iOS App Architectures and Design Patterns

Mauro Arantes

Adapter Design Pattern

The Adapter Design Pattern converts the interface of a class to another interface which the client expects to find, translating user requests to a compatible format with the adaptee class and redirecting them. This way, the adapter allows classes with incompatible interfaces to work together.

The Target defines the interface of a specific domain the Client utilizes, the Adapter adapts the Adaptee interface to a Target class interface, the Adaptee defines an existing interface that needs to be adapted and the Client collaborates with objects in conformity with the Target interface.

MVVM Architecture

The MVVM (Model-View-ViewModel) is an architecture created in 2005, which works through separate layers. The Model layer doesn't know the View layer and vice-versa, in fact, View only knows the ViewModel layer and communicates with it through the binding mechanism, notifying events occurrences and commands triggers. The ViewModel responds to this notification by performing certain actions in the model, either getting data, updating or adding information in the Model.

The View's responsibility is to create the UI. It connects to the ViewModel through the property DataContext that is set for the class ViewModel corresponding to the View.

The Model encapsulates the data logic and it doesn't reference the View nor ViewModel directly. The Model is responsible for data validation and error reports and it's often utilized as a repository or service.

The MVVM allows a clear separation between the interface (View), its presentation logic (ViewModel) and the data (Model). In such way, there is responsibilities separation and decoupling, making it easier to scale and maintain an application.