



Jak se dělá sociální síť

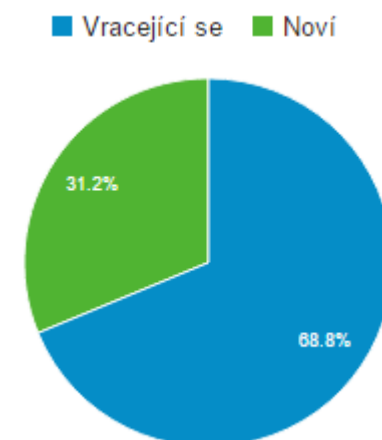
Část 1- Implementace

Implementace, Startup, Motivace

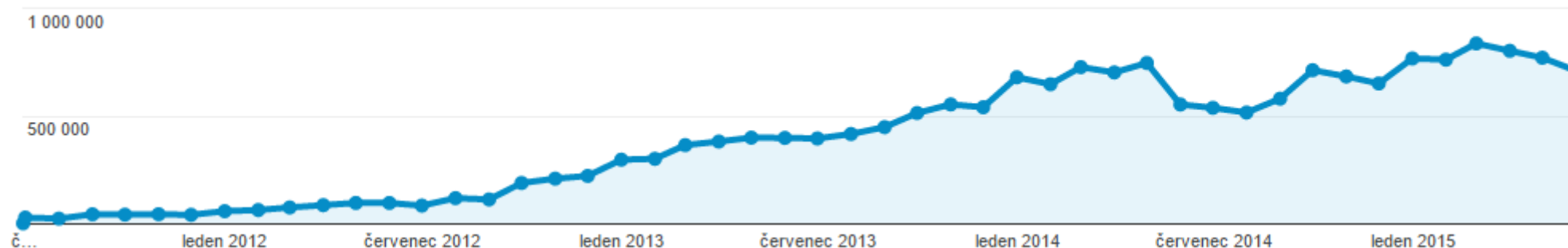
Statistiky

Všichni je milujeme :)

- Téměř 1.000.000 vygenerovaných stránek měsíčně
- 60.000 unikátních uživatelů měsíčně
 - 26.000 registrovaných členů
 - 6.000 aktivních účtů za poslední měsíc
- 4.000 článků
 - 1.900 výukových = 7200 normostran > 30 knih
 - Z toho jsem autorem 800 článků
- 7 minut trvá průměrná návštěva



● Zobrazení stránek



Implementace

```
<?php
```

```
namespace App\System\Controllers;
```

```
use ...
```

```
/** Předek pro kontrolery ...*/
```

```
abstract class Controller
```

```
{
```

```
    const MSG_INFO = 'message-info';
```

```
    const MSG_SUCCESS = 'message-success';
```

```
    const MSG_ERROR = 'message-error';
```

```
/** Pole, sloužící k předání proměnných od modelu pohledu ...*/
```

```
    protected $data = [];
```

```
/** Název pohledu (případně i s cestou) bez přípony ...*/
```

```
    protected $view = '';
```

```
/** Pole vnořených kontrolerů ...*/
```

```
    public $innerControllers = [];
```

```
/** @var string Název metody, která se má zavolat v případě, že není nalezená metoda
```

```
    protected $missingMethodName = '';
```

```
/** Rekurzivně proměnnou pro šablonu pomocí funkce htmlspecialchars ...*/
```

```
    protected function specialize($x = null)
```

```
{
```

```
        if (!isset($x))
```

```
            return null;
```

```
        if (is_string($x))
```

```
            return StringUtils::specialize($x);
```

Technologie

- Sociální sítě
- HTTP
- PHP
- PhpStorm
- Simplex PHP framework
- MySQL
- Taxy!
- Font Awesome
- jQuery



Sociální sítě

- Epocha internetu
- Vytváří komunity, spojují lidi
- Uspokojování sociální potřeby kromě informací
- Princip n^2
- Weby, které by mohly být sociální sítě, ale nejsou?
- Potenciál zvýšení návštěvnosti
- Nákladná implementace



Statistiky

Keep it coming...

- **30.000** řádků vlastního **PHP** a **SQL**
 - 200 tříd
 - 30 jmenných prostorů
- **60** databázových tabulek
- **7.000** řádků **HTML**
 - 130 šablon
- **7.000** řádků **JavaScriptu**
- **4.000** řádků **CSS**
 - 30 souborů

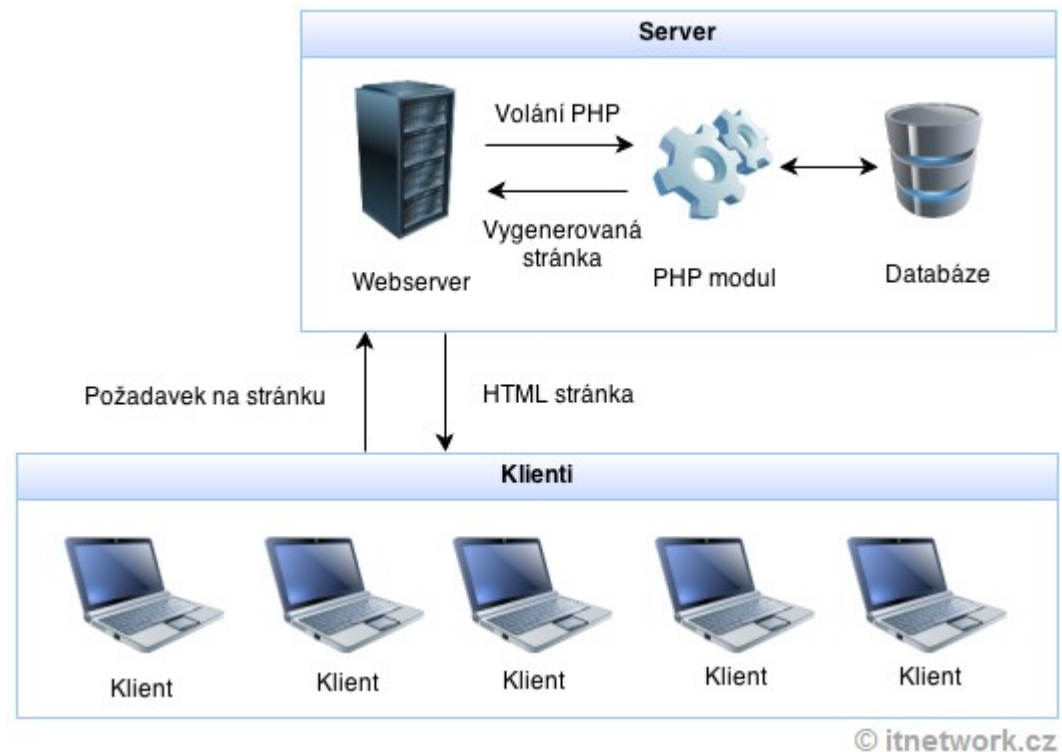


HTTP protokol

- Desktopová aplikace

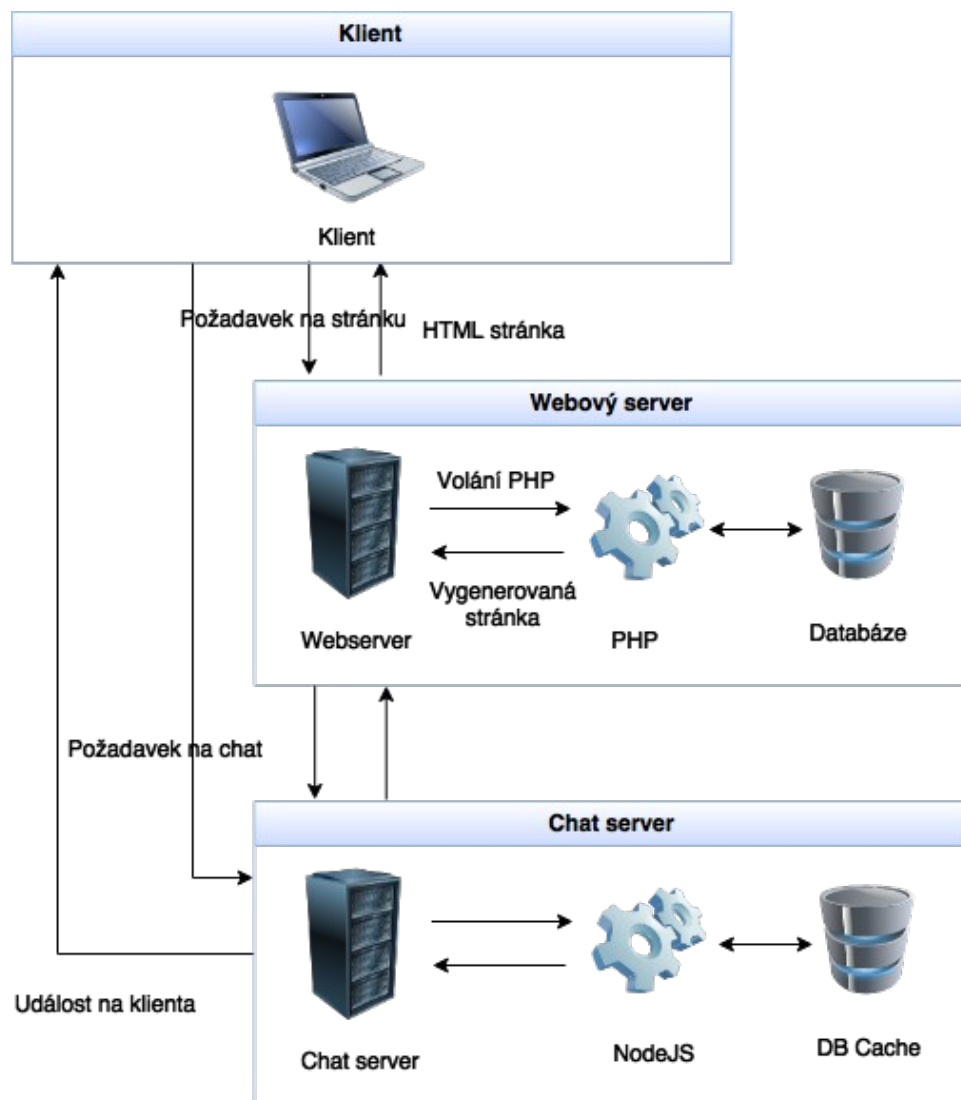


- Webová aplikace



Obcházení HTTP protokolu

- websockets



Návrh a implementace
Drahomír Hanák

MySQL



- Nejpoužívanější (=nejlevnější vývoj)
- Pokročilé featury jako transakce, cizí klíče, fulltext...
- Budoucnost?
- Objektové databáze

Texy!

- WYSIWYG
 - Velký boom → zklamání
 - Sémantika
 - Jednotnost stylů
 - Validace
 - Bezpečnost
 - Firemní zákazníci
- WYSIWYM
 - Weby, které to myslí vážně (Wikipedia), knihy (LaTeX), známé blogy
 - 100% výsledek
 - Absence náhledu
- Kombinace = win-win scenario



By David Grudl

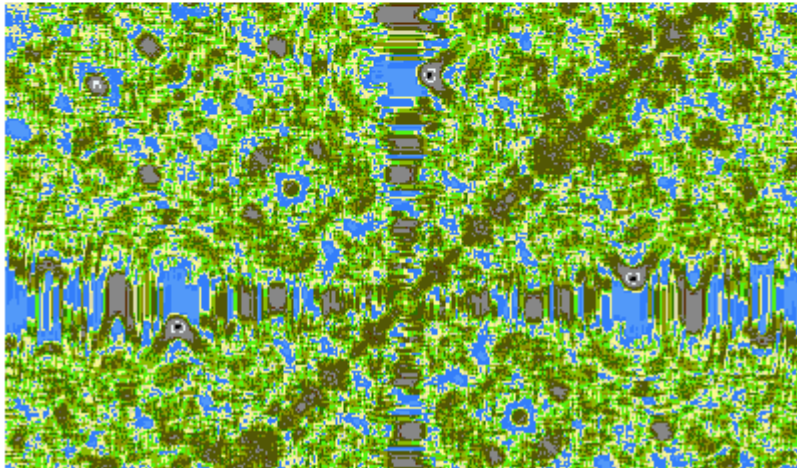
WYSIWYG

Algoritmus

Tenhle týden jsem však objevil *algoritmus*, který dokáže vyrobit velmi zajímavou mapu s několika příjemnými vlastnostmi:

- Mapa je **nekonečná**
- V každé větší části (cca 50 x 50) se nachází všechny biotopy – jezera, tráva, lesy a hory.
- Mapa není nezajímavý šum ani neustále opakovaný vzor, ale různě velké a prolínající se kruhové, eliptické a hyperbolické útvary.
- Generování mapy je na výkon velmi nenáročné.

Zde je ukázka [mapy](#). Výseky jsou 400 x 400 políček:



WYSIWYM

Algoritmus

#####

Tenhle týden jsem však objevil **algoritmus**, který dokáže vyrobit velmi zajímavou mapu s několika příjemnými vlastnostmi:

- Mapa je ***nekonečná***
- V každé větší části (cca 50 x 50) se nachází všechny biotopy - jezera, tráva, lesy a hory.
- Mapa není nezajímavý šum ani neustále opakovaný vzor, ale různě velké a prolínající se kruhové, eliptické a hyperbolické útvary.
- Generování mapy je na výkon velmi nenáročné.

Zde je ukázka "mapy":<http://www.mapy.cz>. Výseky jsou 400 x 400 políček:

[* <http://www.itnetwork.cz/images/9049/prime1.png> *]

WYSIWYG + WYSIWYM

Algoritmus

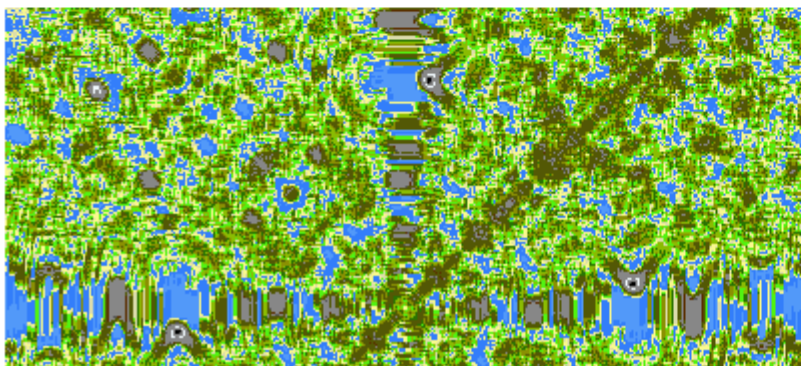
#####

Tenhle týden jsem však objevil **algoritmus**, který dokáže vyrobit velmi zajímavou mapu s několika příjemnými vlastnostmi:

- Mapa je ****nekonečná****
- V každé větší části (cca 50 x 50) se nachází všechny biotopy - jezera, tráva, lesy a hory.
- Mapa není nezajímavý šum ani neustále opakovaný vzor, ale různě velké a prolínající se kruhové, eliptické a hyperbolické útvary.
- Generování mapy je na výkon velmi nenáročné.

Zde je ukázka "mapy":<http://www.mapy.cz>. Výseky jsou 400 x 400 políček:

[* <http://www.itnetwork.cz/images/9049/prime1.png> *]



Originální editor vyvinul
Michal Žůrek

Font awesome a jQuery

- Problém JavaScriptu
- jQuery
- Upouštíme od jQuery UI



- Obrázkové fonty
 - Škálovatelné
 - Jednolitě
 - Univerzální
 - Font Awesome, Twemoji Awesome



Best practices

Programování není o implementaci, ale o návrhu

- KISS
 - API, fluent interface
- SRP
 - SoC, MVC, MVP, MVVM...
- DRY
- IHP
- Další: IoC, DI, LoD
- SOLID = SRP + OCP + LSP + ISP + DIP
- Návrhové vzory, GOF, GRASP

Volba nástroje

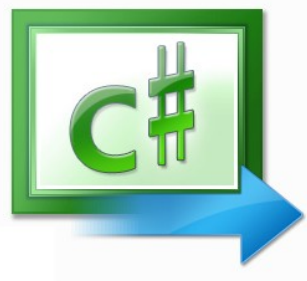
- Každý znak v kódu stojí peníze a to jednorázově i paušálně



```
XMLOutputFactory xof = XMLOutputFactory.newInstance();
XMLStreamWriter xsw = null;
try
{
    xsw = new IndentingXMLStreamWriter(xof.createXMLStreamWriter(new
    FileWriter("soubor.xml")));

    ...
```

Mnoho tříd v Javě navíc nepodporuje TWR, což zvyšuje objem zbytečného kódu obrovským způsobem



```
XmlWriterSettings settings = new XmlWriterSettings();
settings.Indent = true;

using (XmlWriter xw = XmlWriter.Create(@"soubor.xml", settings))
...

```


Gettery/settery v Javě

```
class Student
{
```

```
    private String jmeno;
    private boolean muz;
    private int vek;
    private boolean plnolety;
```

```
    public String getJmeno()
    {
        return jmeno;
    }
```

```
    public void setJmeno(String jmeno) {
        this.jmeno = jmeno;
    }
```

```
    public boolean isMuz() {
        return muz;
    }
```

```
    public int getVek()
    {
        return vek;
    }
```

```
    public void setVek(int vek)
    {
        this.vek = vek;
        // přehodnocení plnoletosti
        plnolety = true;
        if (vek < 18)
            plnolety = false;
    }
```

```
    public boolean isPlnolety() {
        return plnolety;
    }
```



```
class Student
{
```

```
    public string Jmeno { get; private set; }
    public bool Muz { get; private set; }
    public bool Plnolety { get; private set; }
    private int vek;
    public int Vek
```

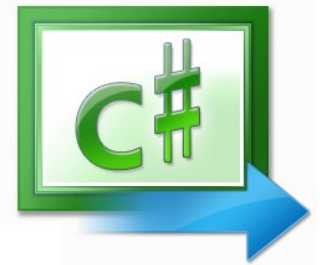
```
    {
        get
        {
            return vek;
        }
```

```
        set
        {
```

```
            vek = value;
            // kontrola plnoletosti
            Plnolety = true;
            if (vek < 18)
                Plnolety = false;
```

```
        }
```

```
    }
```



- Toto není vtip
- Javistům to nepřijde divné
- Problém neřeší scaffolding
- V PHP se veřejné atributy používají běžně a není v tom problém
 - Gettery jen kvůli interface
 - Settery kvůli fluent-api

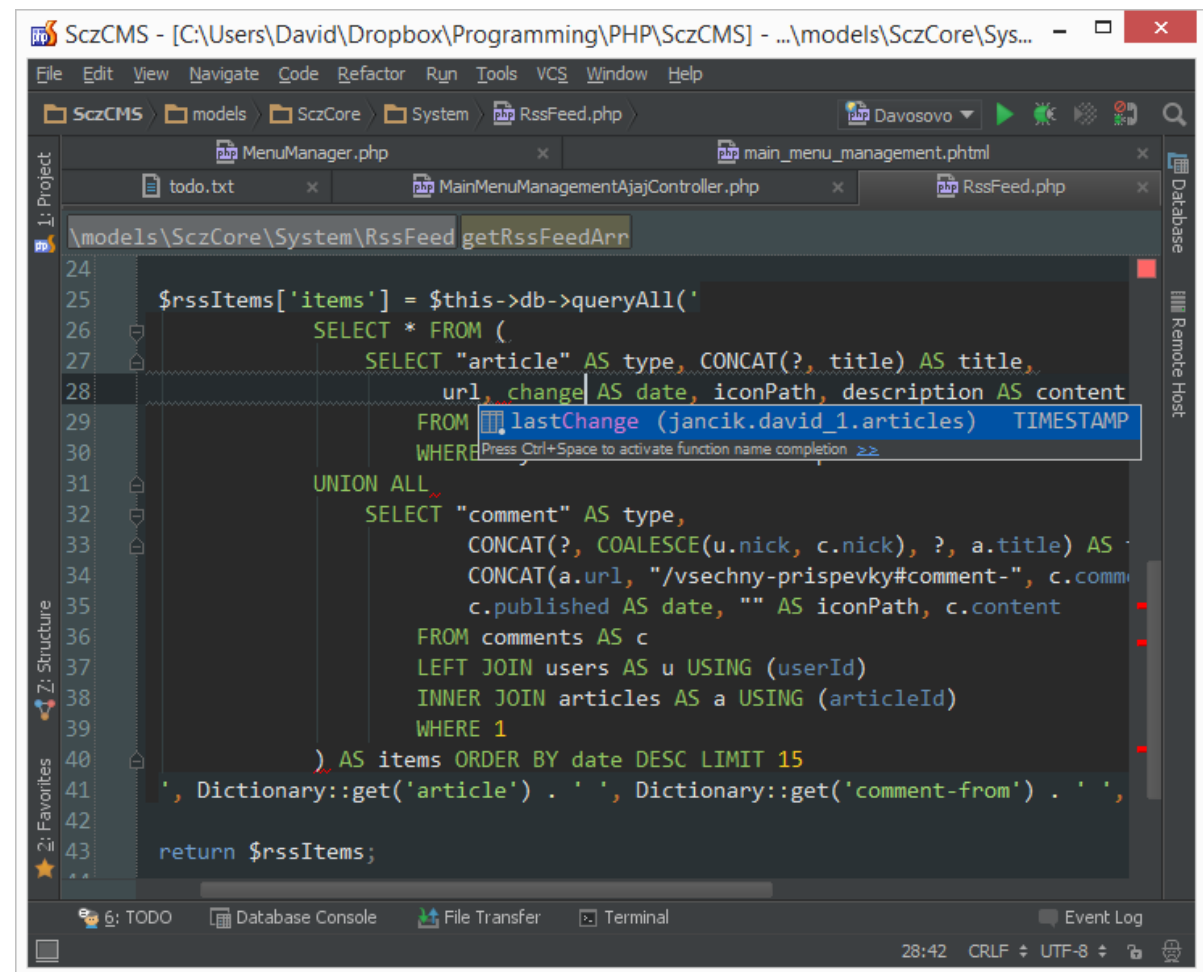
PHP



- Jednoduché
- Pohání Facebook
- Open-source (jistota)
- Nejrozšířenější (= nejlevnější vývoj)
- Vysoce stabilní vývoj a zpětná kompatibilita
- Frameworky – Proč jsou potřeba?
 - Wrappery
 - Proprietární framework vs. framework
 - Oficiální frameworky vs. komunitní frameworky

PhpStorm

- Šetří obrovské náklady
 - Náklady při vývoji
 - Náklady při opravách
 - Cena nástrojů je obecně zanedbatelná
 - Jazyků, IDE...
- Mé oblíbené featury
 - Barví a kontroluje SQL
 - Hlídá datové typy podle phpDoc
 - Rychlé vyhledání čehokoli
 - Refactoring
 - ...



Simplex for PHP

- Simplex = Simple + Complex
 - Jednoduchý framework pro tvorbu komplexních systémů
 - Jednoduchost = udržitelnost = úspěch
 - Proprietární framework, částečně dostupný v prémiových seriálech
 - Není umění vytvořit složitě funkční projekt, ale projekt roky potom udržovat
 - Vlastní vs. cizí projekt
 - Přirozený vývoj projektů, přepisy
-
- MVC
 - Formulářový framework
 - Utility třídy (wrappery na pole, datum, řetězce...)
 - Nemá šablonovací systém (extract a ob)
 - Nemá nástavbu nad SQL (PhpStorm)

Simplex kontroler

```
class CampaignController extends Controller
{
```

```
/**
 * @Action
 */
```

```
public function index()
```

```
{
```

```
    $this->auth('marketing_manager');
    $campaignManager = new CampaignManager();
    $this->data['campaigns'] = $campaignManager->getCampaigns();
    $this->view = 'index';
```

```
}
```

```
/** @ApiAction ... */
```

```
public function click($bannerId)
```

```
{
```

```
    $bannerManager = new BannerManager();
    // Rozděleno na 2x kvůli rychlosti
    $url = $bannerManager->getUrl($bannerId);
    header("Location: " . $url);
    header("Connection: close");
    if ($url)
        $bannerManager->addClick($bannerId);
```

```
}
```

Obsluha formuláře (add/edit)

```
/** @Action ...*/  
public function manage($campaignId = null)  
{  
    $this->auth('marketing_manager');  
  
    $campaignManager = new CampaignManager();  
    $form = $campaignManager->getCampaignForm();  
  
    if ($campaignId)  
    {  
        $campaign = $campaignManager->getCampaign($campaignId);  
        if (!$campaign)  
        {  
            $this->redirect('error');  
            $form->setData($campaign);  
        }  
        // Odeslání formuláře  
        if ($form->isPostBack())  
        {  
            try  
            {  
                $data = $form->getData();  
                $campaignId = $campaignManager->saveCampaign($data);  
                $this->addMessage('Kampaň byla úspěšně uložena.', self::MSG_SUCCESS);  
                $this->redirect('reklamni-kampane/detail/' . $campaignId);  
            }  
            catch (UserException $ex)  
            {  
                $this->addMessage($ex->getMessage(), self::MSG_ERROR);  
            }  
        }  
        $this->data['form'] = $form;  
        $this->view = 'manage';  
    }  
}
```

Model

*/** Vrátí formulář pro správu kampaní ...*/*

```
public function getCampaignForm()
{
    $form = new Form('campaign_form');
    $form->addHiddenBox('campaign_id');
    $form->addDatePicker('starts', 'Začátek', true);
    $form->addDatePicker('ends', 'Konec', true);
    $form->addTextBox('title', 'Titulek', true);
    $form->addNumberBox('clicks', 'Prokliků')
        ->setText(0);
    $form->addNumberBox('max_clicks', 'Limit prokliků')
        ->setText(0);
    $form->addButton('save', 'Uložit');
    return $form;
}
```

*/** Uloží kampaň jako novou nebo aplikuje změny podle campaign_id ...*/*

```
public function saveCampaign($campaign)
{
    if (!$campaign['campaign_id'])
    {
        Db::insert('campaign', $campaign); // vkládáme novou
        return Db::lastInsertId();
    }
    Db::update('campaign', $campaign, 'WHERE campaign_id = %i', $campaign['campaign_id']); // editace
    return $campaign['campaign_id'];
}
```

*/** Odstraní kampaň s daným Id ...*/*

```
public function removeCampaign($campaignId)
{
    Db::query('DELETE FROM campaign WHERE campaign_id = %i', $campaignId);
}
```

PHTML šablona

```
<?php if ($campaigns) : ?>
    <table class="fancytable">
        <thead>
            <tr>
                <th>Od</th>
                <th>Do</th>
                <th>Titulek</th>
                <th>Prokliků</th>
                <th>Aktivní</th>
                <th></th>
            </tr>
        </thead>
        <tbody>
            <?php foreach ($campaigns as $campaign): ?>
                <tr>
                    <td><?= FormatHelper::prettyDate($campaign['starts']) ?></td>
                    <td><?= FormatHelper::prettyDate($campaign['ends']) ?></td>
                    <td><a href="reklamni-kampane/detail/<?= $campaign['campaign_id'] ?>"><?= $campaign['title'] ?></a><
                    <td>
                        <?= $campaign['clicks'] ?>
                        <?php if ($campaign['max_clicks']) : ?> / <?= $campaign['max_clicks'] ?><?php endif ?>
                    </td>
                    <td class="<?= ($campaign['active'] ? 'paid' : 'unpaid') ?>">
                        <?= FormatHelper::boolean($campaign['active']) ?>
                    </td>
                    <td>
                        <a href="reklamni-kampane/manage/<?= $campaign['campaign_id'] ?>">
                            <i class="fa fa-pencil"></i>
                        </a>
                        <a href="reklamni-kampane/delete/<?= $campaign['campaign_id'] ?>" data-confirm="Opravdu si přeje
                            <i class="fa fa-times"></i>
                        </a>
                </td>
            </tr>
        </tbody>
    </table>
</?>
```


Vyrenderovaný pohled index

Reklamní kampaně

localitnetwork/reklamni-kampane

ITnetwork.cz
Sociální síť pro IT profesionály

227

2

2

David Čápka

Hardware

Vyhledávání...

f

rss

Reklamní kampaně

[Zpět na hlavní stranu](#)

CZ
125 Kč

HOSTING NAPLNO A ZDARMA.

VYZKOUŠET NA 60 DNŮ ZDARMA

onebit®

Nová kampaň

Od	Do	Titulek	Prokliků	Aktivní	
1. června	31. prosince	OneBit	820	Ano	
22. června	26. června	Sraz	162	Ne	
7. června	21. června	Facebook	814	Ne	
1. června	18. června	Letní C#	929	Ne	

Editovat článek

Copyright © 2015 localitnetwork. Veškerý obsah webu (pokud není uvedeno jinak) spadá pod autorský zákon. [O projektu](#) | [Reklama](#) | [Loga a bannery](#) | Stránka vygenerována za 177 milisekund.

Projekt localitnetwork vytvořil student vysoké školy [Unicorn College](#).

0

Vyrenderovaný pohled manage

```
<?= $form->render() ?>
```

Reklamní kampaně

[Zpět na hlavní stranu](#)

HOSTING NAPLNO A ZDARMA.

onebit®

VYZKOUŠET NA 60 DNŮ ZDARMA

Začátek	1.6.2015
Konec	31.12.2015
Titulek	OneBit
Prokliků	820
Limit prokliků	0

Uložit

Editovat článek

Copyright © 2015 localitnetwork. Veškerý obsah webu (pokud není uvedeno jinak) spadá pod autorský zákon. [O projektu](#) | [Reklama](#) | [Loga a bannery](#) | Stránka vygenerována za 168 milisekund.

Projekt localitnetwork vytvořil student vysoké školy Unicorn College.

Zajímavé implementace

- Člákové kontrolery (viz seriál)
- Událostní model
 - Reprezentace v DB, registr, deserializace, upomínání a nastavení upomínání, chat
- Soukromé zprávy
- Účetnictví
- Proces schvalování
- Skloňování českých jmen (Jan Vargovský)
- ...

Děkuji za pozornost

A držím palce s vašimi projekty!



P.S. 28.8.2015 se uvidíme u 2. části - Startup