

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

REALIZACE JÁDRA MODULÁRNÍHO WEBOVÉHO
SYSTÉMU

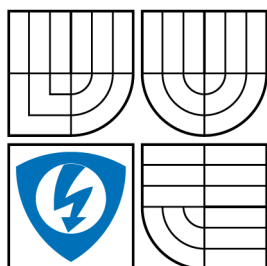
BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

MICHAL KAŠPAREC



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ



FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

REALIZACE JÁDRA MODULÁRNÍHO WEBOVÉHO SYSTÉMU

A REALIZATION OF MODULAR WEB SYSTEM

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

MICHAL KAŠPAREC

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. MARTIN KOUTNÝ

BRNO 2011



**VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ**

**Fakulta elektrotechniky
a komunikačních technologií**

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student: Michal Kašparec

ID: 115198

Ročník: 3

Akademický rok: 2010/2011

NÁZEV TÉMATU:

Realizace modulárního webového systému

POKYNY PRO VYPRACOVÁNÍ:

Prostudujte současné koncepce dostupných framework PHP systémů. V rámci bakalářské práce realizujte modulární, víceuživatelský a vícejazyčný redakční systém. V rámci práce se zaměřte na bezpečnost celého systému (frontend i backend). Zajistěte ochranu portálu vůči standardním útokům. V rámci práce navrhnete a realizujete jádro, modul správy uživatelů, modulů, jazyků a správu šablon. Navrhnete a realizujete ukázkový modul pro záznam a vyhodnocování spotřeby energií.

DOPORUČENÁ LITERATURA:

- [1] CASTAGNETTO, Jesus. PHP : programujeme profesionálně. 2. opravené a aktualizované vydání. Praha : Computer Press, 2002. 626 s. ISBN 80-7226-310-2.
- [2] MCARTHUR, Kevin. Pro PHP: Patterns, Frameworks, Testing and More. : Apress, 2008. 349 s. ISBN 978-1590598191.

Termín zadání: 7.2.2011

Termín odevzdání: 2.6.2011

Vedoucí práce: Ing. Martin Koutný

prof. Ing. Kamil Vrba, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Tato bakalářská práce popisuje současnou koncepci PHP frameworků a jejich hlavní vlastnosti. Tvorbu redakčního systému s využitím propracovaného administračního rozhraní a přístupových práv. Je realizován modul pro záznam a vyhodnocování spotřeby elektrické energie s možností grafického zobrazení průběhu spotřeby.

KLÍČOVÁ SLOVA

PHP framework, Nette, MySQL, dibi, MVC, MVP, AJAX, spotřeba energie, cena energie

ABSTRACT

This bachelor thesis describes the current conception of PHP frameworks and their main characteristics. Creating the content management system with using sophisticated administration interface and access rights. There is realization a module for recording and evaluation of power consumption with enabling to display the power consumption.

KEYWORDS

PHP framework, Nette, MySQL, dibi, MVC, MVP, AJAX, power consumption, energy price

KAŠPAREC, Michal *Realizace jádra modulárního webového systému*: bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2011. 38 s. Vedoucí práce byl Ing. Martin Koutný

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Realizace jádra modulárního webového systému“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

Brno

.....

(podpis autora)

Poděkování

Tímto bych chtěl poděkovat vedoucímu bakalářské práce Ing. Martinu Koutnému za cenné rady a za metodickou pomoc při zpracování této bakalářské práce.

OBSAH

Úvod	10
1 Charakteristika frameworku	11
1.1 K čemu slouží	11
1.2 Dostupnost, licence	11
2 Typy frameworků	12
2.1 Srovnání nejznámějších frameworků	12
2.1.1 Akelos	12
2.1.2 CakePHP	12
2.1.3 CodeIgniter	13
2.1.4 Prado	13
2.1.5 Zend	13
2.1.6 Nette	14
2.1.7 Srovnání rychlostí	14
2.1.8 Srovnání paměťových nároků	15
3 Hlavní vlastnosti Nette	17
3.1 Zabezpečení	17
3.1.1 Cross-site scripting (XSS)	17
3.1.2 Cross-site request forgery (CSRF)	17
3.1.3 Další zabezpečení	17
3.2 Ladící nástroje	18
4 Vlastní realizace jádra	19
4.1 Návrh databáze	19
4.1.1 Tabulky uživatelů	19
4.1.2 Tabulky spotřeby a energie	21
4.2 Zabezpečení proti nepovolenému přístupu	21
4.3 Jádro	22
5 Modul spotřeby	24
5.1 Administrátorská část	24
5.1.1 Dodavatelé	24
5.1.2 Kalkulačky	26
5.2 Veřejná část	26
5.2.1 Úvod	27
5.2.2 Dodavatelé	27

5.2.3	Elektřina	27
5.2.4	Moje spotřeba	29
6	Závěr	31
	Literatura	32
	Seznam symbolů, veličin a zkratk	33
	Seznam příloh	34
A	Ukázky webového systému	35
A.1	Online verze	35
A.2	Schéma databáze	36
A.3	Adresářová struktura	37
B	Obsah přiloženého CD	38

SEZNAM OBRÁZKŮ

2.1	Porovnání rychlostí jednotlivých frameworků.	15
2.2	Využití paměti u jednotlivých frameworků.	15
4.1	Diagram MVP	19
4.2	Návrh tabulek uživatelů	20
4.3	Návrh tabulek spotřeby a energie	21
4.4	Realizace jádra systému	23
5.1	Přidání produktové řady	25
5.2	Formulář kalkulačky pro výpočet ceny za spotřebu	27
5.3	Porovnání cen	29
5.4	Ukázka grafu spotřeby	30
A.1	Databáze	36
A.2	Adresářová struktura	37

SEZNAM TABULEK

2.1	Některé známé frameworky	12
-----	------------------------------------	----

ÚVOD

Se stále vzrůstající cenou za elektrickou energii hledá spousta zákazníků levnější dodavatele elektrické energie a snaží se jakýmkoli způsobem nějak ušetřit. Což bylo hlavní motivací vývoje tohoto systému. Byl vytvořen intuitivní systém, který umí zaznamenávat a vyhodnocovat spotřebu elektrické energie s využitím grafické historie průběhu spotřeby.

V této bakalářské práci je popsáno a porovnáno několik známých PHP frameworků. Následně byl vybrán PHP framework Nette, který je důkladně popsán. Byl realizován jednoduchý redakční systém s administrací a dynamickým Access Control List (ACL). Bylo využito databázového systému MySQL. Dále byl dodělán modul pro záznam a výpočet ceny a spotřeby elektrické energie. A nakonec dodělána podpora více jazyků.

V první kapitole je stručně napsáno co je to framework, k čemu se využívá, a jak ho lze získat a šířit. A také stručný popis licence open-source.

Druhá kapitola se věnuje srovnání a popisu známých a nejpoužívanějších PHP frameworků. Jsou zde porovnány i z hlediska rychlosti a paměťových nároků na systém.

Ve třetí kapitole je detailně popsán PHP framework Nette, na kterém se zakládá tato práce. Tato kapitola se věnuje zejména zabezpečení a ladícím nástrojům, které má Nette opravdu brilantní.

Další kapitola se zabývá realizací jádra, na kterém je postaven celý systém, návrhem databáze a popisem zabezpečení do administrační části, včetně popisu jednotlivých důležitých presenterů.

V poslední kapitole je popis realizace modulu spotřeby, vzorový příklad výpočtu ceny za spotřebu elektrické energie s obrázkem srovnání s jinými dodavateli a přiblížení zaznamenávání historie spotřeby uživatele.

1 CHARAKTERISTIKA FRAMEWORKU

Framework je balík naprogramovaných tříd a objektů, ve kterých je definováno, jak se mají používat, upravovat nebo vytvářet nové třídy, objekty a modely.

1.1 K čemu slouží

Framework má za úkol převzít typické problémy dané oblasti, a tím usnadnit vývoj tak, aby návrháři a vývojáři mohli pracovat pouze na svém zadání. Při používání frameworku je potřeba dodržovat určitá pravidla kvůli základní funkčnosti frameworku a některá doporučená pravidla, aby se dalo orientovat v cizích zdrojových kódech.

Objevují se námitky, že při použití frameworku bude kód pomalý, či jinak neefektivní, a že čas, který se ušetří použitím cizího kódu, se musí věnovat nastudování frameworku. Nicméně při jeho opakovaném nasazení nebo ve velkém projektu dojde k výrazné úspoře času.

1.2 Dostupnost, licence

Existuje několik typů frameworků. „Soukromé“ frameworky, které vytváří pouze jeden programátor a pouze pro vlastní účely. Jejich velkou nevýhodou je, že nad tvorbou frameworku musí programátor strávit spoustu času než ho vytvoří, otestuje a opraví chyby. Dále jsou frameworky, které vytváří tým více programátorů v jedné firmě. Tyto frameworky většinou zůstávají v dané firmě a dále se nešíří. A na závěr největší a nejrozsáhlejší skupina zvaná „open-source“ frameworky. Vyvíjí je nezávisle na sobě několik vývojářů a vyvíjí dohromady jeden framework. Kolem takových frameworků vzniká komunita vývojářů, kteří využívají tento framework pro své účely.

Open-source frameworky jsou vyvíjeny pod licencí GNU GPL [3]. Tato licence umožňuje, že software, který je pod touto licencí vydán, může být volně distribuován, spouštěn, měněn, kopírován, studován a zlepšován. Je možné ho jakkoli přizpůsobit pro své účely, přidávat své nápady a poté tato vylepšení zveřejnit ostatním. Nemusí se za něj nic platit a není nutné nikoho žádat o povolení.

2 TYPY FRAMEWORKŮ

Na základě studia nejpoužívanějších frameworků [9] bylo vybráno několik zajímavých frameworků, které byly důkladně prostudovány a srovnány v tab. 2.1.

Tab. 2.1: Některé známé frameworky

Framework	PHP4	PHP5	Multi BD	MVC	AJAX	Moduly
Akelos	ANO	ANO	ANO	ANO	ANO	ANO
CakePHP	ANO	ANO	ANO	ANO	ANO	ANO
CodeIgniter	ANO	ANO	ANO	ANO	NE	NE
Prado	NE	ANO	ANO	ANO	ANO	ANO
Zend	NE	ANO	ANO	ANO	ANO	ANO
Nette	NE	ANO	ANO	ANO	ANO	ANO

2.1 Srovnání nejznámějších frameworků

2.1.1 Akelos

Tento MVC framework je nyní dostupný ve verzi 1.0.1. Akelos je určen pro PHP 4 i PHP 5 [8]. Přebírá některé funkce ze známého frameworku Ruby on Rails, určeného pro programovací jazyk Ruby.

- Pro přístup k databázi používá Active Record. Podporuje databáze MySQL [7], PostgreSQL a SQLite.
- Obsahuje několik základních pluginů, jako je kalendář, asistent pro vytváření základního administračního rozhraní, či `dynamic_finder`, který umožňuje magické vyhledávání v Active Record modelech.
- Chybí pokročilejší validace.
- V každém kontroléru definujeme modely, se kterými je svázán. Poté můžeme využívat jejich funkcí.
- Jednoduchá konfigurace za pomoci grafického rozhraní.

2.1.2 CakePHP

Framework je dostupný ve verzi 1.3.6. CakePHP je určen pro PHP 4 i pro PHP 5. Patří mezi nejvíce rozšířené frameworky, a proto je na internetu k dispozici i mnoho českých návodů.

- Používá ORM knihovnu. Možnost připojení na databáze MySQL, PostgreSQL, ADOdb, Firebird DB2, MSSQL, Oracle, SQLite, ODBC.

- Obsahuje komponenty pro práci s emaily, autentifikaci a mnoho dalších.
- Velmi dobře zpracovaná validace, která se provádí v modelu.
- Kromě standardních helperů zde můžeme nalézt také helpery pro AJAX či RSS.

2.1.3 CodeIgniter

Framework je aktuálně dostupný ve verzi 1.7.3. Je určen pro PHP 4 i pro PHP 5. Jedná se o velmi jednoduchý a intuitivní framework, který je vyvíjen firmou a ne komunitou. Má velmi dobrou dokumentaci.

- Podpora databází MySQL, MSSQL, PostgreSQL, Oracle, SQLite, ODBC. Pro práci s databází používá upravenou třídu Active Record. Tato „ORM“ třída však neumí pracovat s relacemi.
- Najdeme zde několik užitečných pluginů pro práci s Excelem, CAPTCHA, generátory PDF, či geografické lokátory, které podle IP adresy určují pozici.
- Validace probíhá v kontroléru, kde je možné definovat jednoduchým zápisem i složitější pravidla.
- Modely, které chceme v dané metodě používat, je třeba načíst pomocí funkce load.

2.1.4 Prado

Aktuálně se nachází ve verzi 3.1.7. Je určen pouze pro PHP 5. Prado není klasický PHP framework. Je založený na komponentovém přístupu a programování událostí. Jeho vzorem při vývoji nebyl Ruby on Rails jako u většiny PHP frameworků, ale inspiroval se spíše ASP.NET.

- Podporuje databáze MySQL, PostgreSQL, SQLite, MSSQL, Oracle.
- Pro přístup do databáze je možné volit mezi PDO, Active Record nebo kompletním objektově mapovým schématem SQLMap.
- Obsahuje validaci základních datových typů, dále je také možnost přidat k formuláři kontrolu CAPTCHA.

2.1.5 Zend

Patří mezi nejznámější PHP frameworky. Aktuální verze je 1.11.1. Je vyvíjen přímo firmou Zend, která vyvíjí jádro PHP. Celý framework je objektově orientovaný a implementovaný v PHP 5. Zend framework jako celek využívá návrhový vzor MVC.

- Podporuje databáze MySQL, MSSQL, SQLite, Oracle, PostgreSQL, IBM DB2.
- Pro přístup do databáze využívá PDO.

- Součástí Zendu je celá řada modulů, např. ověřování práv, posílání emailů nebo tvorba PDF dokumentů.
- Lze využít šablonovací systém SMARTY¹.

2.1.6 Nette

Jedná se o český PHP framework jehož autorem je David Grundl. Aktuální stabilní verze je 0.9.7b a je psaný v PHP 5 s plným využitím objektově orientovaným programováním. Původně byl psán jako „soukromý“ framework, avšak později byl uvolněn jako open-source. Díky tomu má možnost se rychleji a lépe rozšiřovat o nové funkce.

- Pro práci s databází používá knihovnu dibi²
- Používá vlastní šablonovací systém.
- Konfigurace úpravou .ini souboru.

Hlavní výhoda PHP frameworků, které nepodporují PHP 4, ale pouze PHP 5 je, že obsahují zcela nový objektový model, který přináší možnost plnohodnotného objektově orientovaného návrhu včetně abstraktních rozhraní.

2.1.7 Srovnání rychlostí

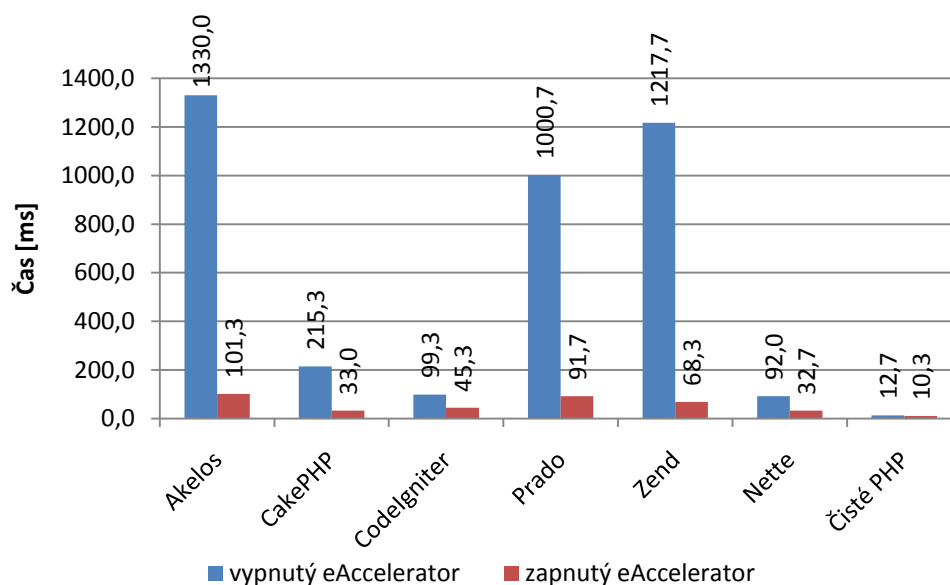
Hodnoty, uvedené v následujícím grafu, jsou průměry hodnot získané z testování [1]. Test byl složen ze čtyř kroků (30x zobrazení členů, 30x zobrazení uživatelů, 30x editační formulář, 30x update dotaz na databázi MySQL). Do testu bylo pro srovnání zahrnuto i čisté PHP.

Z grafu na obr. 2.1 je vidět, že problém s rychlostí má několik PHP frameworků, jejichž průměrný čas překračoval jednu sekundu. Překvapivě mezi ně patří i velice oblíbený Zend framework. Při zapnutí eAcceleratoru³ se časy všech frameworků značně zlepšily a už by nebyl problém s jejich provozem.

¹SMARTY - je komplexní a funkčně velmi dobře vybavený šablonovací systém pro PHP.

²Dibi - databázová vrstva určená pro PHP 5.

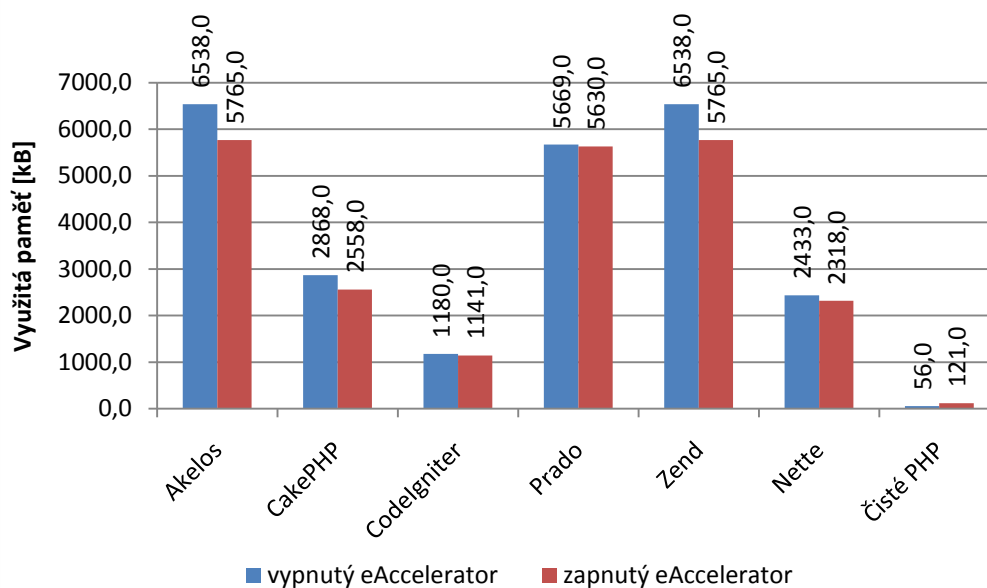
³eAccelerator je velmi užitečný nástroj pro „zrychlení“ PHP. Funguje na principu vytváření mezipaměti (cache) PHP skriptů.



Obr. 2.1: Porovnání rychlostí jednotlivých frameworků.

2.1.8 Srovnání paměťových nároků

Kromě rychlosti frameworků hraje významnou roli také jejich paměťová náročnost. Jak je vidět na obr. 2.2, tak zapnutí eAcceleratoru nemá žádný význam. Většina frameworků se drží okolo 5MB využité paměti.



Obr. 2.2: Využití paměti u jednotlivých frameworků.

Z výše uvedených PHP frameworků byl vybrán Nette framework. Má velkou komunitu, která zajišťuje, že se framework stále vyvíjí a vytváří se mnoho doplňků. Dalším důležitým aspektem je, že se jedná o český framework. V testech rychlosti patří mezi nejrychlejší PHP frameworky a má jedny z nejmenších paměťových nároků. Práce je postavena na *Nette Framework 2.0-alpha2 pro PHP 5.3* z důvodu lepší kompatibility doplňků a hlavně pro snazší přechod na novější beta verzi, která bude velmi brzy uvolněna, než při přechodu z verze stable (0.9.7b).

3 HLAVNÍ VLASTNOSTI NETTE

Nette Framework je výkonný framework pro pohodlné a rychlé vytváření kvalitních a moderních webových aplikací v PHP 5. Eliminuje bezpečnostní rizika, podporuje AJAX, DRY, KISS, MVC a znovupoužitelnost kódu.[4]

3.1 Zabezpečení

3.1.1 Cross-site scripting (XSS)

Cross-site scripting je metoda narušení webových stránek zneužívající neošetřených výstupů. Útočník pak dokáže do stránky podstrčit svůj vlastní kód a tím může stránku pozměnit nebo dokonce získat citlivé údaje o návštěvnicích. Proti XSS se lze bránit jen důsledným a korektním ošetřením všech řetězců.

Nette Framework přichází s revoluční technologií *Context-aware escaping*, která vás provždy zbaví rizika XSS. Všechny výstupy totiž ošetřuje automaticky. Pokud např. zadáte:

```
$message = 'Šířka 1/2''
```

framework vygeneruje kód:

```
&quot;Šířka 1\2\&quot;&quot;;
```

3.1.2 Cross-site request forgery (CSRF)

Cross-site request forgery spočívá v tom, že přimějeme uživatele navštívit stránku, která skrytě vykoná útok na webovou aplikaci, kde je uživatel právě přihlášen. Lze tak například pozměnit nebo smazat článek, aniž by si toho uživatel všiml. Proti útoku se lze bránit generováním a ověřováním autorizačního tokenu.

Nette Framework obranu formulářů před útokem CSRF zjednodušuje tak, že snadnější to být nemůže. Stačí totiž jediný příkaz:

```
$form->addProtection();
```

3.1.3 Další zabezpečení

Nette obsahuje ještě spoustu dalších zabezpečení. Mezi další významné patří např. *URL attack*, *control codes*, *invalid UTF-8*. Tyto zabezpečení obstarávají důsledné ošetřování všech vstupů na úrovni jednotlivých bajtů. Tím Nette brání útočníkovi podstrčit webové aplikaci *škodlivý* vstup.

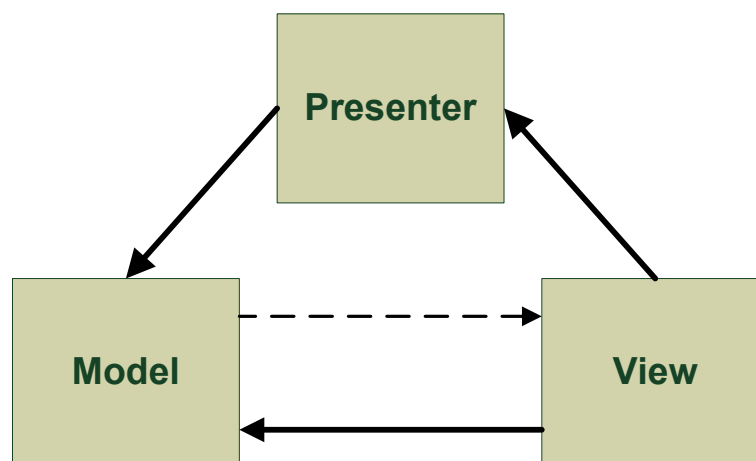
Za zmínku ještě stojí *session hijacking*, *session stealing* a *session fixation*. Nette si samo nakonfiguruje PHP na serveru tak, aby útočník nemohl zcizit nebo podstrčit uživateli své session ID a díky tomu získat přístup do webové aplikace, aniž by znal přihlašovací údaje uživatele.

3.2 Ladící nástroje

Zajímavou výhodou je debugovací nástroj zvaný „laděnka“. V klasickém případě, kdy je udělána při programování chyba, se zobrazí chybová hláška, kde méně znalý programátor těžce dohledává chybu. Při zapnutí debugovacího nástroje je chování mnohem přívětivější k vyhledání dané chyby. V objektu, kde nastala chyba, zobrazí část PHP souboru s jeho kódem a zvýrazní kód, kde se chyba vyskytla. Defaultně je nastaveno, že framework sám pozná zda se nachází na vývojovém serveru a nebo na produkčním. Podle toho „laděnkou“ zapne nebo vypne. Pokud je však nastaveno přímo, aby byla zapnuta, tak může nastat bezpečnostní riziko při nasazení projektu na „ostrý“ provoz. V takovém případě může potenciální útočník prohlížet přímo zdrojové kódy.

4 VLASTNÍ REALIZACE JÁDRA

Celé jádro je realizováno s ohledem na zabezpečení celé aplikace. Podle toho také byly realizovány jednotlivé presentery a hlavně autentizace a databáze. Jelikož Nette využívá vzor Model-View-Presenter (MVP)¹ [5], tak podle toho byly dále vyvíjeny všechny části aplikace.



Obr. 4.1: Diagram MVP

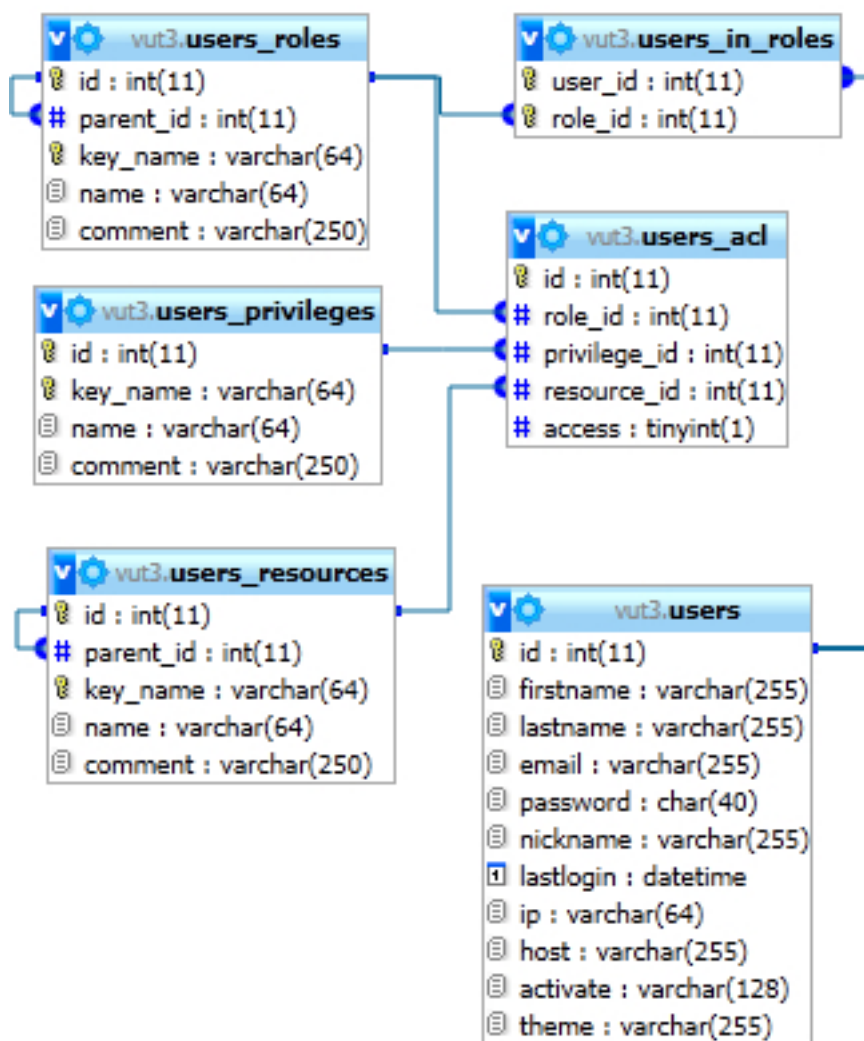
Činnosti modelů jsou omezeny pouze na získávání dat z databáze, a jak vyplývá z MVP vzoru, model neví o tom, že nějaké view (šablony) a presentery existují. Šablony byly navrženy tak, aby získávaly data z modelu přes presenter, kde se předtím zpracovala. Logika v šablonách byla omezena pouze na iterace, if, else. Presenter spojuje šablonu s modelem a realizuje různé uživatelské akce.

4.1 Návrh databáze

4.1.1 Tabulky uživatelů

Celá databáze byla navrhnutá s ohledem na zabezpečení celé aplikace. Jednotlivé tabulky, které se starají o uživatelské role a oprávnění, byly navrhnuty tak, aby se daly jednoduše přidávat, odebírat nebo měnit role, zdroje a oprávnění. Návrh databáze uživatelů je znázorněn na obr. 4.2.

¹MVP je obdoba MVC, kde *presenter* v Nette je totéž, co *controller* v jiných frameworkích.



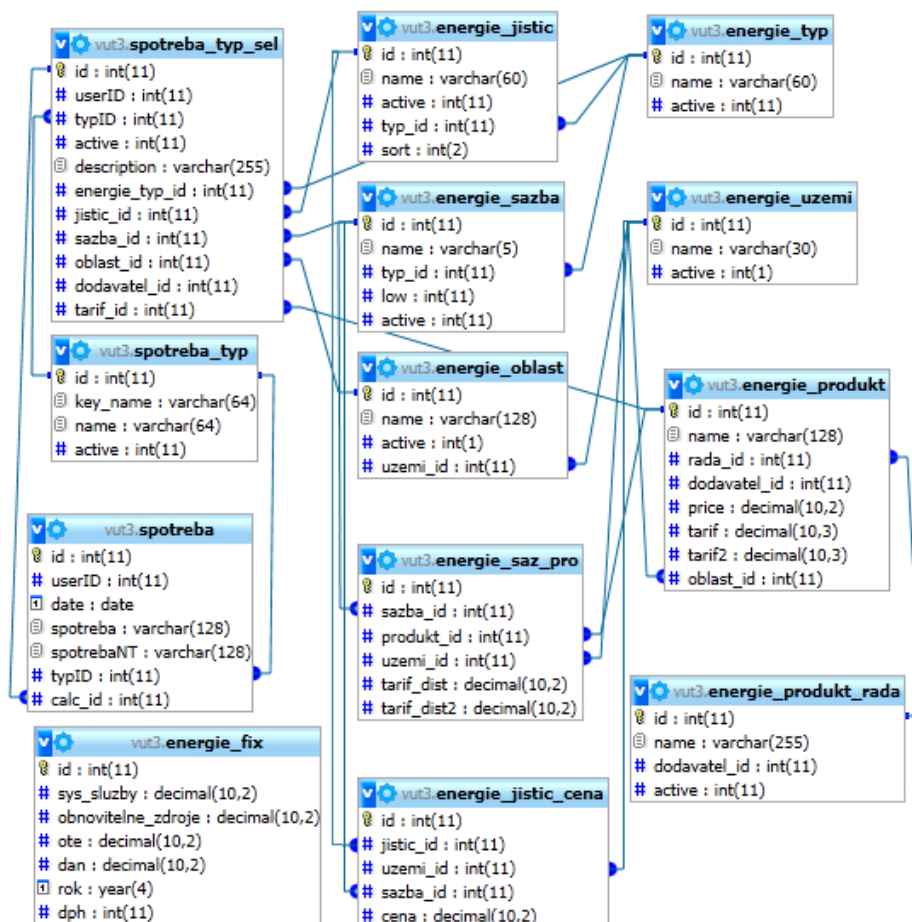
Obr. 4.2: Návrh tabulek uživatelů

Jelikož byla zvolena metoda dynamických rolí, musela být přepracována i třída ACL. V tabulce `users_roles` jsou definovány uživatelské role, do kterých mohou být jednotliví uživatelé přiřazeni. Jednotlivé role mohou být zděděny od rolí s nižší prioritou, ale výjimečně i s vyšší. Tabulka `users_resources` určuje, ke kterým částem aplikace může uživatel přistupovat (např. administrační rozhraní). Tabulka `users_privileges` definuje, jaké akce může uživatel vykonávat.

V tabulce `users_acl` jsou jednotlivým rolím přidělována práva. Tabulka uživatelé v rolích (`users_in_roles`) přiřazuje jednotlivým uživatelům role. Každý uživatel může mít libovolný počet rolí. Výchozí role pro každého „návštěvníka“ je role *guest*. Po přihlášení se jeho role změní na *registered*.

4.1.2 Tabulky spotřeby a energie

Tabulky spotřeby a energie (obr. 4.3) jsou vytvořeny tak, aby co nejefektivněji plnily svůj úkol a byly snadno rozšiřitelné o další parametry. Díky relacím a referenční integritě je hlídáno, aby nebyla smazána data, na které jsou vázány další záznamy.



Obr. 4.3: Návrh tabulek spotřeby a energie

Význam většiny tabulek je zřejmý. Tabulka **energie_fix** obsahuje pevné ceny, které jsou pro všechny stejné a neměnné. V tabulce **energie_jistic_cena** jsou ceny za jednotlivé jističe pro všechny sazby a území. Tabulka **energie_saz_pro** obsahuje ceníky všech produktů a dodavatelů.

4.2 Zabezpečení proti nepovolenému přístupu

Celá aplikace je chráněna přihlašovacím systémem, který vyžaduje na ověření zadat email a heslo. Bez správně zadaných údajů není umožněn přístup do administrační části aplikace.

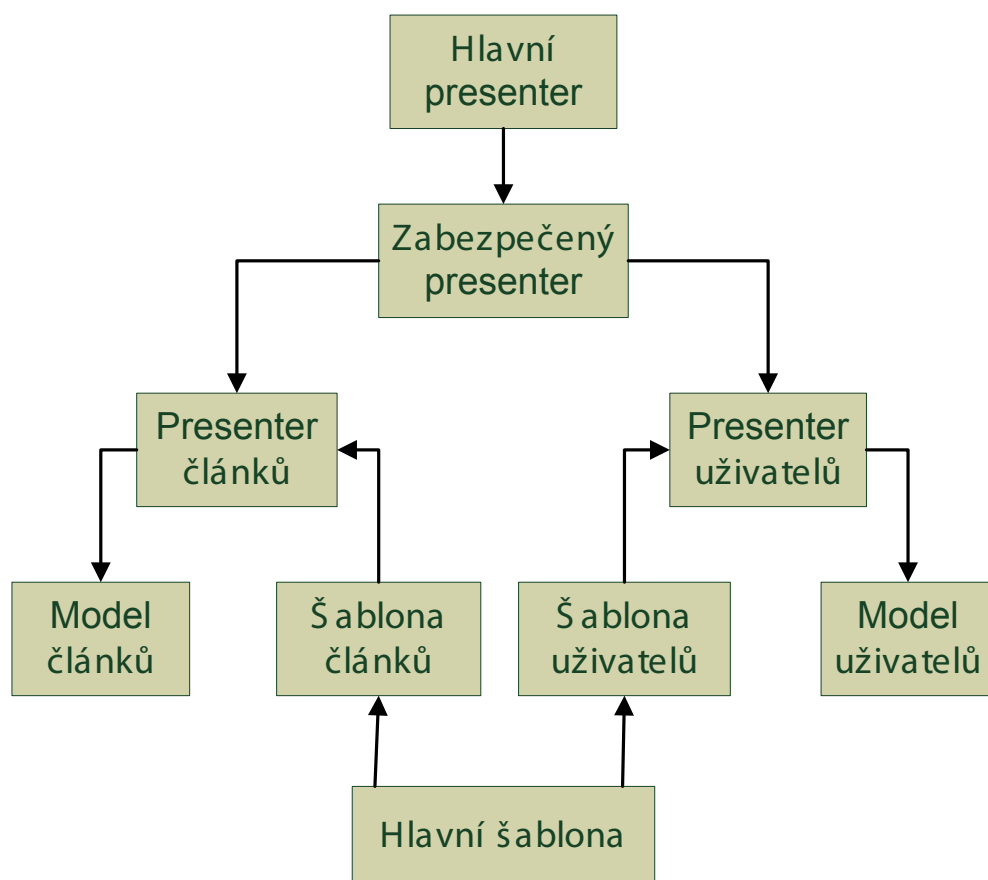
Po vyplnění přihlašovacích údajů a odeslání na server přijde na řadu autentizační skript (viz. příloha). Při autentizaci je zavolána metoda `authenticate(array $credentials)`, která má jako vstupní parametry zadané přihlašovací údaje. Nejdříve otestuje, zda je zadaná emailová adresa zaregistrována. Pokud není uživatel v databázi nalezen, vrátí systém výjimku a skript se ukončí. Při správně zadaném emailu se pokračuje na ověření hesla.

Heslo je v databázi uloženo, z bezpečnostních důvodů, v šifrované formě. Jako šifrovací algoritmus byla použita metoda SHA1. Aby byla hesla ještě více ochráněna, byla rozšířena o dalších 10 náhodných znaků. Řetězec je uložen v konfiguračním souboru a nesmí se později měnit. Při ověřování se použije příkazem `$config = Environment::getConfig('security')` a k heslu se přiloží pomocí příkazu `$config->salt`. Jestliže autentizace úspěšně projde, vrátí metoda zpět identitu uživatele, která je uchována v super globální proměnné `$_SESSION`, ve formě jména uživatele a jeho role.

4.3 Jádro

Na obrázku 4.4 je vidět návrh administrační části systému. Frontend systému je řešen stejně, jen neobsahuje *zabezpečený presenter*.

- *Hlavní presenter* rozšiřuje třídu *presenter*, která reaguje na události pocházející od uživatele a zajišťuje změny v modelu nebo v pohledu. Hlavní presenter je omezen pouze na nastavení před načtením stránky, jako načtení menu, nastavení výchozího vzhledu a pokud je již uživatel přihlášen, tak načtení uživatelských dat.
- *Zabezpečený presenter* rozšiřuje třídu *hlavní presenter* o kontrolu přístupových práv do administrační části (ACL). Dále hlídá nečinnost uživatele a v případě dlouhé neaktivity ho z bezpečnostních důvodů odhlásí. Pokud uživatel není přihlášen, přesměruje ho na autorizační presenter.
- *Hlavní šablona* je výchozí stránka, která je pro všechny další stránky stejná a mění pouze část, která zobrazuje události vyžádané uživatelem. Ty získává přes další šablony, které jsou na to přizpůsobeny.
- *Šablona článků a šablona uživatelů* zobrazují odezvy na události vyvolané uživatelem, které zpracovává jejich presenter. Ten reaguje na činnost uživatele a podle toho si vyžádá od modelu data z databáze. Samotné presentery vždy rozšiřují *zabezpečený presenter*. Tím je každý presenter ochráněn proti nepovolenému přístupu nebo proti uživatelům, kteří nemají dostatečná oprávnění pro přístup.



Obr. 4.4: Realizace jádra systému

5 MODUL SPOTŘEBY

Modul spotřeby byl realizován tak, aby byl co nejméně závislý na jádře celého systému. Důvodem byla co nejlepší oddělitelnost od jádra při případné aktualizaci modulu, či jeho odstavení. Při odstranění modulu by mělo jádro fungovat bez potíží dále.

Celá realizace probíhala tak, aby ve výsledku bylo pro uživatele co nejjednodušší spravovat veškeré důležité části systému. Při jednotlivých úkonech, které uživatel vykoná, se provede ověření, zda operaci může vykonat a jestli jsou zadány všechny potřebné parametry.

5.1 Administrátorská část

Celý modul byl rozdělen do dvou menších submodulů z důvodu lepší přehlednosti a správy.

- dodavatelé
- kalkulačky

5.1.1 Dodavatelé

Submodul dodavatelů elektrické energie se stará o jednotlivé dodavatele, jejich produktové řady, produkty a o ceníky k jednotlivým produktům a oblastem.

Teoretická část

Česká republika je rozdělena na tři oblasti pro rozvod elektřiny.

- PRE – Praha,
- EON – Jižní Morava + Jižní Čechy,
- ČEZ – Severní Morava + celé Čechy (mimo Prahu a Jižních Čechy).

Podle tohoto rozdělení určují různí dodavatelé své ceníky. Tím je způsobeno, že jeden dodavatel má pro oblast, která patří pod oblast PRE, jiné ceny než pro oblasti EON a ČEZ.

Dodavatelé

V této části systému je možné přidat, upravit a nebo odebrat jednotlivé dodavatele. Při vytvoření nového záznamu o dodavateli je nutné vyplnit několik základních údajů. Jako nejdůležitější je název dodavatele. Dále je možné vyplnit další údaje jako je typ společnosti (právní forma), datum vzniku, sídlo firmy a jiné. Všechny údaje je možné později změnit přes úpravy. Při mazání dodavatele se kontroluje jestli

jsou na dodavatele vázány další data. Pokud má některý uživatel vybraný produkt od dodavatele, který má být smazán, je jeho smazání znemožněno. V případě, že jsou u dodavatele pouze vyplněny produktové řady a produkty, je možné je smazat po odsouhlasení varování, že data budou nenávratně smazána. Přidání dodavatele je umožněno oprávněným uživatelům i z veřejné části systému.

Produktové řady

Někteří dodavatelé mají více produktových řad svých produktů. Tyto produktové řady je možné využívat při splnění určitých podmínek nebo jsou pro určité období.

Produktové řady dodavatele

Název řady

Dodavatel

Stav

Řada	Dodavatel	Stav	Akce
Garance	BOHEMIA ENERGY	✓	Upravit Smazat
Online	BOHEMIA ENERGY	✓	Upravit Smazat
Comfort	ČEZ Prodej, s.r.o.	✓	Upravit Smazat
Elektřina	E.ON Energie, a.s.	✓	Upravit Smazat

Obr. 5.1: Přidání produktové řady

Produkty

Každý dodavatel má několik produktů, které jsou rozděleny podle distribučních sazeb. Produkty jsou vázané na produktové řady a tím je možné mít jen několik produktů a jejich ceníky se dělí podle toho do které produktové řady patří.

Přidání produktu je podmíněno výběrem dodavatele a produktové řady dodavatele. Při splnění této podmínky bude umožněno pokračovat dále, kde je potřeba vyplnit název produktu, ceny za silovou elektřinu (měsíční poplatek a cena za dodanou energii na MWh) a oblast pro kterou je produkt určen.

Ceníky

Ceníky dodavatelů jsou složeny z několika částí. Daň z elektřiny je stejná pro všechny dodavatele a oblasti. Cena silové elektřiny je dána měsíčním poplatkem za dodávku elektřiny a poplatkem za dodávku elektřiny na MWh, která je rozdělena na VT a NT. Dále jsou poplatky podle rozhodnutí Energetický regulační úřad (ERÚ). Jedná

se o poplatky za systémové služby (155,40 Kč/MWh), cena na podporu výkupu elektřiny (370,00 Kč/MWh) a cena za činnost zúčtování OTE (4,75 Kč/MWh). A v neposlední řadě cena za distribuované množství energie (Kč/MWh), která je rozdělena na cenu za VT a NT (pokud daná sazba má i nízký tarif).

Při přidávání ceníku je nutné vybrat produkt a poté ve vstupním formuláři zvolit pro který typ odběru bude určen (domácnost nebo firma). Tím je umožněno načíst aplikaci příslušné hodnoty pro sazby. Podle sazby se zjistí, zda má nízký tarif či nikoli. Podle toho je umožněno zapsat hodnotu ceny za distribuci.

5.1.2 Kalkulačky

V části kalkulaček je možné spravovat všechny důležité parametry, které se týkají výpočtu ceny za spotřebovanou energii.

Území

V této části jsou přiřazeny jednotlivé území odběru pod jednotlivé oblasti (ČEZ, EON, PRE).

Ceny za jističe

Podle hodnoty jističe, distribuční sazby a oblasti se zde vyplňuje cena, která se platí měsíčně za nainstalovaný jistič. Pro jednotlivé oblasti jsou ceny za jistič shodné. Toho bylo využito při přidávání ceníků, které se tímto značně zjednodušilo a zrychlilo.

Pevné poplatky

Zde se zadávají pevné poplatky, které určuje každý rok ERÚ, daň z elektřiny a DPH a jsou stále po celou dobu pro všechny oblasti a dodavatele.

5.2 Veřejná část

Celá část systému určená pro veřejnost je lokalizována do dvou jazyků. Standardně je systém psaný v češtině a může být přepnut do angličtiny. Výchozí nastavení jazyka se provádí podle nastavení prohlížeče.

Vzhled stránky je konstruován tak, aby byl co nejpřehlednější a bylo jasné co který prvek dělá. Vrchní část stránky obsahuje možnost přepnutí jazyka zobrazení a hlavní navigační menu. V prostřední části je zobrazován obsah stránky, který se dynamicky mění a v pravé části je přihlašovací formulář.

5.2.1 Úvod

Na úvodní stránce je souhrn důležitých věcí. Nachází se zde poslední novinky a zjednodušená kalkulačka spotřeby elektrické energie.

5.2.2 Dodavatelé

Na stránce dodavatelé se nachází tabulka s dodavateli elektrické energie. U každého dodavatele je zde uvedeno jméno jak je zapsán v obchodním rejstříku, zda dodává elektřinu, plyn a nebo oboje. Po rozkliknutí se dostaneme do detailu, kde je uvedeno ještě několik dalších údajů (právní norma, datum vzniku, sídlo, hlavní předmět podnikání, počet zákazníků, počet zaměstnanců, odkaz na stránky a stručný popis) včetně komentářů od uživatelů.

5.2.3 Elektřina

Tato část se věnuje výpočtům ceny za spotřebovanou elektrickou energii a návrhu cen ostatních dodavatelů. Na obrázku 5.2 je zobrazen formulář pro výpočet ceny za spotřebovanou elektrickou energii.

Typ odběru	
Domácnost ▼	
Distribuční sazba	Nainstalovaný jistič
D25d ▼	nad 3x16 A do 3x20 A včetně ▼
Oblast odběru	
Jihomoravský kraj ▼	
Současný dodavatel	Produkt
BOHEMIA ENERGY ▼	Home Aku 8 ▼
Spotřeba VT	Spotřeba NT
0	0
Odeslat	

Obr. 5.2: Formulář kalkulačky pro výpočet ceny za spotřebu

Formulář se musí vyplňovat postupně, protože se příslušná data načítají podle zvolených filtrů. Po vybrání typu odběru se načtou položky distribuční sazba a nainstalované jističe. Z distribuční sazby se poté určí zda má i nízký tarif. Oblast odběru slouží k určení ceníku podle oblasti, kde se zadavatel nachází. Z těchto zadaných hodnot se teprve může vygenerovat položka dodavatelů a jejich produktů.

Poté stačí už jen zadat spotřebu ve vysokém, případně i v nízkém tarifu v kWh.

Celková roční platba za elektřinu se skládá ze tří částí [2]:

- Stálé platby = $12 * (A + B)$,
- Platba za spotřebu ve VT = Spotřeba VT [MWh] * $(1 + 3 + 5 + 6 + 7 + 8)$,
- Platba za spotřebu v NT = Spotřeba NT [MWh] * $(2 + 4 + 5 + 6 + 7 + 8)$.

Roční platba celkem = Stálé platby + Platba za spotřebu ve VT + Platba za spotřebu v NT

Vysvětlivky:

A = platba za silovou elektřinu,

B = platba za příkon podle proudové hodnoty hlavního jističe,

1 = cena za silovou elektřinu ve VT za MWh,

2 = cena za silovou elektřinu v NT za MWh,

3 = cena za distribuci ve VT za MWh,

4 = cena za distribuci v NT za MWh,

5 = cena systémových služeb za MWh,

6 = cena na podporu výkupu elektřiny za MWh,

7 = cena za činnost zúčtování OTE za MWh,

8 = daň z elektřiny za MWh.

Vzorový příklad:

Typ odběru:	domácnost
Distribuční sazba:	D25
Nainstalovaný jistič:	3x20 A
Oblast odběru:	Jihomoravský kraj
Současný dodavatel:	Bohemia Energy
Produkt:	Home Aku 8
Spotřeba VT:	2000 kWh
Spotřeba NT:	3000 kWh

Výpočet:

$$A = 12 * (55 + 84) = 1688 \text{ Kč},$$

$$B = 2 * (1752 + 1846,39 + 155,40 + 370 + 4,75 + 28,30) = 8313,68 \text{ Kč},$$

$$C = 3 * (975 + 27,63 + 155,40 + 370 + 4,75 + 28,30) = 4683,24 \text{ Kč},$$

kde A = stálé platby, B = platba za spotřebu VT, C = platba za spotřebu NT.

Cena celkem = $1688 + 8313,68 + 4683,24 = 14664,92$ Kč (17597.90 Kč s DPH).

Systém vypočítá cenu ze zadaných údajů a poté přepočítá ceny i pro zbývajících dodavatele a navzájem je porovná a zobrazí přehled v tabulce (5.3).

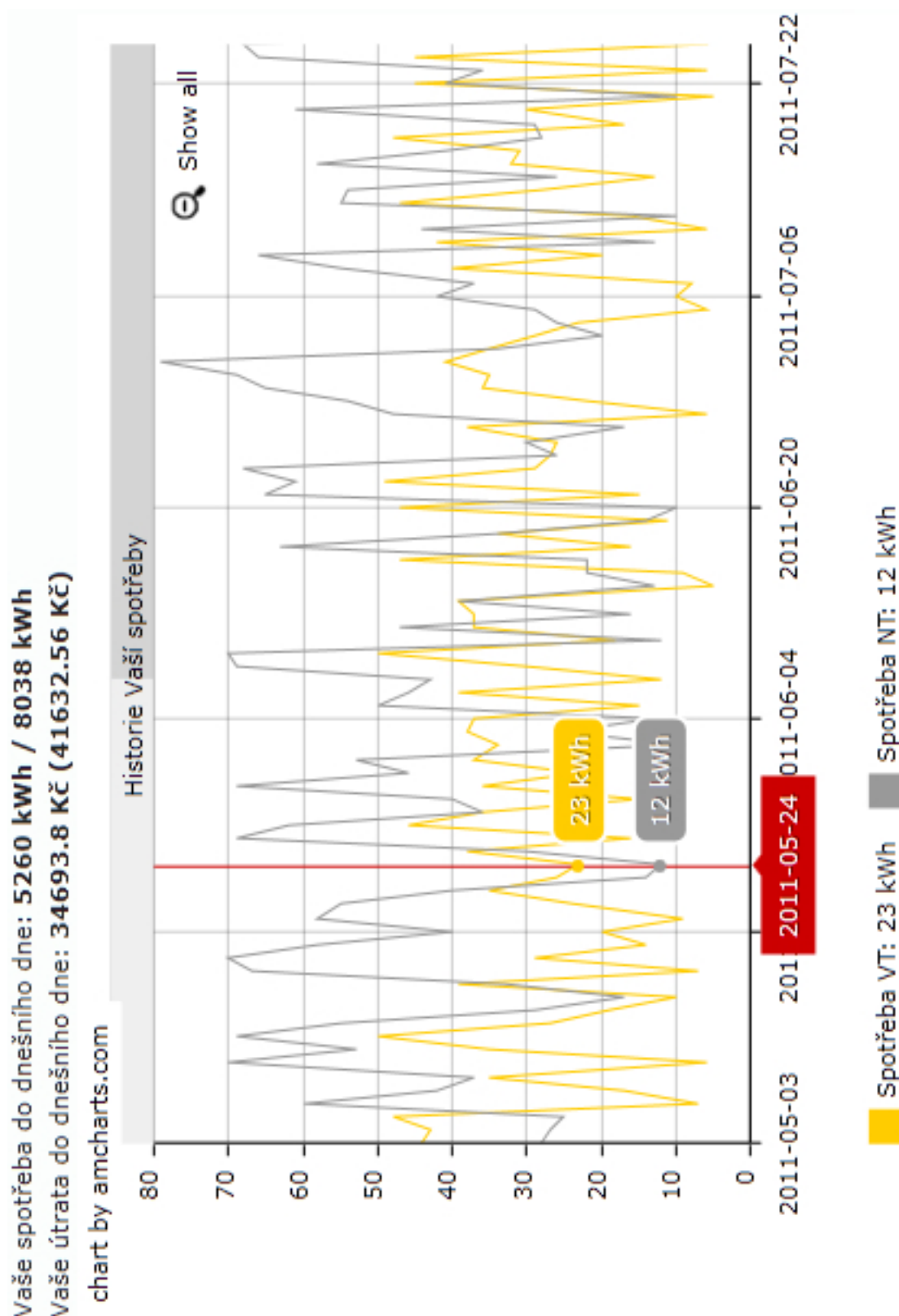
Dodavatel, produkt	Cena	Cena s DPH	Úspora (s DPH)
Europe Easy Energy EASY, EASY Aku 8	14539.92 Kč	17447.9 Kč	125 Kč (150 Kč)
BOHEMIA ENERGY Garance, Home Aku 8	14664.92 Kč	17597.9 Kč	stávající produkt
E.ON Energie, a.s. Elektřina, ElektřinaAku	15030.92 Kč	18037.1 Kč	-366 Kč (-439.2 Kč)

Obr. 5.3: Porovnání cen

5.2.4 Moje spotřeba

Modul moje spotřeba slouží k ukládání spotřeby uživatele podle jednotlivých dní a následné zobrazení historie spotřeby v grafu. Před prvním použitím je nutné nejdříve si aktivovat některou kalkulačku, která může být později spravována na kartě můj účet v sekci kalkulačky. Zobrazuje i celkovou spotřebovanou energii od prvního zadání spotřeby a dále se přepočítává podle zadaných údajů na cenu. Příklad zobrazení je vidět na obrázku 5.4.

Grafy jsou vykreslovány pomocí javascriptu, proto je nutné mít ho v prohlížeči povolen. Výsledný graf je zobrazen jako SVG objekt. Tím je umožněno vybrání určitého úseku grafu a přiblížit ho. Dále po najetí myši jsou zobrazeny hodnoty spotřeby v daném datu. Při dvoutarifové sazbě jsou v grafu zobrazeny průběhy spotřeby jak pro VT, tak pro NT. Ve spodní části grafu je legenda a při kliknutí na ni je možné jeden nebo oba grafy skrýt.



Obr. 5.4: Ukázka grafu spotřeby

6 ZÁVĚR

V této práci jsou srovnány nejznámější PHP frameworky. Všechny zde uvedené frameworky podporují moderní technologie včetně podpory více databází, MVC modelu a dnes hodně využívané technologie AJAX. PHP frameworky, které podporují PHP 4, mají výhodu, že by byly funkční i na starších systémech bez podpory PHP 5, ale v dnešní době je PHP 5 dostupné na každém hostingu, a tím pádem je podpora starší verze PHP spíše na obtíž. Zbytečně brzdí vývoj celé aplikace, protože až od verze PHP 5 je možný plnohodnotný objektově orientovaný návrh.

Po důkladném prozkoumání všech vlastností byl vybrán PHP framework Nette, ve kterém byl navrhnut celý systém. Hlavní důraz byl kladen na bezpečnost a jednoduchost celé aplikace. Proto je kladen větší důraz na návrh databáze a autentizaci uživatelů. Neregistrovaní uživatelé dostávají automaticky od systému roli *guest*, se kterou mají oprávnění pouze prohlížet stránky. Do administrační části má přístup pouze uživatel v roli *admin* nebo jím určená další role.

Po zabezpečení celé aplikace byl vytvořen modul pro záznam a vyhodnocování spotřeby energií. Kalkulačka pro jednorázový výpočet ceny za spotřebovanou elektrickou energii a následné porovnání cen s konkurenčními dodavateli, kde jsou jednotlivé ceny rozděleny barevně podle výhodnosti. Další modul spotřeby, který umožňuje dlouhodobý záznam spotřeby s možností zobrazovat celkovou spotřebu a cenu. Průběh spotřeby je zobrazován v jednoduchém grafu, který umí skrýt jednotlivé křivky, různě vybírat části grafu a přibližovat je. Nakonec bylo v systému umožněno využívat podporu více jazyků.

LITERATURA

- [1] DANĚK, P.: Velký test PHP frameworků. [online], c2008 [cit. 1. 12. 2010].
URL <http://www.root.cz/clanky/velky-test-php-frameworku-2008/>
- [2] ČEZ: Skladba ceny elektřiny. [online], [cit. 1. 5. 2011].
URL <http://www.cez.cz/cs/pro-zakazniky/faktury-a-platby/jak-se-sklada-cena-elektriny.html>
- [3] Free_Software_Foundation: GNU General Public License. [online], c2007 [cit. 25. 5. 2011].
URL <http://www.gnu.org/licenses/gpl.html>
- [4] GRUNDL, D.: Nette framework. [online], c2008 [cit. 20. 11. 2010].
URL <http://nette.org>
- [5] HUNTER, M.: Tidying the House: The MVPC Software Design Pattern. [online], c2006 [cit. 25. 5. 2011].
URL <http://www.martinhunter.co.nz/articles/MVPC.pdf>
- [6] KOSEK, J.: *PHP – Tvorba interaktivních internetových aplikací*. Praha: Grada Publishing, 1999, ISBN 80-7169-373-1.
- [7] MySQL: MySQL manual 5.5. [online], c2011 [cit. 25. 5. 2011].
URL <http://dev.mysql.com/doc/refman/5.5/en/>
- [8] PHP: PHP manual. [online], c2011 [cit. 25. 5. 2011].
URL <http://cz.php.net/manual/en/>
- [9] PHPFrameworks.com: PHP Frameworks. [online], c2007 [cit. 18. 11. 2010].
URL <http://www.phpframeworks.com/>

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

ACL	Access Control List
AJAX	Asynchronous JavaScript and XML
CAPTCHA	Completely Automated Public Turing test to tell Computers and Humans Apart
CSRF	Cross-site request forgery
DRY	Don't repeat yourself
ERÚ	Energetický regulační úřad
IP	Internet Protocol
KISS	Keep it short and simple
MVC	Model-View-Controller
MVP	Model-View-Presenter
MWh	Megawatthodina
NT	Nízký tarif
ODBC	Open Database Connectivity
ORM	Object-relational mapping
OTE	Operátor trhu s elektřinou
PDF	Portable Document Format
PDO	PHP Data Objects
PHP	PHP: Hypertext Preprocessor
RSS	Really Simple Syndication
SHA1	Secure Hash Algorithm
SQL	Structured Query Language
SVG	Scalable Vector Graphics
VT	Vysoký tarif
XSS	Cross-site scripting

SEZNAM PŘÍLOH

A	Ukázky webového systému	35
A.1	Online verze	35
A.2	Schéma databáze	36
A.3	Adresářová struktura	37
B	Obsah přiloženého CD	38

A UKÁZKY WEBOVÉHO SYSTÉMU

A.1 Online verze

adresa URL: <http://vut.kashpi.cz>

URL do administrace: <http://vut.kashpi.cz/admin>

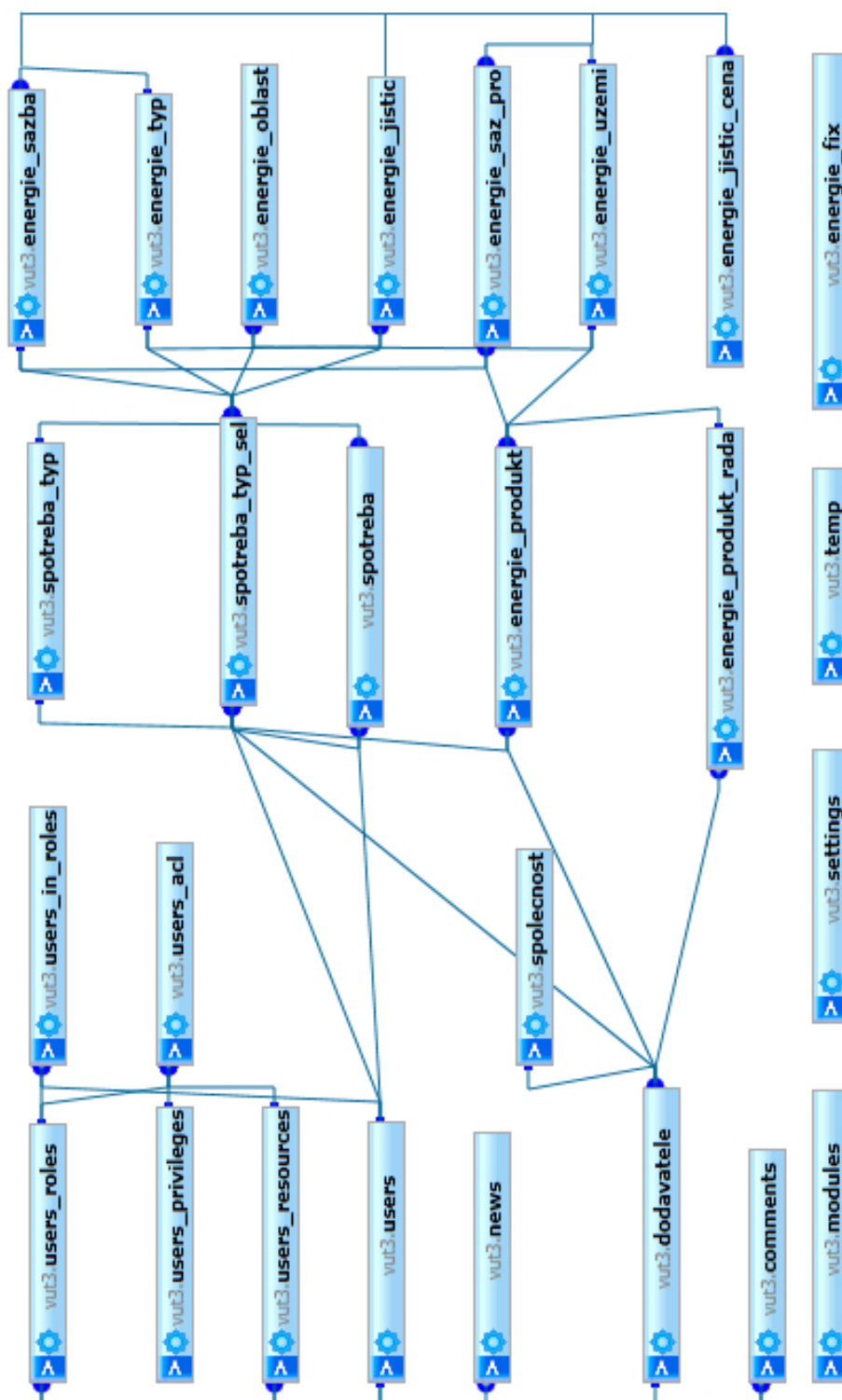
Uživatel v roli admin:

- email: admin@admin.cz
- heslo: admin

Uživatel bez role (role „registrovaný“)

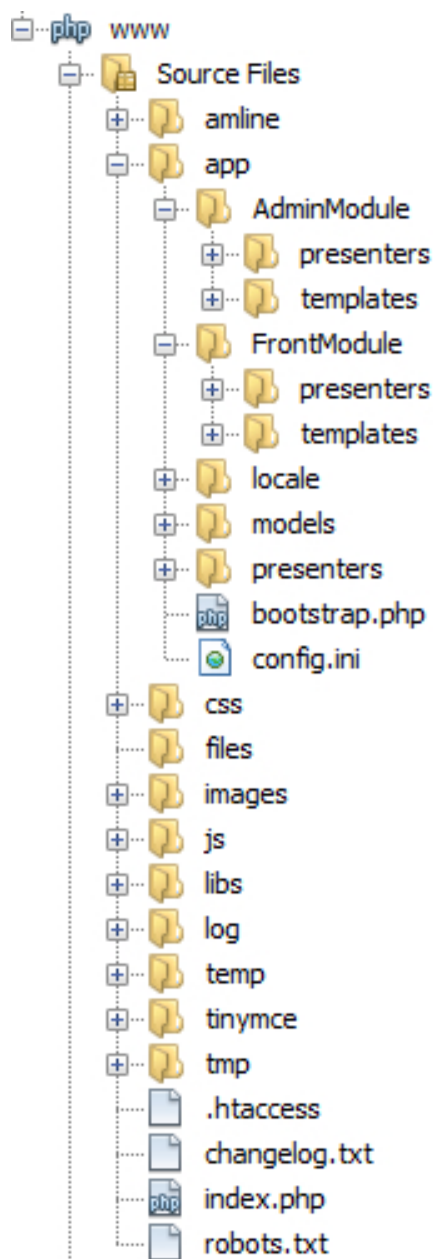
- email: user@user.cz
- heslo: user

A.2 Schéma databáze



Obr. A.1: Databáze

A.3 Adresářová struktura



Obr. A.2: Adresářová struktura

B OBSAH PŘILOŽENÉHO CD

Na přiloženém CD je adresář **www** se zdrojovými kódy webového systému a adresář **mysql** se skriptem pro vytvoření databáze. V kořenovém adresáři je soubor s textem bakalářské práce.

V adresáři **www/app** je nutné upravit soubor **config.ini** tak, aby se webová aplikace mohla připojit k databázovému serveru MySQL.

Celý systém byl vyvíjen a testován na webovém serveru Apache 2.2.17 s rozšířením PHP 5.3.5. Jako databázový server byl použit MySQL 5.5.8.