

Univerzita Pardubice  
Fakulta elektrotechniky a informatiky

David Baláček

# Mateřská školka Sluníčko

*SEMESTRÁLNÍ PRÁCE*

## Obsah

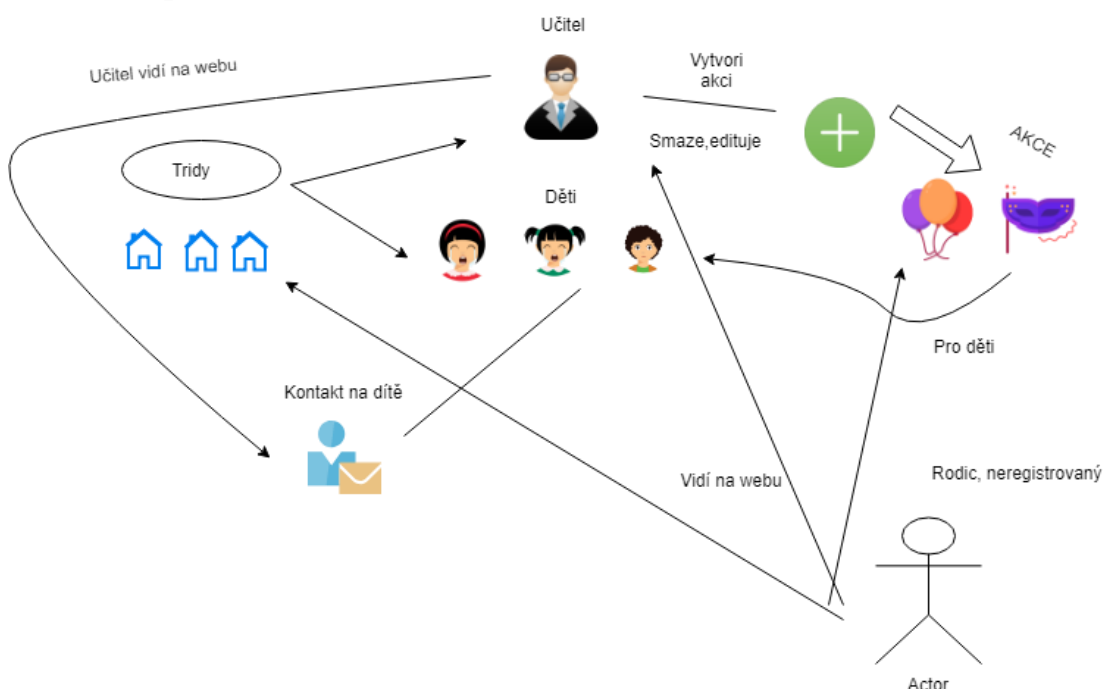
1	Úvod .....	2
1.1	Popis aplikace.....	2
1.2	Rich picture .....	3
1.3	Architektura.....	3
2	Analýza .....	3
2.1	Aktéři systému.....	3
2.2	.....	4
2.3	UML use case diagram .....	4
2.4	UML Activity diagram .....	5
2.5	Databázový model.....	6
3	Implementace.....	7
3.1	Adresářová struktura .....	7
3.2	.....	7
3.3	Ukázky zdrojového kódu .....	9

## 1 Úvod

### 1.1 Popis aplikace

Aplikace slouží jako informační systém mateřské školky. Rodiče dětí zde najdu třídy ve kterých jsou rozzařeny děti. Kontakty na zaměstnance školky, jídelníček na celý týden a plánované akce školky. Každý učitel má přidělenou třídu a spravuje si děti a má možnost vytvořit akci pro ni. Je ovšem omezen svoji třídou. Ředitel školky může editovat cokoliv včetně samotných zaměstnanců a jako dodatek vytváří týdenní jídelníček nahráním do systému.

## 1.2 Rich picture



## 1.3 Architektura

Aplikace využívá Docker, je napsána v HTML a stylována v CSS, pozadí (napojení do database, redacní system) napsán v PHP. Jako IDE využito JetBrains PhpStorm. Využívá MySQL databázi.

## 2 Analýza

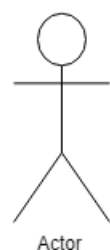
### 2.1 Aktéři systému

- Neregistrovaný uživatel – rodič, má informativní přístup ke všem složkám (zaměstnancům, třídám, dětem ve třídě, jídelníčku, akcích školy).
- Registrovaný uživatel – učitel, je přiřazený do třídy, do své třídy může přidávat nové děti, upravovat je, mazat je. Má možnost vytvořit akci pro svoji třídu.
- Administrator – ředitel školy, má přístup ke všem složkám, spravuje uživatele (učitele), má přístup ke spravování všech tříd a dětí v nich, zároveň všech akcí. Zároveň má příležitost možnost nahrát jídelníček z jsonu, smazat jej či stáhnout si jej.

## 2.2

### 2.3 UML use case diagram

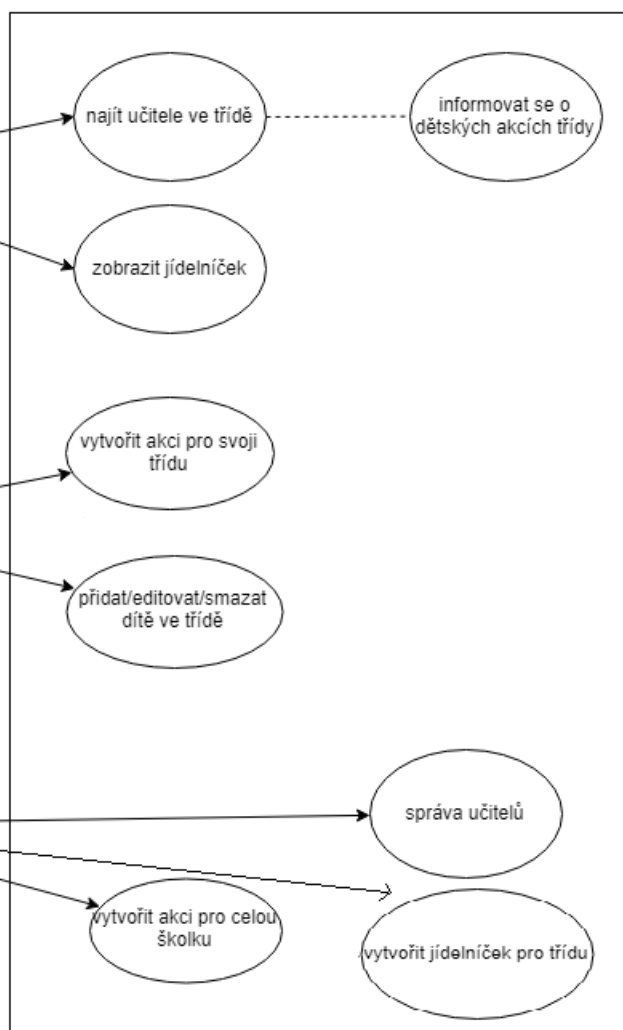
Neregistrovaný



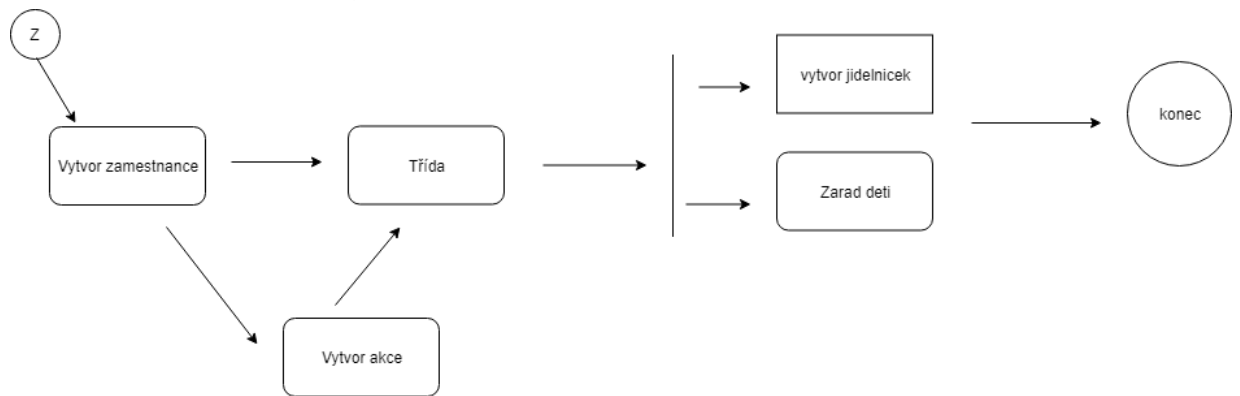
Registrovaný



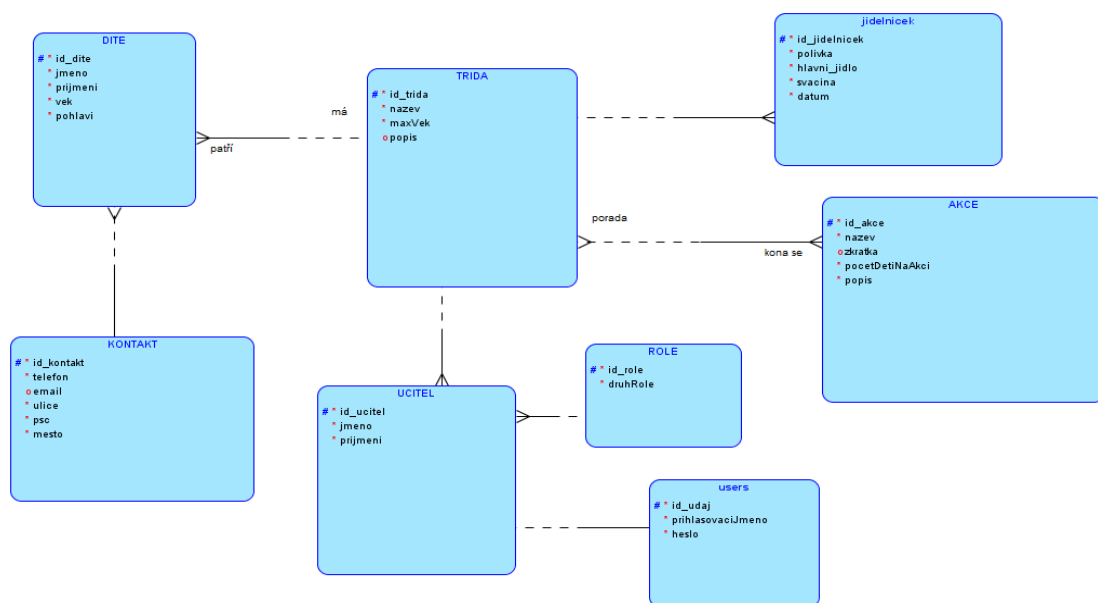
Registrovaný



## 2.4 UML Activity diagram

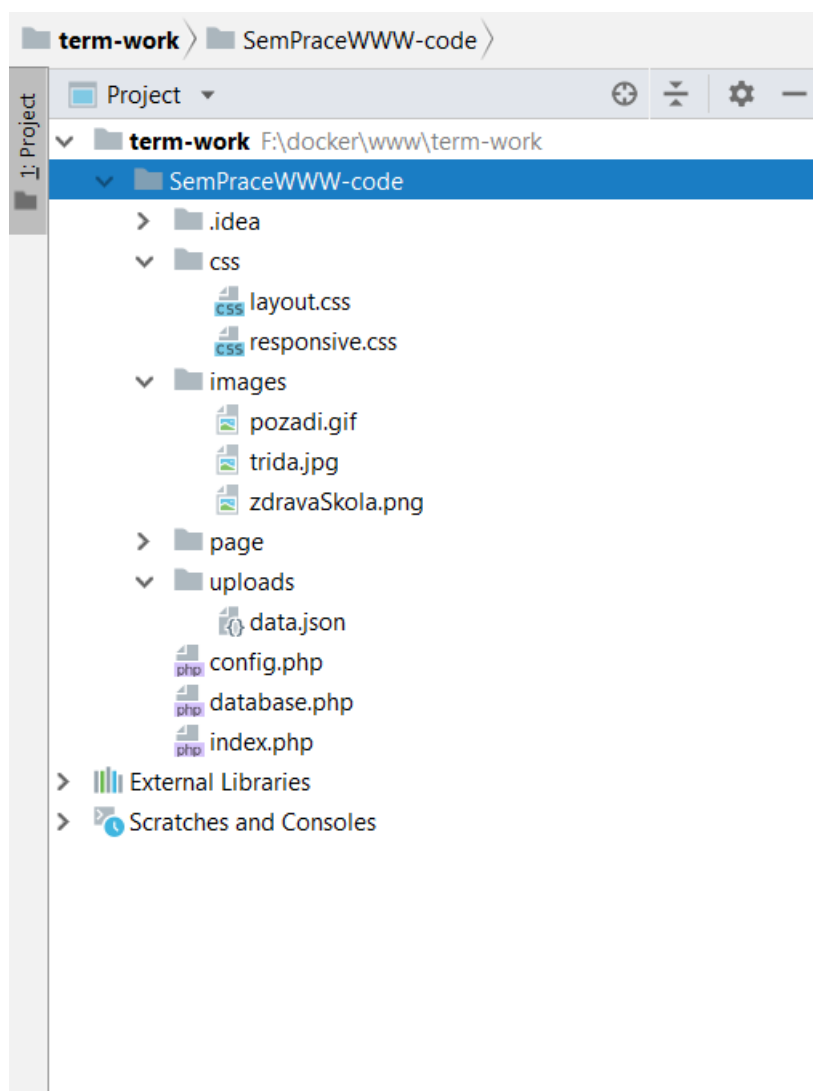


## 2.5 Databázový model

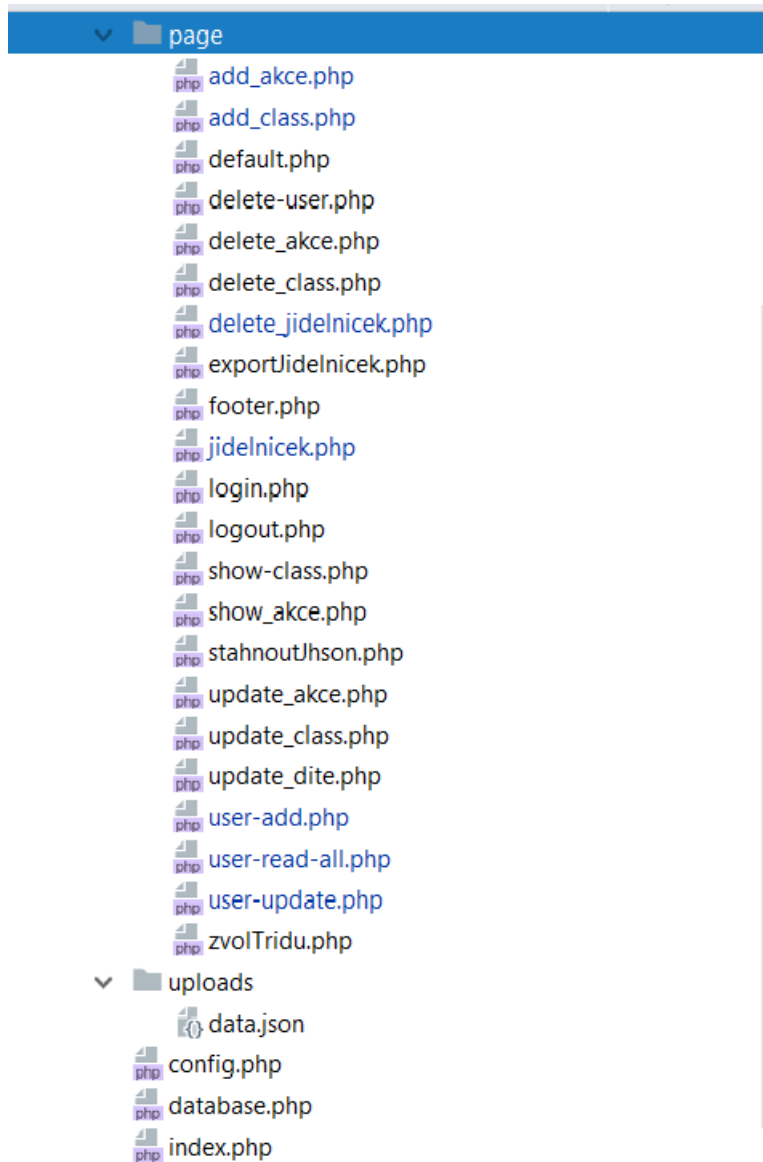


## 3 Implementace

### 3.1 Adresářová struktura



3.2





### 3.3 Ukázky zdrojového kódu

```
13
14 if (isset($_POST["buttonImport"])) {
15     $uploadaddir = './uploads/';
16     $uploadfile = $uploadaddir . basename($_FILES['jsonFile']['name']);
17     $extension = array("json", "JSON");
18     $uploadOk = true;
19
20     $ext = pathinfo($_FILES["jsonFile"]["name"], options: PATHINFO_EXTENSION);
21     if (in_array($ext, $extension) == false) {
22         $uploadOk = false;
23         echo "<p class='hlaska'>Neplatny soubor</p>";
24     }
25 }
26
27 if ($uploadOk == true) {
28     try {
29         if (move_uploaded_file($_FILES['jsonFile']['tmp_name'], $uploadfile)) {
30             $data = file_get_contents($uploadfile); // put the contents of the file into a variable
31             $obj = json_decode($data); // decode the JSON feed
32
33             foreach ($obj as $obj) {
34                 $stmt = $conn->prepare( statement: "INSERT INTO jidelnicek(polivka, hlavni_jidlo, svacina, id_trida, den)");
35                 $stmt->bindParam( parameter: ':polivka', &variable: $obj->polivka);
36                 $stmt->bindParam( parameter: ':hlavni_jidlo', &variable: $obj->hlavniJidlo);
37                 $stmt->bindParam( parameter: ':svacina', &variable: $obj->svacina);
38                 $stmt->bindParam( parameter: ':id_trida', &variable: $obj->idTrida);
39                 $stmt->bindParam( parameter: ':den', &variable: $obj->den);
40                 $stmt->execute();
41             }
42         }
43     } catch (Exception $e) {
44         echo 'Chyba: ' . $e->getMessage();
45     }
46     echo "Import proveden!";
47 }
```