iOS PREWORK

-[INTRODUCTION]-

Mihir Suvarna

Sophomore @ UT Austin
Majoring in CS, with focus on AI/ML

×

-[iOS PREWORK: TIP CALCULATOR APP]-

What's The Tip?

Designed to make finding a tip intuitive, and at the tip of your fingers. Created with an optional dark mode, tip slider, and locale-specific currency.

Big takeaway: why calculate a tip manually when an app can do it with one touch?

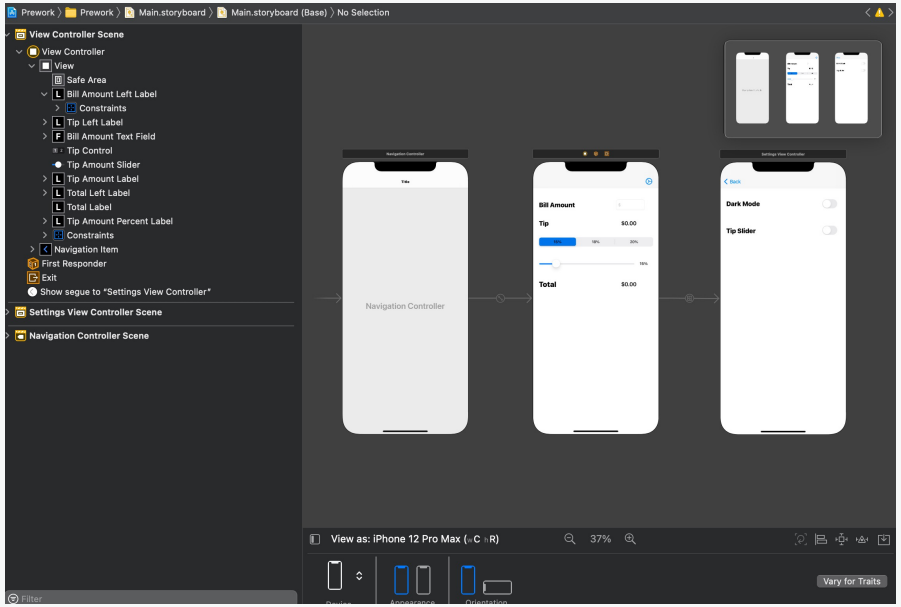
SETTING UP XCODE & LEARNING TO USE SWIFT

HOW I DID IT

The first step of my pre-work consisted of setting up Xcode and learning to use Swift.

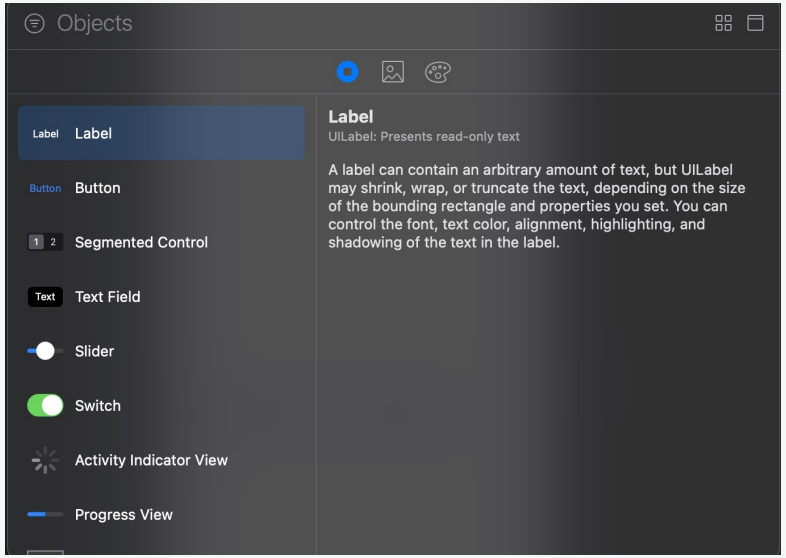
I needed to learn:

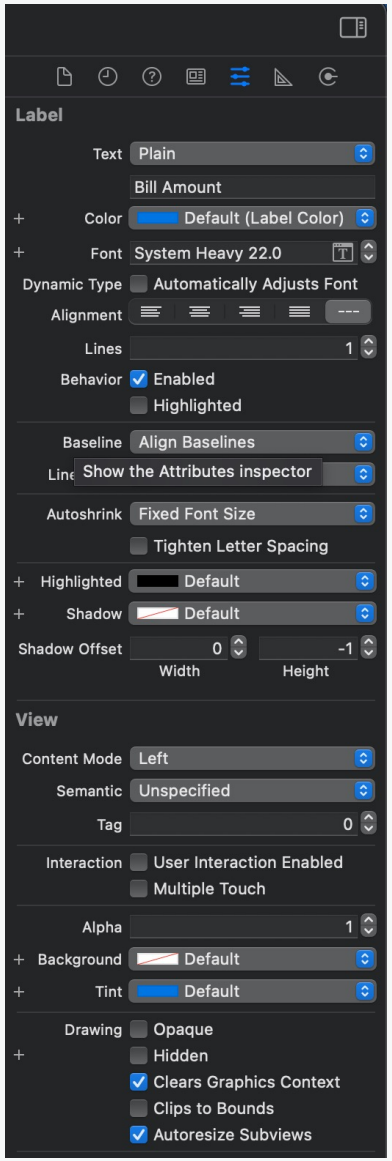
- Storyboard
- Connections to View Controller
- Outlets & Verifying Connections
- Functions, Structs



Basic UI aspects + optional features

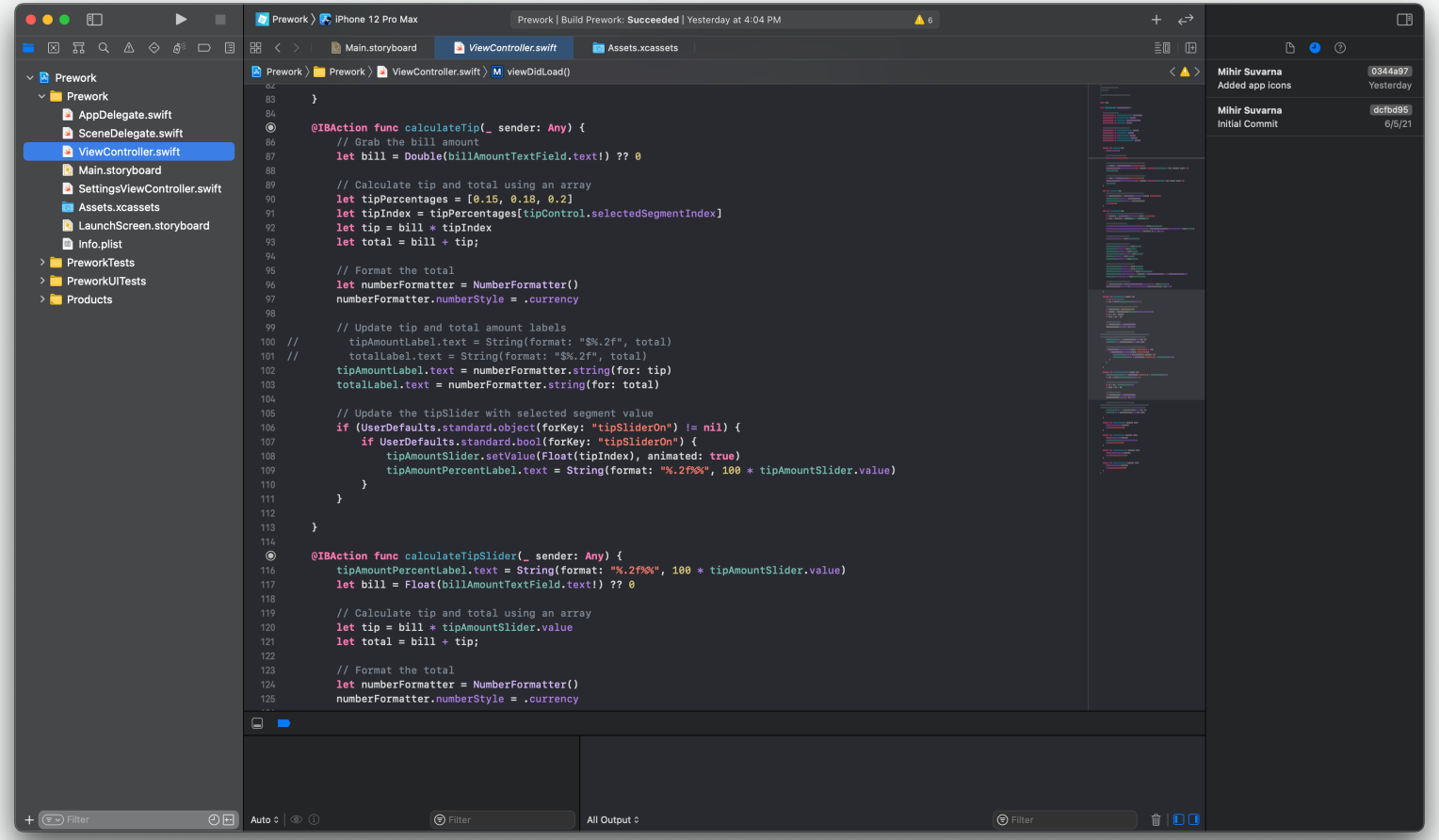
Used object library and attribute inspector.





- Text Field/Decimal Pad
- Responsive Labels
- Tip Slider and Seg. Control
- Dark Mode

View Controllers




```
    }

    @IBAction func calculateTip(_ sender: Any) {
        // Grab the bill amount
        let bill = Double(billAmountTextField.text!) ?? 0

        // Calculate tip and total using an array
        let tipPercentages = [0.15, 0.18, 0.2]
        let tipIndex = tipPercentages[tipControl.selectedSegmentIndex]
        let tip = bill * tipIndex
        let total = bill + tip;

        // Format the total
        let numberFormatter = NumberFormatter()
        numberFormatter.numberStyle = .currency

        // Update tip and total amount labels
//          tipAmountLabel.text = String(format: "$%.2f", total)
//          totalLabel.text = String(format: "$%.2f", total)
        tipAmountLabel.text = numberFormatter.string(for: tip)
        totalLabel.text = numberFormatter.string(for: total)

        // Update the tipSlider with selected segment value
        if (UserDefaults.standard.object(forKey: "tipSliderOn") != nil) {
            if UserDefaults.standard.bool(forKey: "tipSliderOn") {
                tipAmountSlider.setValue(Float(tipIndex), animated: true)
                tipAmountPercentLabel.text = String(format: "%.2f%%", 100 * tipAmountSlider.value)
            }
        }

    }

    @IBAction func calculateTipSlider(_ sender: Any) {
        tipAmountPercentLabel.text = String(format: "%.2f%%", 100 * tipAmountSlider.value)
        let bill = Float(billAmountTextField.text!) ?? 0

        // Calculate tip and total using an array
        let tip = bill * tipAmountSlider.value
        let total = bill + tip;

        // Format the total
        let numberFormatter = NumberFormatter()
        numberFormatter.numberStyle = .currency
```

Connecting UI to Code

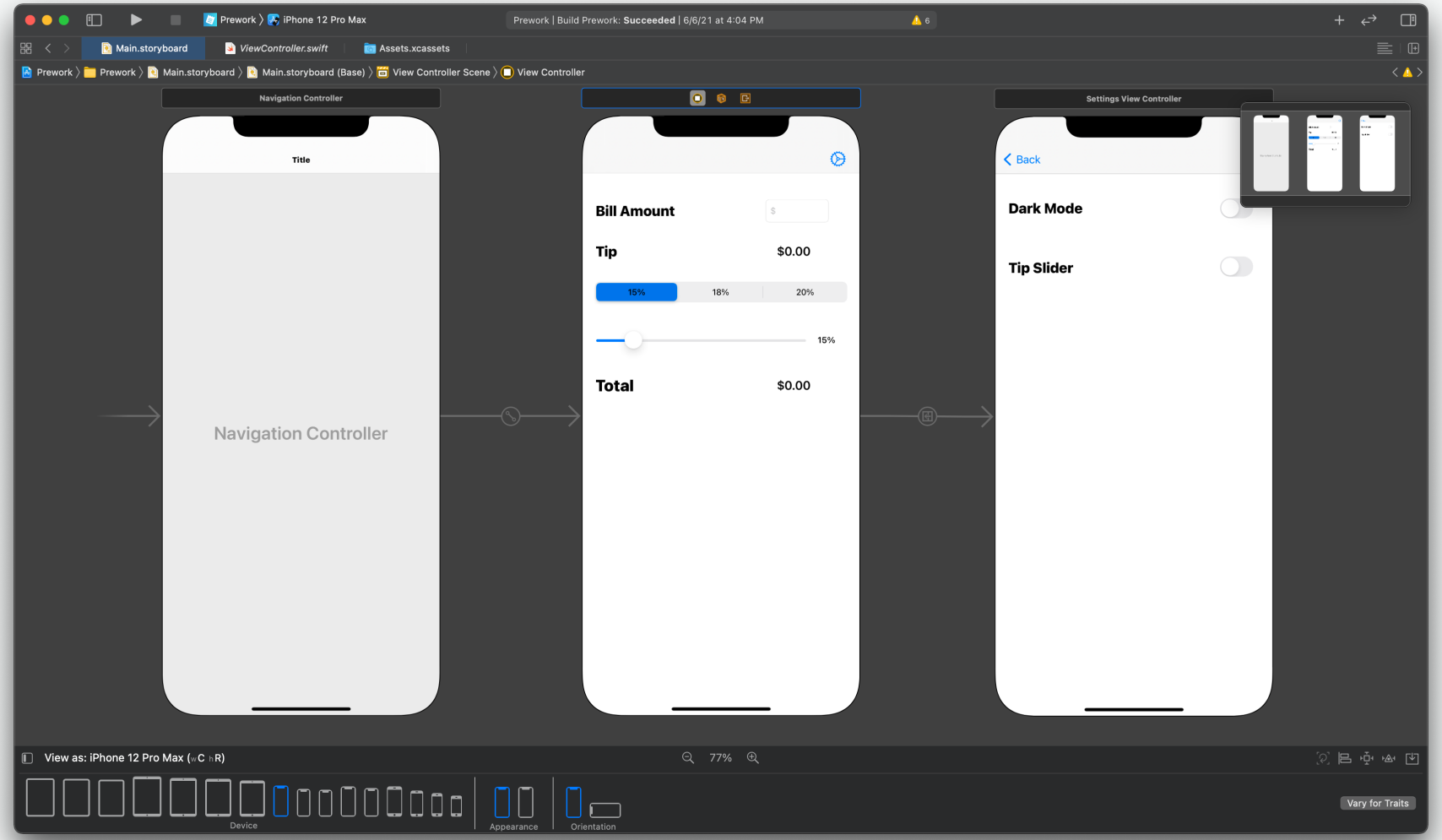
Once all objects are in place, I needed to use outlets and actions to make my UI functional.

Coded:

- Bill Text Field
- Tip/Total 'logic'
- Slider Functionality

Added:

- Clean Dark UI
- Currency Locale
- Settings w/ Switches



Navigation Controller

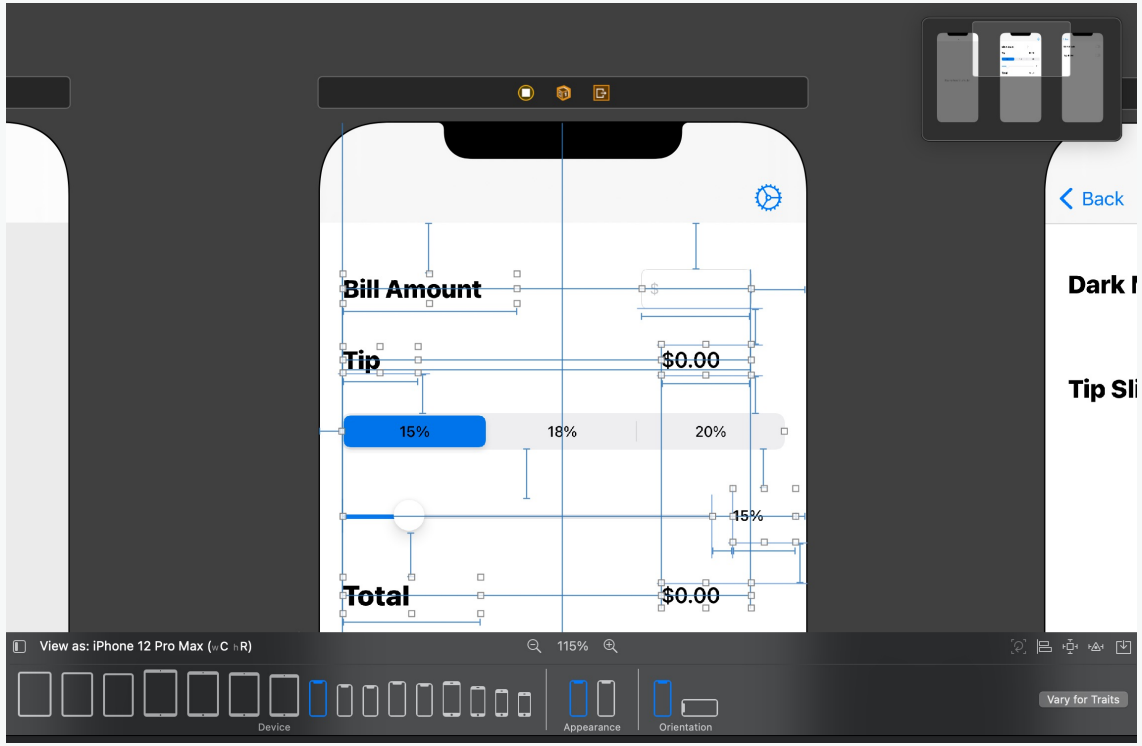
Simple top-bar navigation, settings icon included title and back button.

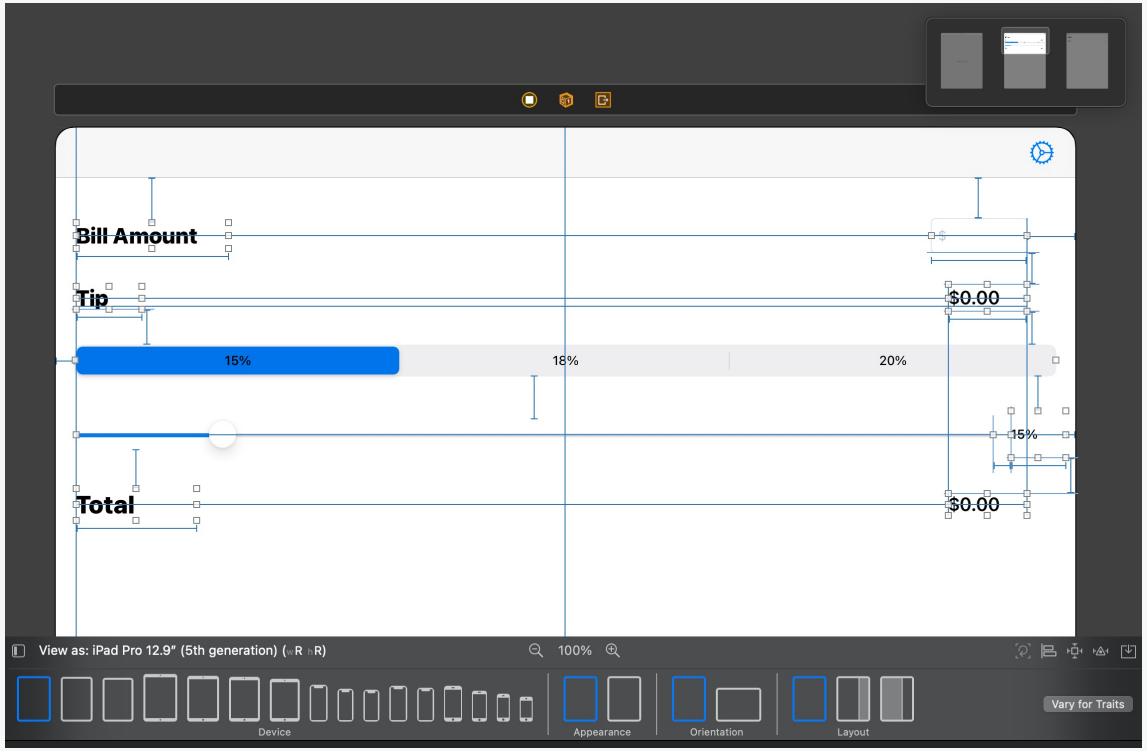
Tip View Controller

Main UI, responsive elements, first responder.

Settings View Controller

Optional settings screen, includes switches for tip slider & app-wide dark mode.





Constraints:
Love/Hate

Shown above are iOS constraints and iPadOS constraints for '*What's the Tip?*'.

Notice the # needed to ensure proper resizing across different devices; even more painful for watchOS & macOS.

With various screen sizes, constraints are a *must.* Adding them can be a pain.

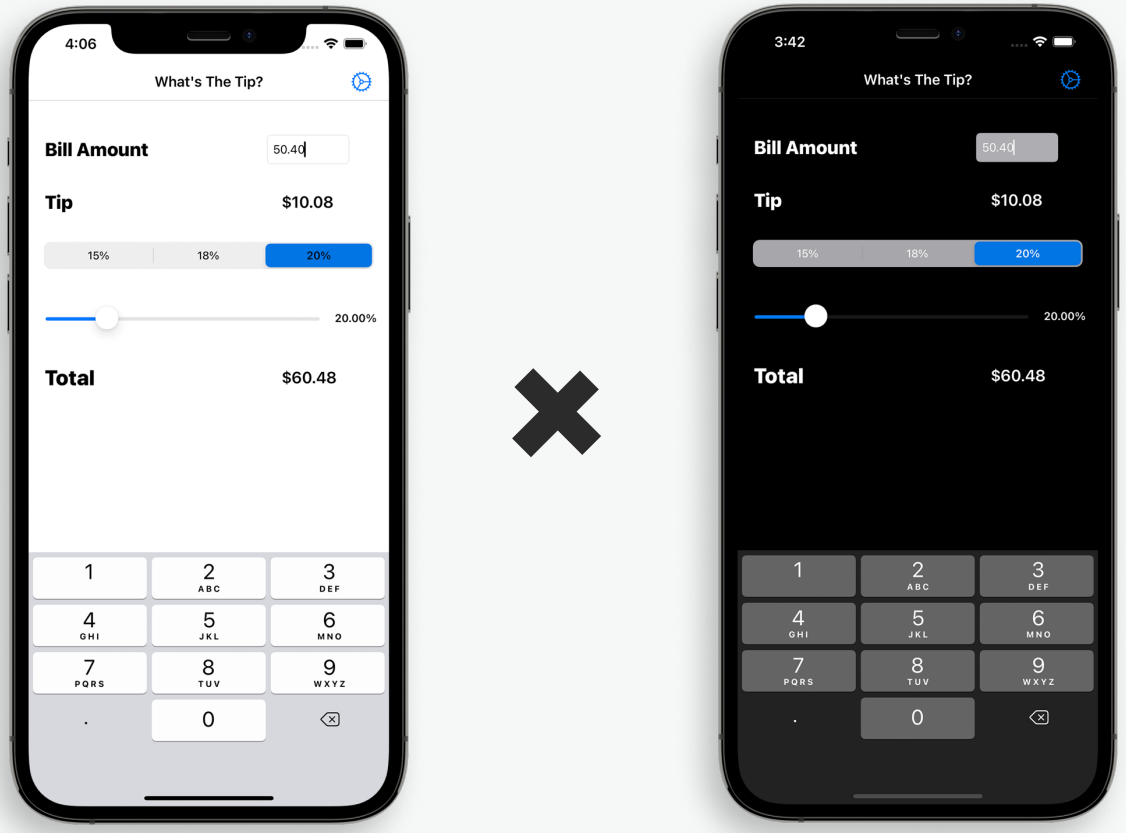
Xcode's auto-layout is not as reliable; manual adjustment is necessary.

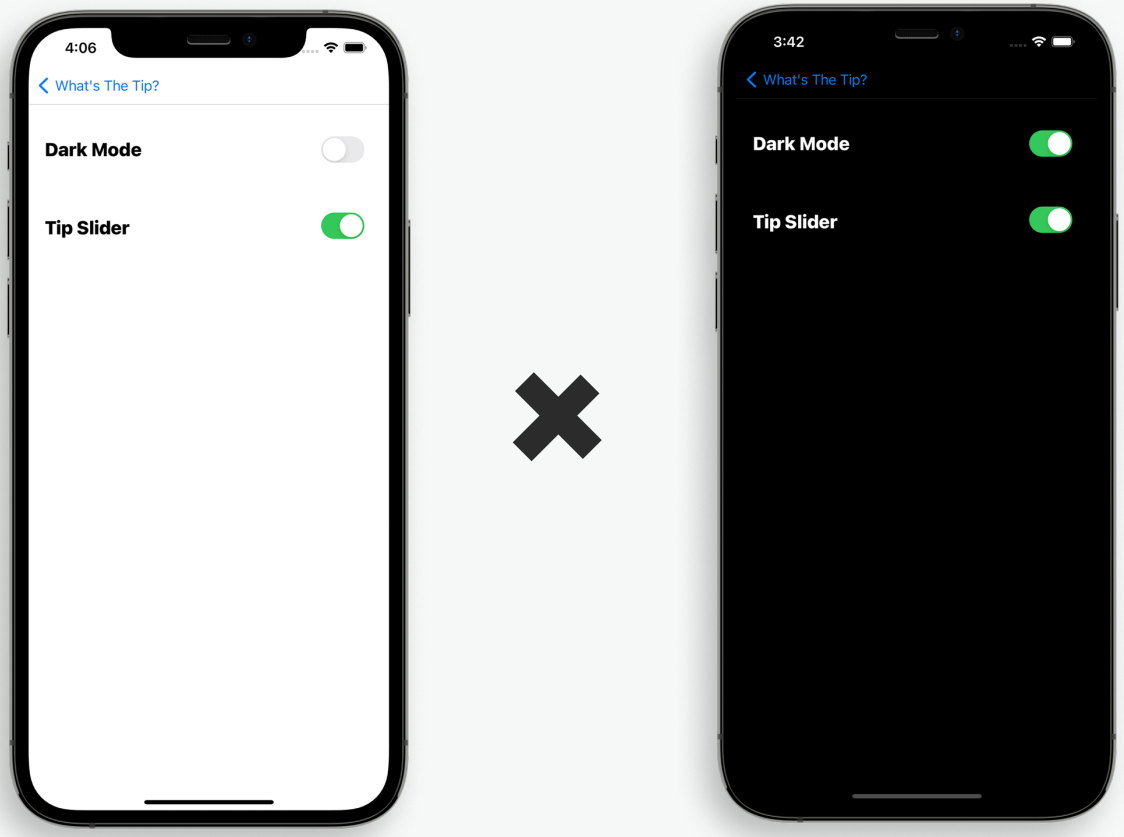
-[DARK MODE]-

Dark Mode: An Interesting Obstacle

App-wide dark mode was quite difficult.

Many UI elements individually customized, special colors needed from Apple's color set.





-[DARK MODE]-

Other View Controllers?

Had to add dark mode to settings view controller.

Think about scalability; can be *way* harder for more than just two view controllers.

Each UI has to be custom-designed; no easy Apple 'dark' switch exists.

iOS Preview

All UI elements in place, full product mockup.

Displayed on: iPhone 12 Pro Max

-[WHAT'S THE TIP?]-

iPadOS Preview

Added support for iPadOS; here is all UI elements in place, full mockup.

Displayed on: 11-inch iPad Pro 3rd Generation

Development Summary

1 hr

DESIGN THE UI

Create a flexible design, main view.

2.5 hrs

CONNECT AND CODE

Connect all elements and code a functional UI.

3 hrs

OPTIONAL FEATURES

Implement other features (dark mode, tip slider, locale).

1.5 hrs

PRODUCT REVIEW

Ensure product is complete and works on all devices.

THANK YOU.

QUESTIONS/COMMENTS?

WTT