SolarDashboard

The Architecture
Environment Graph

AppDelegate \Rightarrow \text{ObservableObject}
Environment Graph

AppDelegate

⇒ ObservableObject

AppEnvironment
Environment Graph
⇒ ObservableObject

- AppDelegate
- AppEnvironment
  - SolarPowerEnv
  - TeslaEnv
  - AccessoryEnv
  - LoginEnv
AppEnvironment subscribes to children environments

Environment Graph

AppDelegate

AppEnvironment

SolarPowerEnv
  API

TeslaEnv
  TeslaAPI

AccessoryEnv
  HomeKit

LoginEnv
  API
Routing Graph

AppEnvironment
Routing Graph

AppEnvironment

RootRouter

SolarEdgeRouter
  - Login
  - Forget Password

AccessoryRouter
  - Add HomeKit Device
  - TeslaRouter
Add new accessory (Tesla)

RootRouter → TeslaRouter

RootViewController

AccessoryEnv

AccessoryBrowser View
Add new accessory (Tesla)

Tap “Add Tesla”
Add new accessory (Tesla)

RootRouter → TeslaRouter

RootViewController

AccessoryEnv

AccessoryBrowser

View

LoginTeslaView

TeslaEnv

Tap “Add Tesla”
Add new accessory (Tesla)

Tap “Add Tesla”
Add new accessory (Tesla)

Tap "Add Tesla"
Architecture Recap

- Each SwiftUI View has an environment

- Environments are structured according to features (i.e. TeslaEnv)

- Update of the environment will propagate an update of the Views

- Views always represent the correct state since they have a binding to the environment

- AppEnvironment (parent) subscribes to children environments. It would be possible to update all Views when changing any environment in the hierarchy

- Environments hold a reference to a router

- Router handle the navigation within the app
Thanks

SolarDashboard: https://apps.apple.com/de/app/solardashboard/id1446793167

Christian Menschel & Manuel Carrasco Molina “Stuff mc” 27.2.2020 CocoaHeads Aachen

Green power!