

Pastabin

Martin Beránek

28. prosince 2015

Obsah

1	Předmluva autora	3
2	Úvod	4
3	Funkční specifikace	5
3.1	Datový model	5
3.2	Uživatelské role	5
3.3	Návrh uživatelského rozhraní	5
4	Technický specifikace	7
4.1	Relační model	7
4.2	Architektura aplikace	8
4.2.1	Pohledy	8
5	Závěr	10
A	Reference pojmů	11

Seznam obrázků

3.1	Datový model	5
3.2	Úvodní obrazovka aplikace	6
3.3	Úvodní obrazovka aplikace	6
4.1	Vrstvy frameworku Nette	7
4.2	EER reprezentace entit	7
4.3	Entity vrstvy modelů	8
4.4	Entity vrstvy prezentérů	8

Kapitola 1

Předmluva autora

Aplikace je volnou kopií úspěšného portálu Pastebin (<http://pastebin.com/>). Důvodem, proč jsem si vybral její zpracování je, že všechny portály jsou zdánlivě napsané. Žijeme v době, kdy není vhodné přicházet s vynálezy kola znovu a znovu. Portály se kupují a následně přizpůsobují zákazníkovi. Proto jsem si vybral zdánlivě jednoduchou webovou aplikaci.

V zadání také bylo, že je nutné vytvořit strukturu uživatelů. Je nutné si uvědomit, že dnes není záhodno schraňovat informace o uživatelích. Daleko vhodnější je delegovat správu uživatelů na portály (Facebook, Google, ...), které jsou tomu určeny a používat přístup k nim pomocí sjednoceného **API**.

Kapitola 2

Úvod

Pro tuto práci byl vybrán portál, který má za úkol schraňovat cosi, čemu se dá jednoduše říkat paste. To může být volně vložený text pomocí klávesové kombinace `CTRL+C` `CTRL+V`. Portál může pasty uchovávat a následně je mohou uživatelé dále prohlížet.

Kapitola 3

Funkční specifikace

V této části nejdete specifikaci funkce portálu. Není zde popis technické realizace portálu. Ta je až v technické specifikaci.

Portál musí umět pracovat s uživatelskými vstupy. Pro tyto účely uchovává základní databázi uživatelů – ty mohou být registrovaní, či náhodní. Každý uživatel může vložit paste, tedy text. Portál jej uloží a předá uživateli odkaz, který může dál sdílet.

3.1 Datový model

Datový model předpokládá jednoduchou relaci mezi uživatelem a jejich pasty. Nepsaným pravidlem je, že anonymní paste je přiřazen uživateli anonym, který je dále funkčně specifikován. Pasty pak obsahují text (na obrázku 3.1 jako obsah). U uživatele je zaznamenáváno jedinečné jméno, které jej identifikuje.



Obrázek 3.1: Datový model

Relace mezi entitami slouží k vyjádření vztahu uživatelů a jejich pastů. Každý uživatel může mít bezpočet pastů. Na druhou stranu každý paste je jednoznačně přiřazen k uživateli.

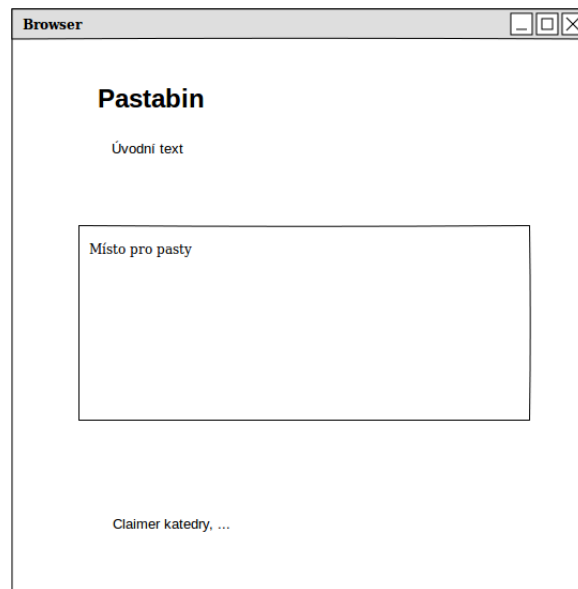
3.2 Uživatelské role

Jak již bylo zmíněno výše, aplikace obsahuje dvě uživatelské role. Jednou z nich je anonymní role uživatele. Ta slouží pro náhodné vkladatele pastů. Oproti tomu existuje registrovaný uživatel, který je uchováván v databázi aplikace.

Uživatel je autentizován pomocí jména a hesla. Od té chvíle je autorizován ke vkládání časově neomezených pastů.

3.3 Návrh uživatelského rozhraní

Uživatelské vychází z následujících Wireframů (ty jsou vytvořeny v aplikaci Pencil [1]):



Obrázek 3.2: Úvodní obrazovka aplikace

Wireframe obsahuje jen hrubý nárys aplikace. Není v něm zaneseno okénko pro přihlášení. Po vložení svého pastu se uživatel přenesse do druhého okna aplikace. To už obsahuje samotný paste.



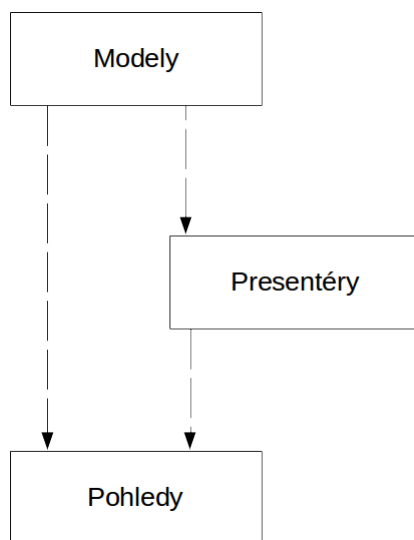
Obrázek 3.3: Úvodní obrazovka aplikace

Kapitola 4

Technický specifikace

V následující části textu jsou rozebrány technické prvky realizace portálu.

Pro vytvoření portálu na straně serveru byl použit jazyk PHP. S přihlédnutím ke kvalitnější metodice vývoje je použit framework **Nette**. Pomocí frameworku jsou vytvořeny nativní tři vrstvy aplikace (obrázek 4.1). Nette používá relaxovaný model, takže je možné posloupnost vrstev porušit. V tomto projektu se tak však nestalo.

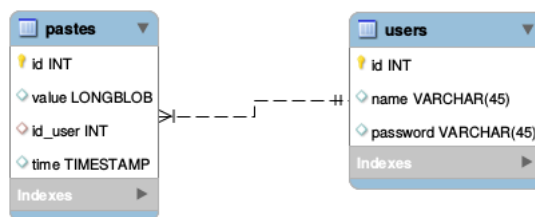


Obrázek 4.1: Vrstvy frameworku Nette

Obsah vrstev je rozebrán v následujícím textu. V modelech jsou umístěny objekty mapující tabulky databáze. V prezentérech je připravován obsah pro uživatele, který je následně zobrazen pomocí pohledů. V jiné literatuře se můžeme setkat s pojmem kontrolér, **Nette** ale používá výraz prezentér [2].

4.1 Relační model

Pro uchování dat portálu byla použita databáze MySQL. MySQL používá vlastní návrhář entit, ten se jmenuje MySQL WorkBench. Pomocí tohoto nástroje je vytvořen digram EER zobrazující vztah mezi entitou uživatelů a jejich pasty.



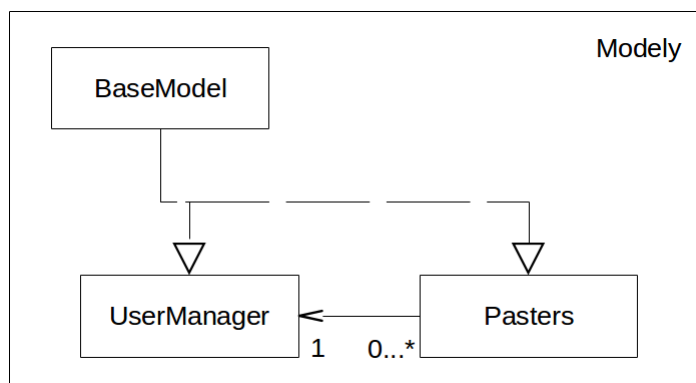
Obrázek 4.2: EER reprezentace entit

Jak je vidět z digramu, vazba mezi uživateli a pasty je volná. Může existovat paste, který nikomu nepatří. Stejně tak může existovat uživatel, který nemá žádný paste.

Za zmínku stojí, že velikost pastů je omezena maximální velikostí datového typu `LONGBLOB`. Paste může být tedy doopravdy veliký. Při každém vložení paste je zaznamenán čas pomocí datové struktury `TIMESTAMP`. Čas zapisuje databáze pomocí následující `Default` hodnoty:

```
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
```

Datový model je dále namapován na modely v architektuře aplikace. Ta je následující:



Obrázek 4.3: Entity vrstvy modelů

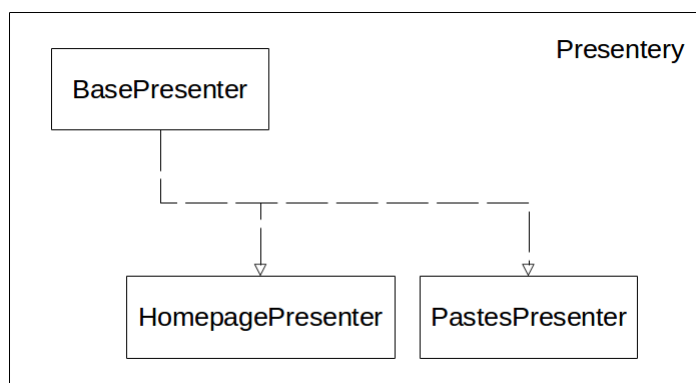
`BaseModel` je entita, která má za úkol vyvolat společné atributy následných tříd. Obsahuje tedy volání na databázi.

Za zmínku stojí třída `UserManager`, která spravuje třídy pro autentizaci, autorizaci a záznam uživatelů. Všechny ostatní entity jsou už vestavěny ve frameworku `Nette`. Proto, ačkoliv nikde není reprezentace modelu uživatel, prezenty pracují s entitou uživatel, která je volána právě z třídy `UserManager`.

Třída `Pastes` mapuje entitu `pastes` z datového modelu.

4.2 Architektura aplikace

Hlavní částí aplikace jsou prezenty. Ty mají za úkol připravit obsah portálu pro uživatele.



Obrázek 4.4: Entity vrstvy prezentérů

Jak můžeme vidět, `BasePresenter` slouží zase jen k zavedení společných částí entit prezentační vrstvy.

`HomepagePresenter` slouží k reprezentaci funkcionality čelní stránky aplikace. Pokrývá většinu práce s uživatelem. Oproti tomu `PastesPresenter` pracuje už pouze s pasty na stránce.

4.2.1 Pohledy

Aplikace pracuje s pohledy, které jsou odvozeny z prezentérů. To znamená, že pro každý prezenter existuje aspoň jeden pohled. `Nette` pro vytváření pohledů používá jazyk `Latte`. Společné vlastnosti pohledů jsou v pohledu `@layout.latte`. V tom jsou rozvrženy pomocí bloků, které se mění podle toho, jak je prezenter nastaví.

V následující ukázce je ukázán celý layout stránky:

```

{**
 * @param string    $basePath web base path
 * @param array     $flashes  flash messages
 *}

<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">

<title>{ifset title}{include title|striptags} | {/ifset}Nette Sandbox</title>
<link rel="stylesheet" href="{ $basePath}/css/bootstrap.min.css">
<link rel="stylesheet" href="{ $basePath}/css/own.css">
<link rel="shortcut icon" href="{ $basePath}/favicon.ico">
<meta name="viewport" content="width=device-width">
{block head}/{/block}
</head>

<body>

    <div class="container">
<div class="jumbotron">
{ifset title_page}{include title_page}/{/ifset}
</div>

<div n:foreach="$flashes as $flash" n:class="flash, $flash->type">{$flash->message}</div>
<div class="row marketing">
{include content}
</div>
    <footer class="footer">
<p>
Tato práce vznikla jako semestrální úloha z předmětu "Tvorba www aplikací".
</p>
</footer>
    </div>
{block scripts}
<script src="//code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="//nette.github.io/resources/js/netteForms.min.js"></script>
<script src="{ $basePath}/js/main.js"></script>
<script src="{ $basePath}/js/bootstrap.js"></script>
<script src="{ $basePath}/js/own.js"></script>
{/block}
</body>
</html>

```

Všimněte si vestavěného volání knihoven **JQuery** a použití základního stylování pomocí framework **Bootstrap**.

Kapitola 5

Závěr

Portál je koncepčně poměrně jednoduchý. Velkým problémem je zvolená metodika. Na jednu stranu je kvůli použití frameworků usnadněn následný rozvoj aplikace. Na druhou však je vývoj o to složitější, protože samotný framework není tak intuitivní jako nativní PHP.

Pro hlubší pochopení portálu doporučuji projít dokumentaci vygenerovanou pomocí `Doxygen`.

Celý projekt je na portálu `GitHub` <https://github.com/Nemo112/pastabin>.

Příloha A

Reference pojmů

Termín	popis
Paste	Vložený text pomocí CTRL+C CTRL+V.
Odkaz	V tomto textu chápáno podle standardu URL.
EER	Enhanced entity-relationship model.
Framework	Struktura pomáhající při vývoji aplikace.

Literatura

- [1] Evolus: Pencil Project. [cit. 2015-12-23]. Dostupné z: <http://pencil.evolus.vn/>
- [2] Foundation, N.: Seznámení s Nette Frameworkem. [cit. 2015-12-27]. Dostupné z: <https://doc.nette.org/cs/2.3/getting-started>