CodeIgniter

CodeIgniter je open-soursový PHP framework využívaný pre rýchly vývoj dynamických webových aplikácií. Je založený na MVC (model-view-controller) návrhovom vzore, avšak pre prácu s ním je povinné využívať iba controllery zatiaľ čo dátové modely a view (pohľady) zostávajú nepovinnými. Pred ostatnými frameworkami vyniká najme vďaka svojej rýchlosti a malým nárokom. Jeho prvá verzia bola zverejnená 28. 2. 2006 spoločnosťou ElliseLab. Aktuálne je dostupný vo verzii 3.1.4 .

Architektúra aplikácie

Aplikácia je vytvorená za pomoci frameworku Symfony takže využíva MVC (model-view-controller) návrhový vzor. Tento návrhový vzor rozdeľuje aplikáciu na 3 vzájomne prepojené vrstvy.

Prvou vrstvou je dátový model (model) ktorý predstavuje jednotlivé objekty ktoré sú abstrakciou reality a prácu s nimi. Tento model následne definuje formát dát, logiku a pravidlá pre prácu s nimi.

Tzv. pohľad (view) reprezentuje vrchnú vrstvu aplikácie ktorá je viditeľná užívateľovi. Jeho hlavnou úlohou je zobrazovanie dát. Tieto dáta sú užívateľovi zasielané formou odpovedi (response). Pre účel zobrazenia dát využíva zväčša značkovací jazyk HTML doplnený niektorým zo šablónovacích nástrojov (template engines) ktorým je v našom prípade twig. Ten nám dovoľuje volať predom definované či nami vytvorené makrá priamo v HTML kóde a vďaka tomu umožňuje zobrazovanie rovnakých dát v rôznej forme.

Poslednú vrstvu tvorí tzv. radič (controller). Ten je prostredníkom medzi dátovým modelom a pohľadom. Jeho hlavnou úlohou je spracovávanie žiadostí (request), čiže prijímanie vstupov od užívateľa a ich následné transformovanie do príkazov ktorým rozumie či už dátový model alebo pohľad, záležiac na aktuálnych potrebách.



PHP

Doplnit/zkopcit zo starej BP!

OOP a PHP

Objektovo orientované programovanie, ďalej iba OOP, je v dnešnej dobe často skloňovaným výrazom v programátorskom svete. Ide o programovacie paradigma založené na koncepte objektov. Tieto objekty predstavujú abstrakciu reality a vo väčšine OO jazykoch sú vytvárané na základe predpisu – triedy ku ktorej patria a rovnako je tomu aj v jazyku PHP. Prvá podpora pre OOP sa objavila už vo verzii PHP 3 kde bolo umožnené vytváranie tried, objektov a ich metód. Plná podpora pre OOP sa následne objavila až vo verzii PHP 5 kedy bolo pridané našepkávanie argumentov, kľúčové slová private a protected pre premenné, menné priestory (namespaces), výnimky (exceptions) a mnohé ďalšie.

[z knihy - Matt Zandstra, PHP Objects, Patterns and Practice, Apress, 2010. ISBN 978-1-4302-2925-4.] – cit. 6. 4. 2017

PHP framework

PHP framework je OO systém - štruktúra zahŕňajúca štandardné riešenia problémov bežne vyskytujúcich sa pri popri tvorbe aplikácie čím urýchľuje tento proces. Tieto riešenia sa riadia určitými vzormi (patterns) ktoré definuje práve samotný framework a tým určuje aj výslednú architektúru aplikácie (odkaz na kap. architektúra aplikácie). Pre každý z riešených problémov existuje zväčša samostatný balík (package) ktorý poskytuje jeho komplexné riešenie. Framework nám následne tieto balíky spája spolu s ďalšími knižnicami či pomocnými programami a zabezpečuje nám komunikáciu medzi nimi, čím vytvára jeden celok ktorým je výsledná aplikácia. Rôzne frameworky nám poskytujú rozdielne riešenia rovnakých problémov ktorými môže byť napríklad vytváranie a spracovávanie formulárov alebo komunikácia s API rozhraním či databázou. Pre správu jednotlivých balíkov využívajú frameworky zvyčajne externé nástroje ako sú NPM, bower alebo composer ktorý je detailnejšie popísaný v kapitole (odkaz na kapitolu composer).

[z článku - R. E. Johnson,  Frameworks = (Components + Patterns), Communications of the ACM. ISSN 0002-0782, 1997, vol. 40, no. 10, p. 41] – cit. 6. 4. 2017

Doctrine Project

Doctrine Project je zoskupenie niekoľkých PHP knižníc zameraných na prácu s databázou a mapovanie objektov. Kľúčovými súčasťami tohoto projektu je objektovo-relačný mapper (ORM) a abstraktná databázová vrstva (DBAL) ktorá je postavená nad ním. Entity v Doctrine sú odľahčené PHP objekty uchovávajúce dáta. Triedy týchto entít predstavujú tabuľky databáze pričom jednotlivé premenné entity predstavujú stĺpce tabuľky. Entity samotné následne predstavujú riadky tabuľky ktoré je možné vkladať alebo vyberať z databázy. Toto je možné pravé vďaka mapovaniu ktoré zabezpečuje ORM. DBAL nám zase slúži na odtienenie použitých technológii, takže pri práci s Doctrine nemusíme riešiť či sú dáta uložené v MySQL, PostgreSQL či SQLite databáze. Na starosť má taktiež spracovávanie databázových udalostí či obsluhu databázových transakcií. Jej súčasťou je taktiež schema manager a query builder. V práci je využitá verzia Doctrine 2.

(zdroje <https://en.wikipedia.org/wiki/Doctrine_(PHP)> <http://jwage.com/post/31080076112/doctrine-dbal-php-database-abstraction-layer> )

Composer

Composer je nástroj na manažovanie závislostí v programovacom jazyku PHP. Bol vytvorený a zverejnený dvojicou Nils Adermann a Jordi Boggiano v marci 2012. Jeho hlavnou úlohou je manažovanie balíkov a knižníc tretých strán na úrovni aplikácie takže zvyčajne nič neinštaluje globálne aj keď poskytuje aj túto možnosť. Pod manažovaní je myslená inštalácia, aktualizácia, alebo odstránenie týchto balíkov či knižníc. Tie sú štandardne umiestnené v adresári vendor ale v princípe ich umiestnenie záleží iba na nás. V kontexte tejto práce bol composer využitý na inštaláciu frameworku Symfony 3 a následne na inštaláciu a správu balíkov tretích strán.

(zdroje <https://getcomposer.org/doc/00-intro.md> a <https://en.wikipedia.org/wiki/Composer_(software)> ) – cit. 6. 4. 2017