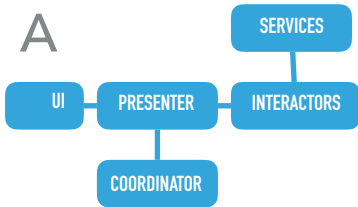


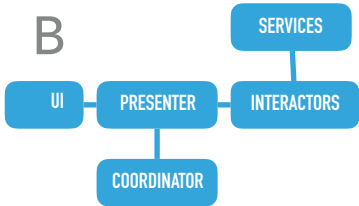
DEPENDENCY INJECTION, MODULES



A



B



C

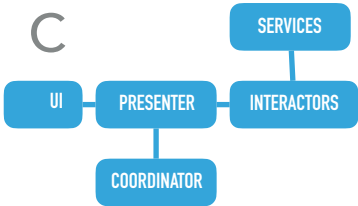
SERVICES

UI

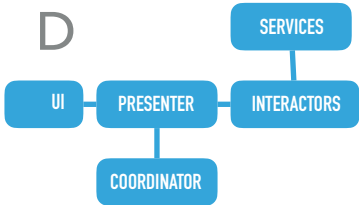
PRESENTER

INTERACTORS

COORDINATOR



D



E

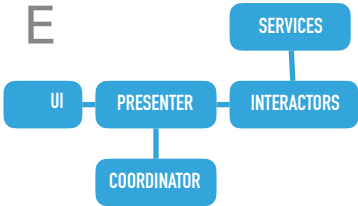
SERVICES

UI

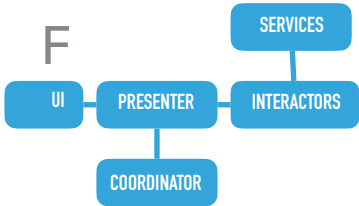
PRESENTER

INTERACTORS

COORDINATOR



F

























































































































































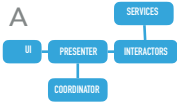




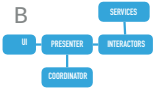


Let's create a class that is responsible (**on demand - very important**) for initialising all the feature internals, creating dependencies, etc.

A



B



C

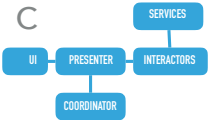
SERVICES

UI

PRESENTER

INTERACTORS

COORDINATOR



D

SERVICES

UI

PRESENTER

INTERACTORS

COORDINATOR



E

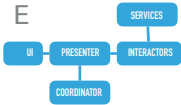
SERVICES

UI

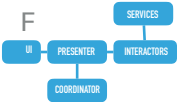
PRESENTER

INTERACTORS

COORDINATOR



F



MODULES

DEPENDENCY INJECTION,





















public protocol Application Modules {

func addd > / module: T

func modvle>T

fundamental values \Rightarrow [T]

Let's create a class that is responsible (and very important) for initialising all the features

internals, creating dependencies, etc.

A



