

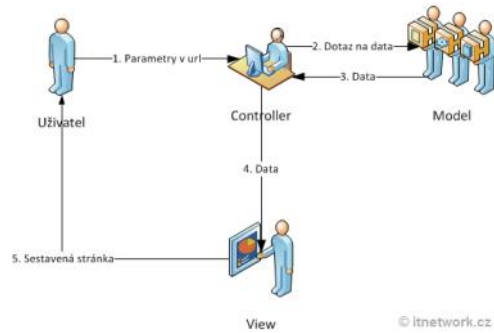
19 Frameworky architektury MVC

Wednesday, 19 January 2022 09:04

MVC, MVP, MVVM, Mode, View, Controller, Presenter, Page, PageModel, request, response, http

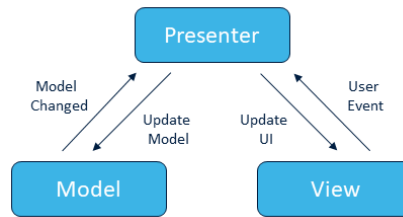
1. MVC (Model-View-Controller)

- populární na webu
- oddělení logiky od vstupu
- např. ASP.NET MVC, Ruby on Rails, Django
- a. Model
 - data a logika aplikace
- b. View
 - zobrazení výstupu uživateli
 - měl by obsahovat minimum logiky
- c. Controller
 - propojení Modelu a View
 - příjem vstupu od uživatele a převod na příkazy pro Model
 - metody, které se dají volat skrze REST nebo routování, se nazývají akce
 - View může mít více controllerů
- o PageModel
 - modifikace MVC
 - spojení Modelu a Controlleru
 - např. v ASP.NET
 - i. Model
 - data a stav aplikace
 - ii. Page (.cshtml)
 - iii. PageModel (.cshtml.cs)



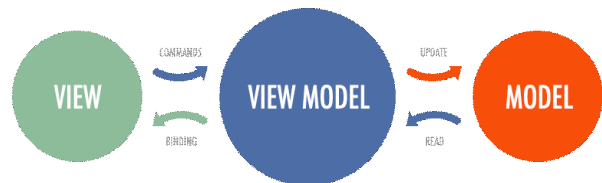
2. MVP (Model-View-Presenter)

- stejně jako MVC
- skoro stejné jako PageModel
- a. Model
 - při změně modelu předá informaci Presenterovi a ten ji předá View
- b. View
 - formátuje se podle dat získaných z Presenteru
 - každý View má jednoho Presentera
 - při uživatelské interakci předává info Presenterovi a ten ji předá Modelu
- c. Presenter
 - více vázaný na View než na Model, oproti MVC



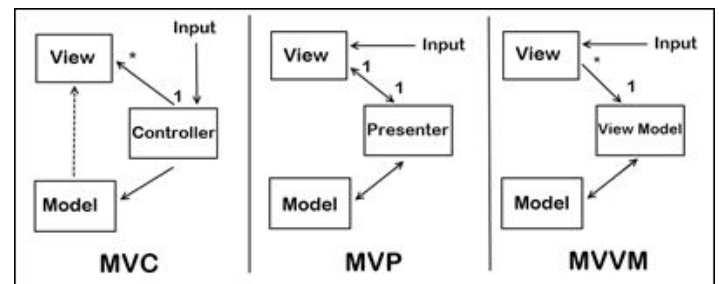
3. MVVM (Model-View-ViewModel)

- mimo .NET znám taky jako MVB (Model-View-Binder)
- SW architektura (návrhový vzor)
- a. Model
 - logika aplikace a data
 - neví nic o stavu ovládacích prvků
- b. View
 - UI
- c. ViewModel
 - binding
 - synchronizace dat mezi Modelem a ViewModelem
 - prostředky:
 - a) Command
 - objekt reprezentující událost
 - vnitřní delegát zavolán při eventu ve View
 - taky nabitovaný s vlastností ovládacích prvků
 - b) Converter
 - metoda modifikující data (přetypování)
 - např. bool → enum Visibility
 - c) Observer
 - závislost 1:0..N mezi poskytovatelem a konzumenty
 - při změně poskytovatele jsou upozorněni všichni konzumenti
 - ti mohou na změnu reagovat

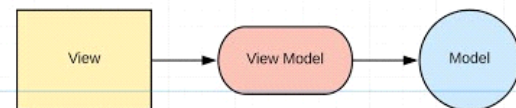


• klient-server aplikace

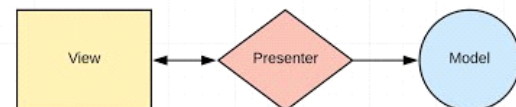
- aplikace běží na serveru a zobrazuje si ji klient
- komunikace protokolem HTTP
- request
 - klient žádá o data
- response
 - server vrací data
 - i. 1xx Info
 - ii. 2xx Success
 - iii. 3xx Redirect
 - iv. 4xx Client Error
 - v. 5xx Server Error
- REST (Representational State Transfer)
 - URL reprezentuje přímý přístup k datům
 - metody:
 - GET
 - získání dat
 - POST
 - vytvoření dat
 - DELETE
 - vymazání dat
 - PUT
 - aktualizace celé entity
 - PATCH
 - aktualizace části entity
- MPA (Multi Page Application)
 - aplikace s více stránkami běží na serveru
 - tradiční přístup
 - pro robustní aplikace
- SPA (Single Page Application)
 - aplikace s jednou stránkou běží na klientu
 - rendering v rámci jedné stránky
 - moderní přístup



MVVM :



MVP :



MVC :

