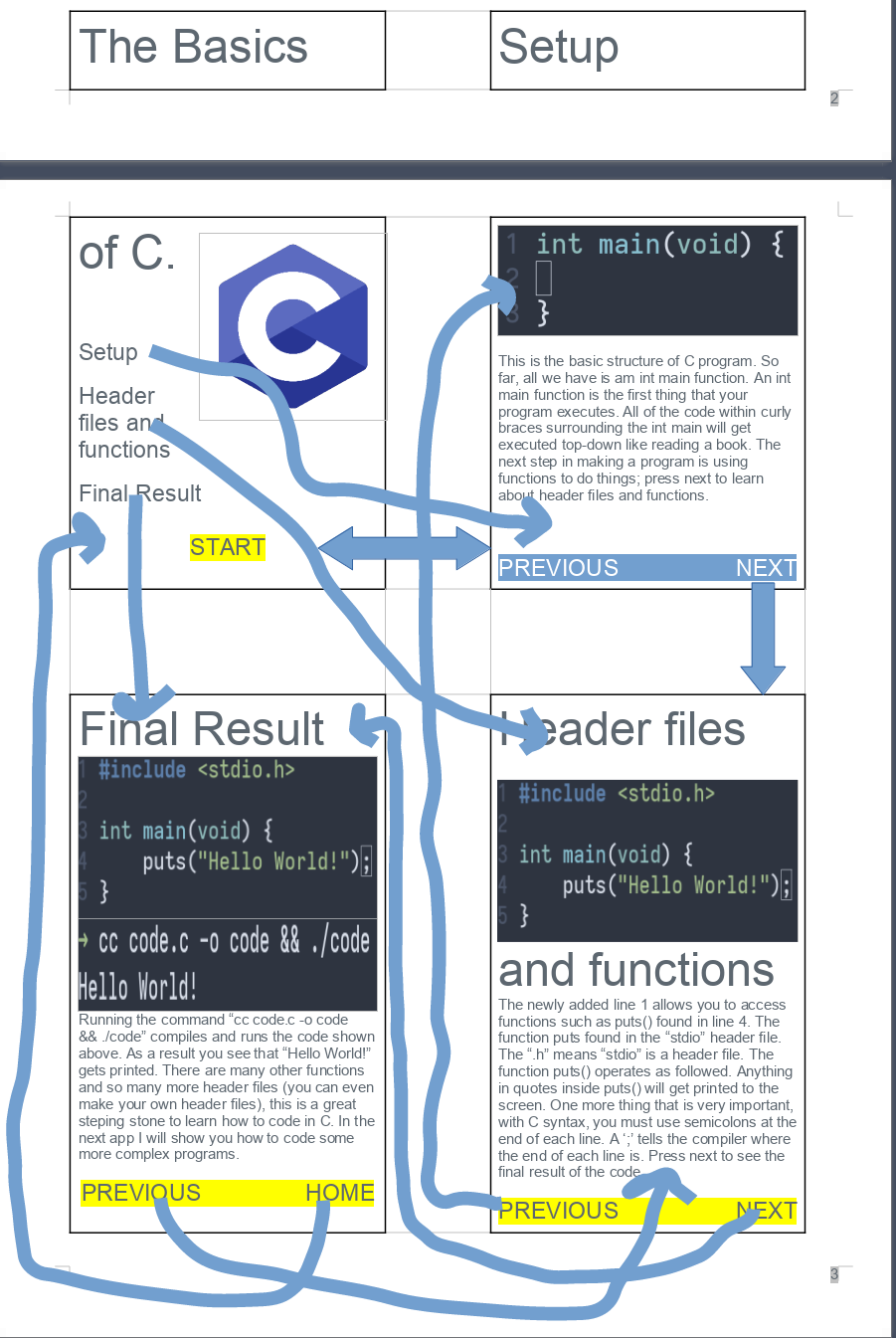
|  |  |  |
| --- | --- | --- |
| **Name** | **What do they already know about your topic?** | **What do they need or want to learn about your topic?** |
| Alan Sikarskie | My father already knows that C is a low level imperative programming langurage made by Dennis Ritchie. My father has had some experience coding in C. | My father would like to relearn the basic syntax structure and how to use functions inside the main function. |
| Amy Sikarskie | My mother does not know anything about the progtramming langurage C, except for the fact that I have been coding in it for a liittle bit. | My mother would like to lean for the first time how to print hello world and what header files are/ what header files are used for. |

My targeted audience is people who want to learn how to code in C. What I intended for them to do with the app is to learn the basics of C coding. I intend for the audience to learn how to make a “Hello World!” program coded in C. The audience would want to use the app because learning how to code will help you in life even if it's not your job. Knowing how to code will allow you to create basic scripts to do tasks that would otherwise take you hours to do.

## 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.





## Building Phase

**Step 5. Start Building Your App:** Work with your partner 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 pair programming
* 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* |
| “mainScreen” | “none” | This is the main screen of the app. Here are buttons to navigate the app. |
| “setupScreen” | “none” | This is the main screen of the app. Here are buttons to navigate the app. |
| “headerScreen” | “none” | This is the header screen of the app. Here is information about header files and functions. |
| “finalScreen” | “none” | This is the final screen of the app. Here is the final result of the C program with information about the future. |
| “cLogoImage” | “none” | Nothing changes if you press it, but it shows the C coding logo/symbol. |
| “mainTitle” | “none” | Nothing changes if you press it, but it shows the main title of the app. |
| “setupButton” | “click” | Once you click the button, it will send you to the app's setup screen/setup page. |
| “headerButton” | “click” | Once you click the button, it will send you to the header files and functions screen/setup page of the app. |
| “finalResultButton” | “click” | Once you click the button, it will send you to the app's final result screen/setup page. |
| “startButton” | “click” | Once you click the button, it will send you to the app's setup screen/setup page. |
| “templateCodeImage” | “none” | Nothing changes if you press it, but it shows a c code program's basic set up. |
| “setupInfo” | “none” | Nothing changes if you press it, but it shows information about C code's basic setup structure. |
| “setupBackButton” | “click” | Once you click the button, it will send you to the app's home/main screen/setup page. |
| “setupNextButton” | “click” | Once you click the button, it will send you to the header files and functions screen/setup page of the app. |
| “codeImageHeader” | “none” | Nothing changes if you press it, but it shows the C code discussed on the page. |
| “headerInfo” | “none” | Nothing changes if you press it, but it shows information about the basic structure of headers and C code functions. |
| “headerBackButton” | “click” | Once you click the button, it will send you to the app's setup screen/setup page. |
| “headerNextButton” | “click” | Once you click the button, it will send you to the app's final result screen/setup page. |
| “codeImageFinal” | “none” | Nothing changes if you press it, but it shows the C code discussed on the page. |
| “compileImage” | “none” | Nothing changes if you press it, but it shows the C code discussed being compiled and ran. |
| “headerInfo” | “none” | Nothing changes if you press it, but it shows information about the C program's final result with information about the future. |
| “finalBackButton” | “click” | Once you click the button, it will send you to the header files and functions screen/setup page of the app |
| “finalNextButton” | “click” | Once you click the button, it will send you to the app's home/main screen/setup page. |

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 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** |
| Amy Sikarskie | The buttons could have the same height, so it would allow clicking through the pages more accessible. | The text on the screen could be larger and the buttons could be bigger. |
| Alan Sikarskie | The images need to be smaller; they take up too much space and are causing the text to be too small. | The background of the screen and the buttons could have more fun colors. |

**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:

I will make the background and buttons more colorful.

Improvement 2 (Optional):

I will make the text on the screen larger and the images smaller.

**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)

The app that I created has an overall purpose of teaching the user how to code a "Hello World!" program written in C. The purpose of doing this is to have more people code in C and to give users a kickstart in their coding career. The app's functionality is straightforward; there are buttons, images, and text on the screen, which are informative. The app has four screens, one of which is the starting screen. A user input clicks from their mouse; these clicks then outputs a change in the screen. The difference in the screens is switching one screen to another. The four screens have the necessary information about C coding, and after following the steps in the app, one would be able to code a "Hello World!" program by themselves.

**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)

I used my mother and father for the testing of this app, Alan and Amy Sikarskie. Alan Sikarskie suggested that I should make the app more colorful. When he said that it made sense, the app at that point was very lackluster. I needed to make the app look less boring. I changed the color of the backgrounds and the color of the buttons on each slide. This change improved my app because now people might be more intrigued by the app's look to stick around and learn a thing or two about coding in C. My mother, Amy Sikarskie, suggested that I make the buttons have the same x and y coordinate, so flipping between pages would be more comfortable. I did what she said, and this also improved my app because it made it look more professional; it also allowed for easier use.

## 

## 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. |