

# **Dokumentace projektu**

## **Webové aplikace**

Jan Javor 3.B

10.10.2020

## Instalace a spuštění

Ke spuštění projektu, je nutné mít zapnuté servery Apache a MySQL. Po stažení souboru si celý soubor uložíme do xampp/htdocs stáhnutou databázi do xampp/mysql/data. Klikneme na config u serveru Apache a klikneme na první možnost, najdeme si řádek DocumentRoot a přepíšeme ho na cestu k souboru projektu(obr. č. 1). Poté je potřeba napsat do vašeho webové prohlížeče napsat URL „localhost“ což vás přesměruje na úvodní stránku projektu. Pokud prohlížeč hlásí chybu, je nutné si v souboru database.php nastavit heslo(obr. č. 2).

Č.1

```
| DocumentRoot "C:/xampp/htdocs/kina/public"
```

Č.2

```
'password' => '',
```

## Controller

V mém projektu se nachází jeden controller s názvem Home.php s následujícími funkcemi:

```
use App\Models\VypisFilmuModel;  
use App\Models\VypisSaluModel;  
use App\Models\VypisVstupenekModel;
```

V úvodní části controlleru si naimportujeme model.

```
public function index()  
{  
    $filmy = new VypisFilmuModel();  
    $data['filmy'] = $filmy->findAll();  
    $this->ionAuth = new \IonAuth\Libraries\IonAuth();  
    if(!$this->ionAuth->loggedIn())echo view('head');  
    else echo view('headprihlaseny');  
    echo view('uvod');  
}
```

V této funkci naimportujeme data z databáze, zobrazíme meníčko, podle toho zda je uživatel přihlášený nebo ne, a úvodní stranu

```
public function vypis_filmu(){  
    $this->ionAuth = new \IonAuth\Libraries\IonAuth();  
    if(!$this->ionAuth->loggedIn())echo view('head');  
    else echo view('headprihlaseny');  
    $filmy = new VypisFilmuModel();  
    $data['filmy'] = $filmy->findAll();  
    echo view('vypis_filmu', $data);  
}
```

Funkce vypis\_filmu načte menu stránky, získá data z databáze, ty si uloží do proměnné, načte stránku s výpisem filmů s uloženými daty.

```

public function vypis_salu(){
    $this->ionAuth = new \IonAuth\Libraries\IonAuth();
    if(!$this->ionAuth->loggedIn())echo view('head');
    else echo view('headprihlaseny');
    $saly = new VypisSaluModel();
    $data['saly'] = $saly->findAll();
    echo view('vypis_salu', $data);

}

```

Funkce vypis\_salu načte menu stránky, získá data z databáze, ty si uloží do proměnné, načte stránku s výpisem sálů s uloženými daty.

```

public function vypis_vstupenek(){
    $this->ionAuth = new \IonAuth\Libraries\IonAuth();
    if(!$this->ionAuth->loggedIn())echo view('head');
    else echo view('headprihlaseny');
    $prodeje = new VypisVstupenekModel();
    $data['prodeje'] = $prodeje->findAll();
    echo view('vypis_vstupenky', $data);

}

```

Funkce vypis\_vstupenek načte menu stránky, získá data z databáze, ty si uloží do proměnné, načte stránku s výpisem vstupenek s uloženými daty.

```

public function pridat_film(){
    $this->ionAuth = new \IonAuth\Libraries\IonAuth();
    if(!$this->ionAuth->loggedIn())echo view('head');
    else echo view('headprihlaseny');
    echo view('pridat_film');

}

public function pridat_sal(){
    $this->ionAuth = new \IonAuth\Libraries\IonAuth();
    if(!$this->ionAuth->loggedIn())echo view('head');
    else echo view('headprihlaseny');
    echo view('pridat_sal');

}

public function pridat_vstupenku(){
    $this->ionAuth = new \IonAuth\Libraries\IonAuth();
    if(!$this->ionAuth->loggedIn())echo view('head');
    else echo view('headprihlaseny');
    echo view('pridat_vstupenku');

}

```

Tyhle funkce zajišťují zobrazování stránek s formulářem pro přidání filmu, sálu či vstupenky.

```

public function zapsat(){
    $filmy = new VypisFilmuModel();
    $data = [
        'nazev_cz' => $this->request->getPost('nazev_cz'),
        'originalni_nazev' => $this->request->getPost('originalni_nazev'),
        'delka' => $this->request->getPost('delka'),

        'id_zanru_filmu' => $this->request->getPost('id_zanru_filmu'),
        'zeme_id_zeme' => $this->request->getPost('zeme_id_zeme'),
        'id_promitani' => $this->request->getPost('id_promitani'),

    ];
    $filmy->save($data);
    return redirect()->to(base_url('vypis_filmu'));
}

public function zapsatSal(){
    $saly = new VypisSaluModel();
    $data = [
        'cislo_salu' => $this->request->getPost('cislo_salu'),
        'typ_promitani' => $this->request->getPost('typ_promitani'),
        'typ_ozvuceni' => $this->request->getPost('typ_ozvuceni'),

    ];
    $saly->save($data);
    return redirect()->to(base_url('vypis_salu'));
}

public function zapsatVstupenku(){
    $vstupenky = new VypisVstupenekModel();
    $data = [
        'cas_prodeje' => $this->request->getPost('cas_prodeje'),
        'cena_vstupenky' => $this->request->getPost('cena_vstupenky'),
        'misto_v_sale' => $this->request->getPost('misto_v_sale'),
        'id_salu_id' => $this->request->getPost('id_salu_id'),
    ];
    $vstupenky->save($data);
    return redirect()->to(base_url('vypis_vstupenky'));
}

```

Tyto funkce zajišťují, aby se data vyplněné ve formuláři zapsaly do databáze, zase pro každou tabulku zvlášť funkce.

```

public function uprava($id = null){
    $filmy = new VypisFilmuModel();
    $data['filmy'] = $filmy->find($id);
    $this->ionAuth = new \IonAuth\Libraries\IonAuth();
    if(!$this->ionAuth->loggedIn())echo view('head');
    else echo view('headprihlaseny');
    return view('uprava', $data);
}

public function upravaSalu($id = null){
    $saly = new VypisSaluModel();
    $data['saly'] = $saly->find($id);
    $this->ionAuth = new \IonAuth\Libraries\IonAuth();
    if(!$this->ionAuth->loggedIn())echo view('head');
    else echo view('headprihlaseny');
    return view('upravaSalu', $data);
}

public function upravaVstupenky($id = null){
    $vstupenky = new VypisVstupenekModel();
    $data['prodeje'] = $vstupenky->find($id);
    $this->ionAuth = new \IonAuth\Libraries\IonAuth();
    if(!$this->ionAuth->loggedIn())echo view('head');
    else echo view('headprihlaseny');
    return view('upravaVstupenky', $data);
}

```

Tyto funkce mají na starost, aby při kliknutí na úpravu cviku se zobrazila stránka na úpravu filmu, sálu nebo vstupenky. Mají také na starost to, aby tam byla předvyplněné informace o daném filmu, sálu nebo vstupence.

```

public function zapsatUpravu($id = null){
    $filmy = new VypisFilmuModel();
    $data = [
        'nazev_cz' => $this->request->getPost('nazev_cz'),
        'originalni_nazev' => $this->request->getPost('originalni_nazev'),
        'delka' => $this->request->getPost('delka'),

        'id_zanru_filmu' => $this->request->getPost('id_zanru_filmu'),
        'zeme_id_zeme' => $this->request->getPost('zeme_id_zeme'),
        'id_promitani' => $this->request->getPost('id_promitani'),
    ];
    $filmy->update($id, $data);
    return redirect()->to(base_url('vypis_filmu'));
}

public function zapsatUpravuSalu($id = null){
    $saly = new VypisSaluModel();
    $data = [
        'cislo_salu' => $this->request->getPost('cislo_salu'),
        'typ_promitani' => $this->request->getPost('typ_promitani'),
        'typ_ozvuceni' => $this->request->getPost('typ_ozvuceni'),
    ];
    $saly->update($id, $data);
    return redirect()->to(base_url('vypis_salu'));
}

public function zapsatUpravuVstupenky($id = null){
    $vstupenky = new VypisVstupenekModel();
    $data = [
        'cas_prodeje' => $this->request->getPost('cas_prodeje'),
        'cena_vstupenky' => $this->request->getPost('cena_vstupenky'),
        'misto_v_sale' => $this->request->getPost('misto_v_sale'),
        'id_salu_id' => $this->request->getPost('id_salu_id'),
    ];
    $vstupenky->update($id, $data);
    return redirect()->to(base_url('vypis_vstupenek'));
}

```

Funkce zapsatUpravu, zapsatUpravuSalu, zapsatUpravuVstupenky mají na starost, aby když změníte při úpravě nějaké informace, aby se změny zapsaly do databáze



```

public function smazat($id = null){
    $filmy = new VypisFilmuModel();
    $filmy->delete($id);
    return redirect()->to(base_url('vypis_filmu'));
}

public function smazatSal($id = null){
    $saly = new VypisSaluModel();
    $saly->delete($id);
    return redirect()->to(base_url('vypis_salu'));
}

public function smazatVstupenku($id = null){
    $vstupenky = new VypisVstupenekModel();
    $vstupenky->delete($id);
    return redirect()->to(base_url('vypis_vstupenek'));
}

```

Funkce smazat, smazatSal, smazatVstupenku slouží k tomu aby se daná věc odstranila z databáze

## Modely

V mém projektu se nachází tři modely s názvy VypisFilmuModel.php, VypisSaluModel.php a VypisVstupenekModel.php

```
class VypisFilmuModel extends Model
{
    protected $table = 'filmy';

    protected $primaryKey = 'id_filmu';
    protected $allowedFields = [
        'id_filmu',
        'nazev_cz',
        'originalni_nazev',
        'delka',

        'id_zanru_filmu',
        'zeme_id_zeme',
        'id_promitani',
    ];
}
```

V modelu VypisFilmuModel.php je zapsáno které data chceme z databáze čerpat, takhle to funguje i u následujících modelů VypisSaluModel.php a VypisVstupenekModel.php.

## Nastavení databáze v projektu

V tomto souboru je pouze nastaveno, aby se automaticky načetla databáze ,která je definována v souboru database.php

```
public $default = [  
    'DSN' => '',  
    'hostname' => 'localhost',  
    'username' => 'root',  
    'password' => '',  
    'database' => 'kina',  
    'DBDriver' => 'MySQLi',  
    'DBPrefix' => '',  
    'pConnect' => false,  
    'DBDebug' => (ENVIRONMENT !== 'production'),  
    'charset' => 'utf8',  
    'DBCollat' => 'utf8_general_ci',  
    'swapPre' => '',  
    'encrypt' => false,  
    'compress' => false,  
    'strictOn' => false,  
    'failover' => [],  
    'port' => 3306,  
];
```

Do kolonky database napíšeme jméno naší databáze, případně změníme i username a password.

## Routy

```
$routes->get('/', 'Home::index');
$routes->get('vypis_filmu', 'Home::vypis_filmu');
$routes->get('vypis_salu', 'Home::vypis_salu');
$routes->get('vypis_vstupenek', 'Home::vypis_vstupenek');
$routes->get('pridat_film', 'Home::pridat_film');
$routes->get('pridat_sal', 'Home::pridat_sal');
$routes->get('pridat_vstupenku', 'Home::pridat_vstupenku');
$routes->post('zapsat', 'Home::zapsat');
$routes->post('zapsatSal', 'Home::zapsatSal');
$routes->post('zapsatVstupenku', 'Home::zapsatVstupenku');
$routes->get('uprava/(:num)', 'Home::uprava/$1');
$routes->get('upravaSalu/(:num)', 'Home::upravaSalu/$1');
$routes->get('upravaVstupenky/(:num)', 'Home::upravaVstupenky/$1');
$routes->put('zapsatUpravu/(:num)', 'Home::zapsatUpravu/$1');
$routes->get('smazat/(:num)', 'Home::smazat/$1');
$routes->get('smazatSal/(:num)', 'Home::smazatSal/$1');
$routes->get('smazatVstupenku/(:num)', 'Home::smazatVstupenku/$1');
$routes->put('zapsatUpravuSalu/(:num)', 'Home::zapsatUpravuSalu/$1');
$routes->put('zapsatUpravuVstupenky/(:num)', 'Home::zapsatUpravuVstupenky/$1');
```

Zde máme routy, které v projektu používám, první informace v závorce je „zkratka“, kterou jsem si nastavil pro danou funkci(druhá informace v závorce)

```
$routes->group('auth', ['namespace' => 'IonAuth\Controllers'], function ($routes) {
    $routes->add('login', 'Auth::login');
    $routes->get('logout', 'Auth::logout');
    $routes->add('forgot_password', 'Auth::forgot_password');
    //$routes->add('login', 'Auth::login');
    $routes->add('create_user', 'Auth::create_user');
    //($routes->add('forgot_password', 'Auth::forgot_password'))
});
```

Zde to samé, akorát je to hozené ve skupině auth, která slouží pro přihlášení.www