

iOS Showcase

Peter Larson

Projects created in Xcode

Please do not redistribute my work!

TCA-Bible

- Multi-platform iOS 16.0/macOS 13.0 application built with SwiftUI and TCA.

<https://github.com/p-larson/TCA-Bible>

The screenshot shows the GitHub repository page for 'TCA-Bible' owned by 'p-larson'. The repository is public and has 46 commits. The code tab is selected. The repository description states it's a 'Bible reader built with SwiftUI & TCA'. It uses Swift and SwiftUI, and follows the Composable Architecture. The README.md file is present. The Languages section shows 100% Swift. Three mobile device screenshots at the bottom show the app's interface: a navigation bar with 'Books', a list of books like 'Genesis', 'Exodus', and 'Leviticus', and a detailed view of the 'Isaiah' book with chapters numbered 1 through 10.

TCA-Bible (Public)

Code Issues Pull requests Actions Projects Security Insights Settings

main · 2 branches · 0 tags

Go to file Add file <> Code

About

Bible reader built with SwiftUI & TCA

swift swiftui the-composable-architecture

Readme Activity 1 star 1 watching 0 forks

Languages

Swift 100.0%

TCA-Bible

Swift passing

This project is solely a demonstration of a multiplatform iOS / macOS application built with SwiftUI and TCA. Both apps are distinct, but they do share a majority of their code-base. Using TCA to hyper-modularize this project has allowed the two targets to share the most feasible amount of dependencies, which has been a huge advantage.

- iOS
- macOS
- Source
- Tests
- More

2:34 Books Genesis Exodus Leviticus

2:34 Books Isaiah

2:34 Genesis 1 In the beginning God created the heaven and the earth.

2 And the earth was without form, and void; and darkness was upon the face of the deep. And the Spirit of God moved upon the face of the waters.

3 And God said, Let there be light: and there was light.

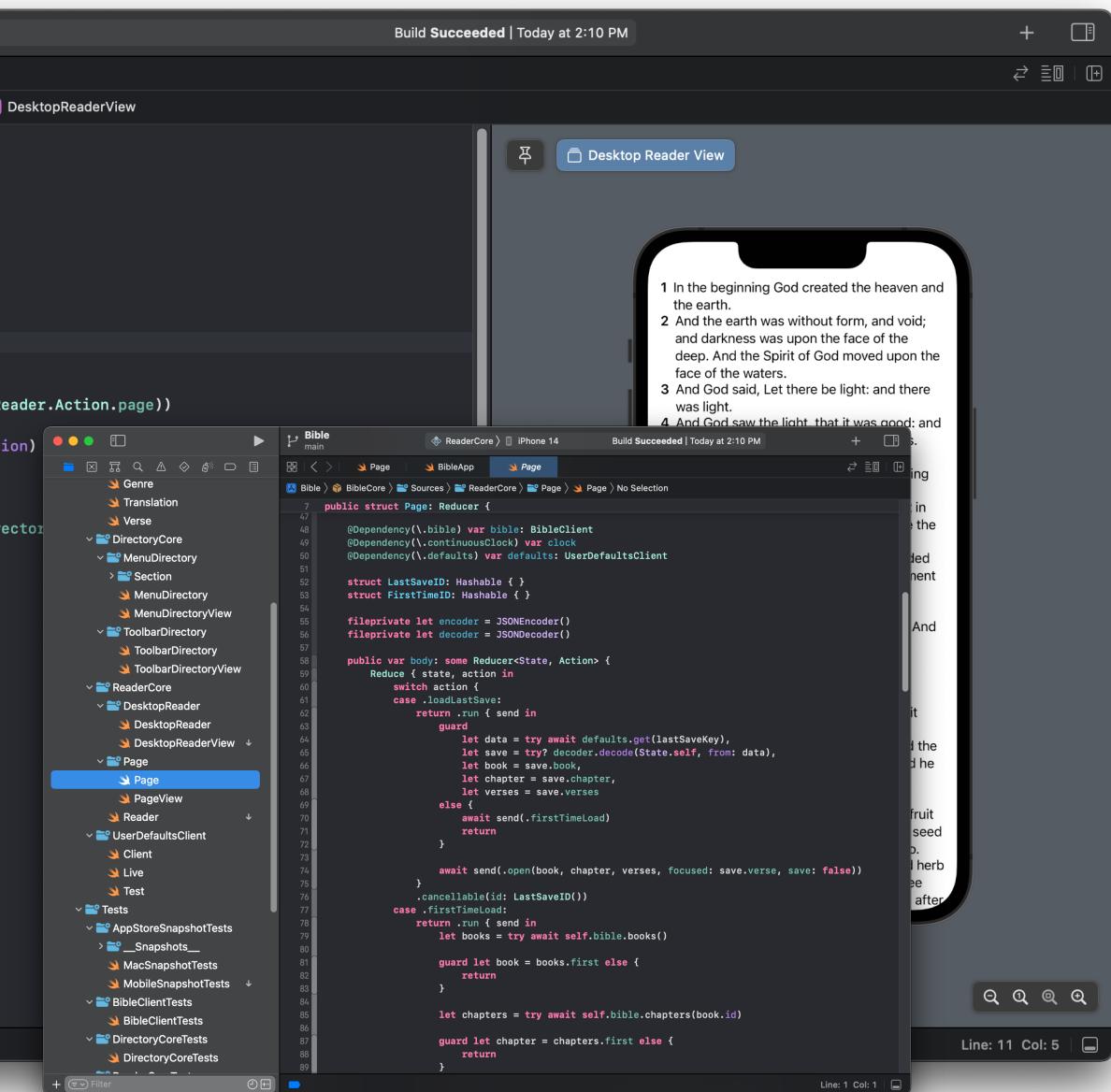
4 And God saw the light, that it was good: and God divided the light from the darkness.

5 And God called the light Day, and the

TCA-Bible

- Multi-platform iOS 16.0/macOS 13.0 application built with SwiftUI and TCA.

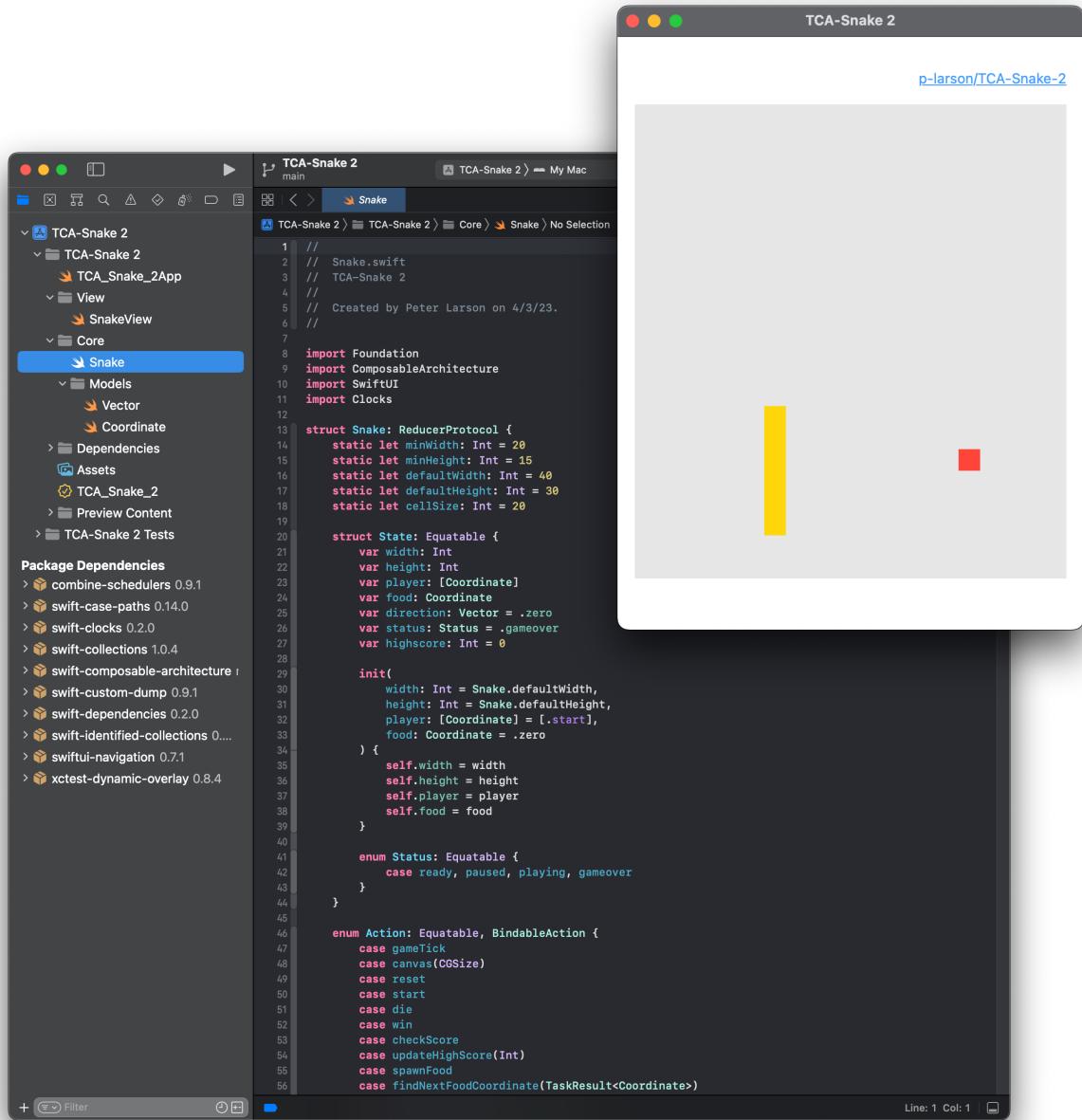
<https://github.com/p-larson/TCA-Bible>



TCA-Snake-2

A macOS interpretation of the popular game Snake, built with SwiftUI and TCA.

<https://github.com/p-larson/TCA-Snake-2>



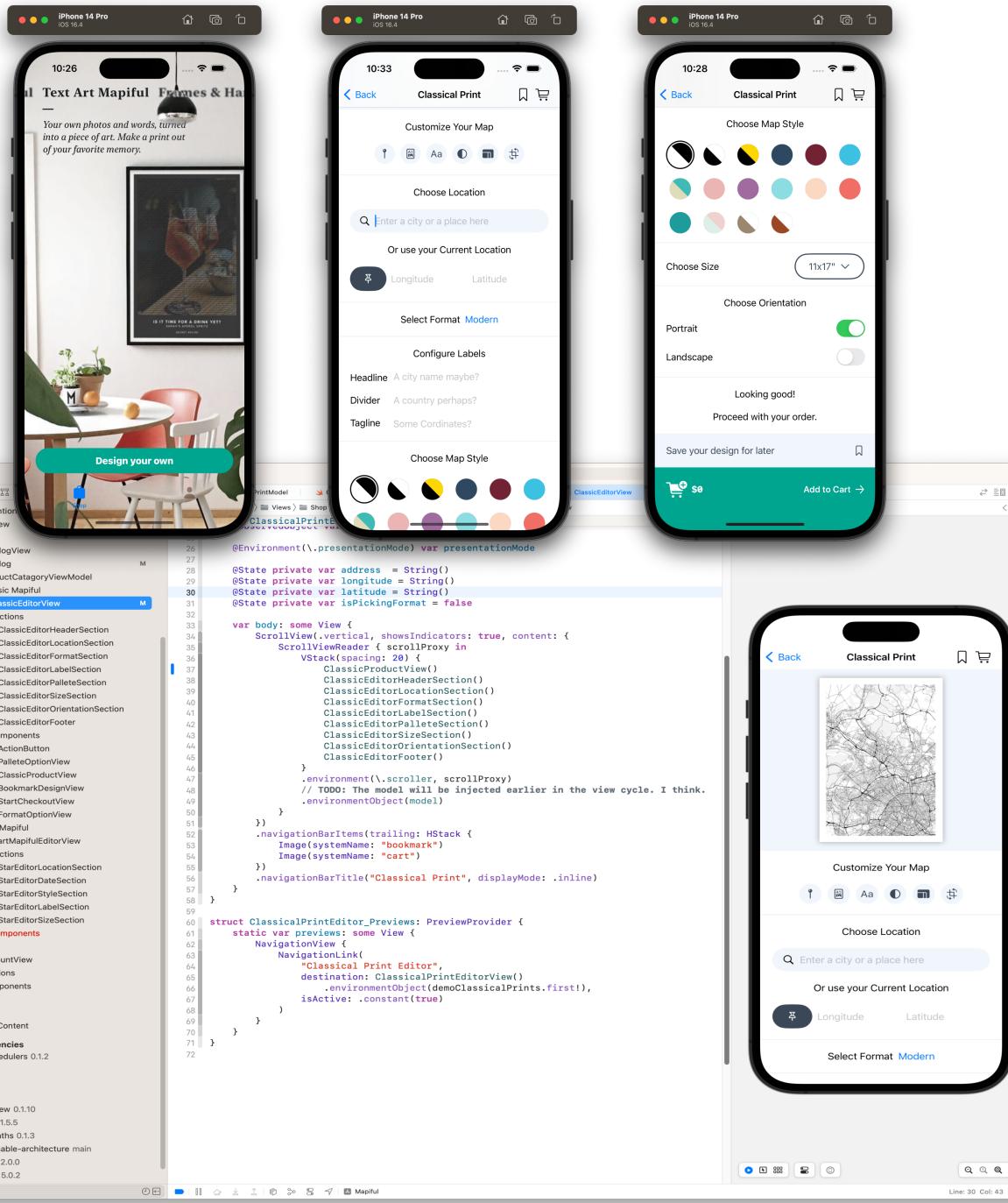
The image shows a screenshot of the Xcode IDE. On the left, the Project Navigator displays the project structure for 'TCA-Snake 2'. It includes a main target 'TCA-Snake 2' which contains subfolders 'View' and 'Core'. The 'Core' folder is expanded, showing a 'Snake' folder which is also expanded, revealing 'Models', 'Vector', and 'Coordinate'. Other sections include 'Dependencies', 'Assets', and 'Preview Content'. Below the project tree, the 'Package Dependencies' section lists various Swift packages. On the right, the Editor area shows the code for the 'Snake.swift' file. The code defines a 'Snake' struct with properties for width, height, player coordinates, food position, direction, status, and highscore. It includes an 'init' function to set initial values and an 'enum Status' with cases for ready, paused, and playing. The 'Action' enum handles game events like game tick, canvas resize, reset, start, die, win, check score, update high score, spawn food, and find next food coordinate. The code is annotated with comments explaining its purpose. To the right of the editor, a preview window titled 'TCA-Snake 2' shows a simple game interface with a yellow border and a red square representing the food or a segment of the snake.

```
1 //  
2 // Snake.swift  
3 // TCA-Snake 2  
4 //  
5 // Created by Peter Larson on 4/3/23.  
6 //  
7  
8 import Foundation  
9 import ComposableArchitecture  
10 import SwiftUI  
11 import Clocks  
12  
13 struct Snake: ReducerProtocol {  
14     static let minWidth: Int = 20  
15     static let minHeight: Int = 15  
16     static let defaultWidth: Int = 40  
17     static let defaultHeight: Int = 30  
18     static let cellSize: Int = 20  
19  
20     struct State: Equatable {  
21         var width: Int  
22         var height: Int  
23         var player: [Coordinate]  
24         var food: Coordinate  
25         var direction: Vector = .zero  
26         var status: Status = .gameover  
27         var highscore: Int = 0  
28     }  
29  
30     init(  
31         width: Int = Snake.defaultWidth,  
32         height: Int = Snake.defaultHeight,  
33         player: [Coordinate] = [.start],  
34         food: Coordinate = .zero  
35     ) {  
36         self.width = width  
37         self.height = height  
38         self.player = player  
39         self.food = food  
40     }  
41  
42     enum Status: Equatable {  
43         case ready, paused, playing, gameover  
44     }  
45  
46     enum Action: Equatable, BindableAction {  
47         case gameTick  
48         case canvas(CGSize)  
49         case reset  
50         case start  
51         case die  
52         case win  
53         case checkScore  
54         case updateHighScore(Int)  
55         case spawnFood  
56         case findNextFoodCoordinate(TaskResult<Coordinate>)  
57     }  
58 }
```

Mapiful Project

Turned a eCommerce webstore into a native iOS app using SwiftUI

<https://github.com/p-larson/Mapiful-iOS>



SwiftUICardStack

Open Sourced Interactable Data Picker SwiftUI View Swift Package for SwiftUI

<https://github.com/p-larson/SwiftUICardStack>

The screenshot shows the GitHub repository page for `p-larson/SwiftUICardStack`. The repository has 2 branches and 2 tags. The main branch contains commits from Jun 8, 2021, including initial commits for the package, example, and sources, as well as updates to LICENSE and README.md. The README.md file includes a screenshot of the SwiftUICardStack Example showing a stack of cards (Executive, Food, Gift, Travel) with names like LARSON and card numbers.

Contents

- Supported Platforms
- Package
- Example
- Extra

Supported Platforms

You can use the CardStack SwiftUI view in the following platforms:

- macOS 10.12+
- iOS 14.0+
- tvOS 14.0+

Package

For your XCode Package

File > Swift Packages > Add Package Dependency: <https://github.com/p-larson/SwiftUICardStack>

Select `main` branch then click `done`.

For Swift Packages

Add a dependency in your `Package.swift`

```
.package(url: "https://github.com/p-larson/SwiftUICardStack", branch: "main")
```

About

Create an easy to peek SwiftUI View to showcase your own data, catalog, images, or anything you'd like.

Tags

- pagination
- stack
- view
- animation
- cardview
- cardstack
- detail
- cardstackview
- swiftui
- swiftuicardstack

Readme

MIT license

17 stars

1 watching

1 fork

Releases

Initial Release (1.0.0) (Latest) on May 30, 2021

Languages

Swift 100.0%

Simple example

```
CardStack{  
    items: cards,  
    selection: $selectedCard,  
    builder: CardView.init(model:)  
}  
.sheet(item: $selectedCard) { card in  
    VStack {  
        Text(card.name)  
        Text(card.description)  
        Text(card.id)  
        Text(card.someMoreDetailedInformation())  
    }  
}
```

See Full Example Usage @ [SwiftUICardStack/Example](#)

Extra

My Favorite Line of Code from the Package ❤️

Line 17 of `Sources/SwiftUICardStack/CardStackSource.swift`

```
private(set) lazy var views: Array<AnyView> = items.map(builder).map(\.eraseToAnyView)
```

Why this is my favorite

I use the power of KeyPaths to type erase the data set in a short and sweet one liner!

See `View#eraseToAnyView`

License

Open Source [MIT License](#), aka: use how you please 😊

Snake Game

Swift, SwiftUI, TheComposableArchitecture

The screenshot shows the GitHub repository page for [p-larson/TCA-Snake](https://github.com/p-larson/TCA-Snake). The repository is public and has 12 commits. The README.md file contains instructions for running the app and links to the Composable Architecture documentation. A preview image of the game is shown on the right.

Code tab selected. Repository details:

- Owner: p-larson
- Name: TCA-Snake
- Description: Snake built on SwiftUI & TCA (The Composable Architecture)
- Branches: main (selected), 1 branch, 1 tag
- Commits: 12 commits (dc8c9cb on Jul 16, 2021)
- Topics: game, swift, ios, app, demo, tutorial, canvas, tca, example, beta, snake, swiftui, thecomposablearchitecture
- Readme: Readme
- License: MIT license
- Stars: 10 stars
- Watching: 1 watching
- Forks: 0 forks

Contents

- Supported Platforms
- Important Files

Supported Platforms

- iOS 15.0+

Important Files

- Sources/GameState.swift
- Sources/GameReducer.swift
- Sources/GameView.swift

License

Open Source [MIT License](#), aka: use how you please 😊

About

Snake built on SwiftUI & TCA (The Composable Architecture)

game, swift, ios, app, demo, tutorial, canvas, tca, example, beta, snake, swiftui, thecomposablearchitecture

Readme, MIT license, 10 stars, 1 watching, 0 forks

Releases

1 tags, Create a new release

Packages

No packages published, Publish your first package

Languages

Swift 100.0%

© 2023 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About

iOS Showcase - Peter Larson

Larson 8

TCA-Snake-2

macOS 13.3 application of the retro videogame Snake built in SwiftUI with TCA

<https://github.com/p-larson/TCA-Snake-2>

About

macOS Snake game, SwiftUI & TCA

Releases 1

Packages

Languages

Architecture

Snake.swift

- State
- Actions
- Reducer

StatefulTabView

Contributed to a SwiftUI Swift Package for a state retaining TabView.

<https://github.com/NicholasBellucci/StatefulTabView>

The screenshot shows the GitHub repository page for `NicholasBellucci / StatefulTabView`. The repository has 56 commits, 2 branches, and 11 tags. It includes sections for About, Releases, Packages, Contributors, and Languages. The `About` section describes it as a SwiftUI TabView that retains the state of each tab. The `Releases` section shows v0.1.0 (Latest) from Oct 31, 2022. The `Packages` section indicates no packages published. The `Contributors` section lists 5 contributors, with one profile circled in red and labeled "Me!". The `Languages` section shows Swift at 100.0%. The repository page also features sections for Code, Issues, Pull requests, Actions, Projects, Security, and Insights.

About

A SwiftUI TabView that retains the state of each tab as well as some other goodies.

Code

master · 2 branches · 11 tags

Issues

Pull requests

Actions

Projects

Security

Insights

Code

Go to file Add file About

Commits

NicholasBellucci Merge pull request #16 from camdenfullmer/return-if... · 9faa432 on Oct 31, 2022 · 56 commits

· swiftpm/xcode/package.xcworkspace First commit 3 years ago

· Example Reverted selected index binding change 3 years ago

· Sources/StatefulTabView Return early if scroll view was found. 7 months ago

· Tests Added more customization modifiers 3 years ago

· .gitignore First commit 3 years ago

· LICENSE Create LICENSE 3 years ago

· Package.swift First commit 3 years ago

· README.md Reverted selected index binding change 3 years ago

README.md

StatefulTabView

A SwiftUI UITabBarController implementation that retains state between tab changes. Big thanks to [Amzd](#) and everyone who helped to refine this [gist](#) as it was a major jumping off point for setting up this project.

Requirements

- iOS 13.0+
- Xcode 11.2+
- Swift 5+

Installation

Swift Package Manager

In Xcode 11 or greater, navigate to File > Swift Packages > Add Package Dependency.... From there just simply add <https://github.com/NicholasBellucci/StatefulTabView> as the package repository url and use the master branch or the most recent version. Master will always be inline with the newest release.

Table of Contents

- Features
- Usage
 - Basic
 - Appearance Modifications
 - Selected Index
 - Badge Value
 - Scroll to Top with Large Titles
- License

Features

- ✓ State driven selected index
- ✓ TabBar appearance configuration
- ✓ TabBar custom tint color
- ✓ TabBar custom background color
- ✓ TabBarItem custom title and image
- ✓ TabBarItem badge value

Releases

v0.1.0 (Latest)
on Oct 31, 2022

+ 10 releases

Packages

No packages published

Contributors 5

Me!

Languages

Swift 100.0%

SwiftUISelector

Created an animated SwiftUI top navigation tab selector

<https://github.com/NicholasBellucci/StatefulTabView>

The image shows two side-by-side screenshots of the GitHub repository for `SwiftUISelector`. The left screenshot displays the repository's main page with the README file open, showing a green rectangular placeholder image for the SwiftUI Selector component. The right screenshot shows the same repository page with the README file open, but the green placeholder image has been replaced by a larger image showing the SwiftUI Selector component in its active state, with a vertical gradient bar transitioning from green to yellow.

Repository Details:

- Owner:** p-larson
- Name:** SwiftUISelector
- Language:** Swift
- Stars:** 4
- Watchers:** 1
- Forks:** 0
- Issues:** 1
- Commits:** 17
- Last Commit:** Nov 2, 2020

README.md Content (Left Screenshot):

```
A perfect option picker. Great for navigation. Simple to use.  
Try it out with @fermoya's package SwiftUIPager!  
Swift Package Manager  
SwiftUISelector is available through Swift Package Manager.  
Add it to an existing Xcode project as a package dependency:  
1. From the File menu, select Swift Packages > Add Package Dependency...  
2. Enter "https://github.com/p-larson/SwiftUISelector" into the package repository URL text field  
or  
https://github.com/p-larson/SwiftUISelector
```

README.md Content (Right Screenshot):

```
A perfect option picker. Great for navigation. Simple to use.  
Zodiac Text Art
```