



Peng

Engineers are the artists of our generation.

Sep 6 · 3 min read

Building App with VIP Part 1: Prerequisite



Photo by John Schnobrich on Unsplash

Today, we have more choices on architecture of iOS app including Apple's MVC, MVVM, MVP, VIPER, VIP, etc. The decision depends on different contexts. So we will start a new series to explain them one by one with practical examples.

VIP(Clean Swift iOS Architecture) is a newcomer which means View Controller, Interactor and Presenter. But there are more components

than these three. I will build a simple app step by step while explaining different roles in VIP including what they are and how they work together.

Prerequisite

In this example, it will display our starred repositories in GitHub. Please start a new project with single view template in Xcode. Before start coding, please install [octokit.swift](#) as our networking layer. [octokit.swift](#) is

| *A Swift API Client for GitHub and GitHub Enterprise*

Here I will use [CocoaPods](#).

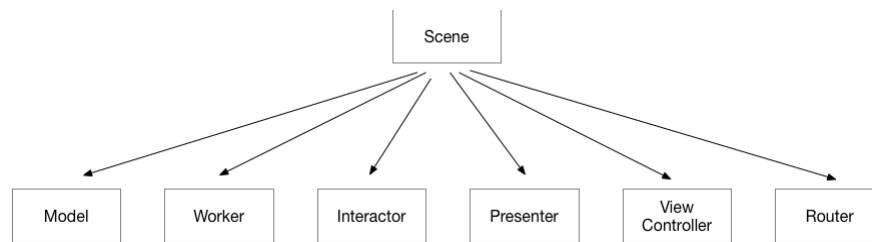
```
1 platform :ios, '9.0'
2
3 target 'VIP101' do
4     use_frameworks!
5
6     pod 'OctoKit.swift'
```

Then please generate a new personal access tokens for our example app in [tokens page](#) of GitHub's Settings. Finally, put this token config in new AppConfig.swift

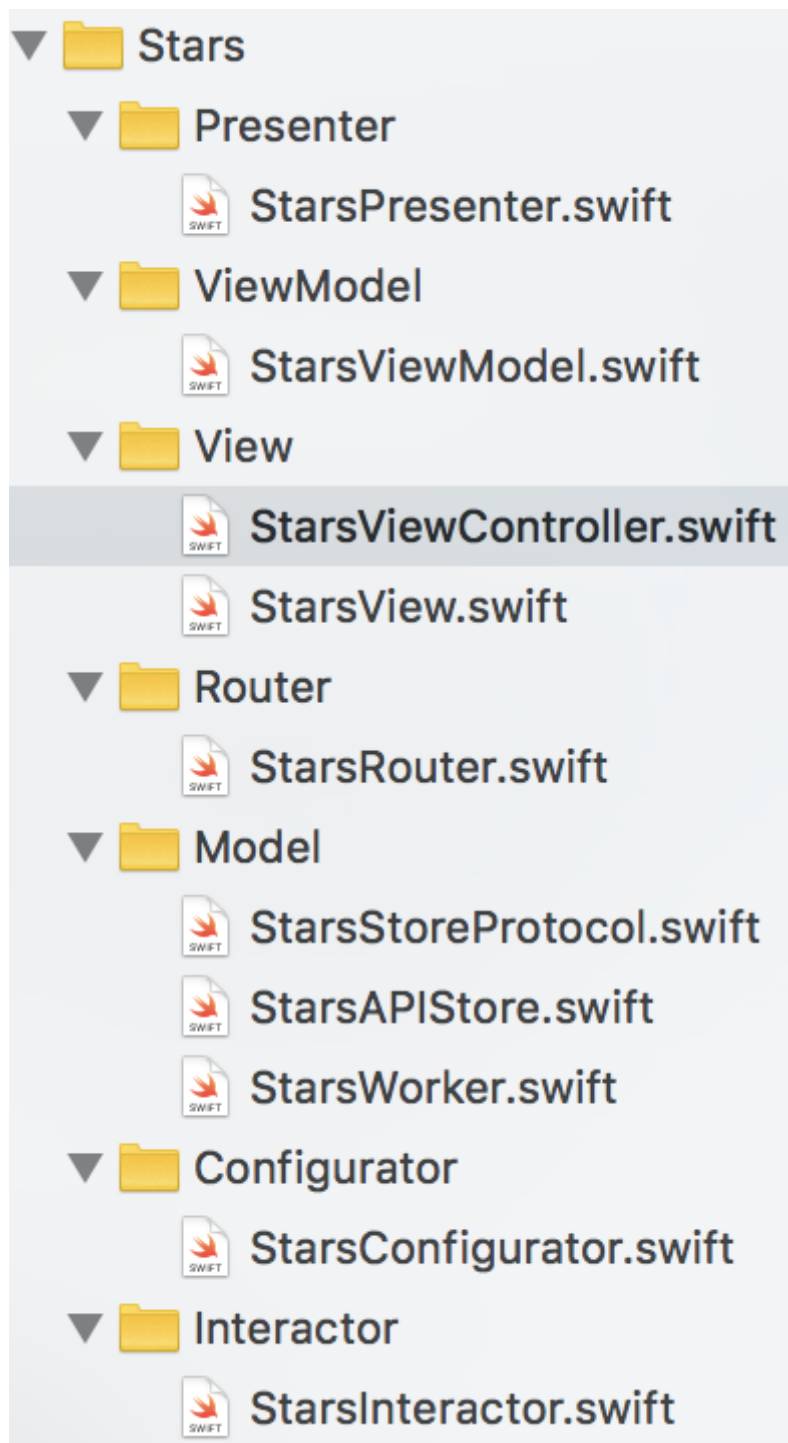
```
1
2 let config = TokenConfiguration("xxx")
```

Scene

In VIP, a scene can be considered as a complete functional “view”. As in the diagram, the scene includes its own models, worker, *interactor*, presenter and router except UI components.



Don't mind if you don't know what they are now. We will explain them one by one with their responsibilities and relationships. For simplicity, in our example there will be only one scene to show list of our starred repositories. The screenshot below gives us a complete view on what this scene looks like physically.



Xcode Template

Don't worry on so many folders. [[VIP Xcode Templates](#)] is a great tool to create the folder structure for you automatically. Please install this template and create new file.

ustwo VIP Templates



Scene

Input the scene name.

New Scene Name:

View Controller Name: StarsViewController

Interactor Name: StarsInteractor

Presenter Name: StarsPresenter

Router Name: StarsRouter

Configurator Name: StarsConfigurator

View Model File Name: StarsViewModel

Worker File Name: StarsWorker

That's all for our first part for VIP. In next article, we will introduce different components of VIP from bottom-up.

Subscribe

Subscribe my weekly to see what happened in world of Swift.

. . .

Thanks for your time. Please **clap** to get this article seen by more people. Please click **Follow** to get latest blogs from me. As a *passionate iOS developer, blogger and open source contributor*, I'm also active on [Twitter](#) and [GitHub](#).

