



EUmember_ios development experience

How did you find Swift and iOS programming? What was difficult, what was easy, and how did you come up with your solution?

During my app development experience, I found Swift and iOS programming relatively straightforward and enjoyable. Certain aspects such as the use of `@State` and `@Binding` property wrappers were easy to grasp, while others like opening the Safari browser via a button and loading data from the web required some research and experimentation.

Country flags are available to download in a single package from <https://flagpedia.net/download/icons>, but I needed to change the name of the Greece flag image from `gr` to `el`. Another trick was to format the Wikipedia URL for countries like the United Kingdom using `\(country.name.replacingOccurrences(of: " ", with: "_"))`.

The process went quite smoothly since we already had the essential parts ready. To come up with my solution, I created the subviews `ButtonView` and `CountryListItem`. I ensured that they worked as intended before adding them to the parent views.

Brief summary of what I learned:

- The `@State` property wrapper is used to declare a value that can be modified within a view. This means that when the value of the variable changes, the view will be automatically re-rendered to reflect the new value.
- On the other hand, when a variable is declared with `@Binding`, it means that the value of the variable can be changed by another view. This creates a two-way connection between the two views, allowing them to share data.
- The `@EnvironmentObject` property wrapper is used to inject the `CountryManager` object into the view. This allows any view that needs to access or modify data related to countries to do so by using the `CountryManager` object.
- The position of a modifier is important.
- Ensure that all variables are initialized before they are used. We can add print statements or use a debugger to check the values of variables at runtime.
- Try commenting out parts of the code to isolate the issue. This can help us identify which part of the code is causing the error.

| Here is a demonstration of my first iOS app:



