



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií ■

Výukový a informační portál "elektromobilita na TUL"

Ročníkový projekt

Studijní program: B2646 – Informační technologie

Studijní obor: 1802R007 – Informační technologie

Autor práce: Miroslav Cimbora

Vedoucí práce: Ing. Pavel Jandura



TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky, informatiky a mezioborových studií

Ústav MTI

Školní rok: 2014/15

ZADÁNÍ SEMESTRÁLNÍHO PROJEKTU

Název projektu: Výukový a informační portál "elektromobilita na TUL"

Řešitel: Miroslav Cimbora

Studijní obor: Informační Technologie

Vedoucí projektu: Pavel Jandura

Zadání:

1. Návrh databáze webového portálu
2. Návrh šablony a celkového vzhledu stránky
3. Realizace redakčního systému.
4. Zabezpečení aplikace

Doporučená literatura:

- [1] PLÍVA, Zdeněk; DRÁBKOVÁ, Jindra. *Metodika zpracování diplomových, bakalářských a vědeckých prací na FM TUL*. Vyd. 1. Liberec : Technická univerzita, 2007. 40 s. Dostupné z WWW: <http://www.fm.tul.cz/files/jak_psat_DP.pdf>. ISBN 978-80-7372-189-3.
- [2] BROWN, Tiffany B, Kerry BUTTERS a Sandeep PANDA. 2014. *HTML5 okamžitě*. 1. vyd. Brno: Computer Press, 256 s. ISBN 978-80-251-4296-7.
- [3] MOLINARO, Anthony. 2009. *SQL: kuchařka programátora*. Vyd. 1. Brno: Computer Press, 573 s. ISBN 978-80-251-2617-2.

Datum zadání:

Podpis vedoucího projektu:



Poděkování

Chtěl bych poděkovat panu Jandurovi za jeho konstruktivní nápady, které pomohly k dokončení projektu a též bych chtěl poděkovat i správě TUL, která byla velmi ochotna spolupracovat a poskytla designové podklady a šablony, které TUL využívá.



Anotace

Cílem bakalářského projektu je vytvoření webového portálu, který by měl veřejnost ale hlavně studenty Technické Univerzity v Liberci informovat o dění na katedře MTI a jejich běžících projektech. V první části práce bude obsažena teoretická část, která předchází samotné tvorbě webových stránek. Následně budou probrány jednotlivé kroky při postupu programování dynamického webu.

Klíčová slova

Redakční systém, Webové stránky, Databáze, Design

Seznam zkratk



Abstract

The goal of the bachelor's project is to create a web portal, that would inform the public and most importantly the students of the TUL University about the events on the the MTI department and it's running projects. In the first part of the project there will be a theoretical part that precedes the creation of the website. After that, single steps of the procedure of programming a dynamical website will be discussed.

Key word

Content Management System, Website, Database, Design



Obsah

Seznam literatury	7
Seznam ilustrací.....	8
1 Úvod	9
2 WireFrame.....	10
3 Návrh databáze	11
3.1 Základní pojmy databází.....	11
3.2 Kardinalita je násobnost vztahu	11
3.3 MySQL.....	11
4 Webový server a PHP	13
4.1 Klient/Server.....	13
4.2 LocalHost	13
4.2 PHP	13
5 Realizace	14
5.1 MVC.....	14
5.1.1 Model	14
5.1.2 View	14
5.1.3 Controler.....	14
5.2 Obsah.....	14
5.3 Design	15
5.3.1 Šablona	15
5.4 Komunikace PHP s MySQL.....	16
5.4.1 Ukázka připojení k databázi pomocí PDO.....	16
5.4.2 Zaslání příkazu do databázového serveru	16
5.4.3 Převzetí dat.....	16
6 Vytvoření administrační zóny	17
7 Zabezpečení.....	18
7.1 Relace(Session).....	18
7.2 Ošetření dat z formulářů.	18
7.3 Základní funkce pro ošetření formulářů:.....	18
7.4 Ošetření přístupu do databáze.....	19
Závěr	20



Seznam literatury

OPPEL, Andrew J. 2008. *SQL bez předchozích znalostí: [průvodce pro samouky]*. Vyd. 1. Brno: Computer Press, 240 s. ISBN 978-80-251-1707-1.

CASTRO, Elizabeth a Bruce HYSLOP. 2012. *HTML5 a CSS3: názorný průvodce tvorbou WWW stránek*. 1. vyd. Brno: Computer Press, 439 s. ISBN 978-80-251-3733-8.

PONKRÁC, Miloslav. 2007. *PHP a MySQL: bez předchozích znalostí : [průvodce pro samouky]*. Vyd. 1. Brno: Computer Press, 221 s. ISBN 978-80-251-1758-3.



Seznam ilustrací

Obrázek 1 WireFrame.....	10
Obrázek 2 Databáze Portálu	12
Obrázek 3 Klient/Server	13
Obrázek 4 Design/šablona.....	15
Obrázek 5 Administrační rozhraní	17
Obrázek 6 Kontrola uživatele	Chyba! Záložka není definována.

Obrázek 1 WireFrame.....	10
Obrázek 2 Databáze Portálu	12
Obrázek 3 Klient/Server	13
Obrázek 4 Design/šablona.....	15
Obrázek 5 Administrační rozhraní	17



1 Úvod

Tato práce pojednává o tvorbě webové stránky s jednoduchým redakčním systémem, která bude sloužit hlavně jako zdroj informací o ústavu MTI a jejich projektech. Webové stránky by také měly poukázat na rozsáhlou škálu zkoumaných oblast s hlavním zaměřením na elektrotechnickou část katedry MTI. Práce by též měla podat studentům stručné informace o katedře jako takové a seznámit s vozidlem, které katedra MTI vyvíjí. Touto cestou rovnou i vozidlo představit veřejnosti a tím pádem ho i prezentovat jakožto výtvor ústavu.

Druhou částí práce je samotné vytvoření webového portálu se zaměřením na vytvoření vlastního redakčního systému. Na internetu je rozsáhlá škála freeware redakčních systémů stejně tak i komerčních, je však zajímavé zkusit implementovat vlastní systém, který je jednoduchý a zároveň plnohodnotný.



2 WireFrame

Každá stránka obsahuje základní webové prvky: záhlaví webu, navigace stránky, patičku, logo hlavní menu a další. Rozmístění těchto elementů je klíčové a musí být takové, aby se uživatel snadno na webové stránce orientoval. Pro stránku je velmi důležité, aby nebyla přehlčená a nepřehledná. Uživatel by měl dosáhnout svých požadavků na pár kliknutí. Pokud je toho velmi jednoduché pravidlo porušeno je jisté, že se uživatel na Vaši stránku už nikdy nevrátí.

Nadměrné užívání reklam a různým rušivých aspektů často uživateli znepráhledňuje orientaci na webu. K odstranění těchto nedostatků složí právě webový wireframe, který by nám měl pomoci si uvědomit, o jaký web se jedná a tím lépe vystihnout uživateli nároky a usnadnit orientaci na webových stránkách



Obrázek 1 WireFrame



3 Návrh databáze

Velmi důležitou vlastností dynamických webových aplikací a redakčních systému je jejich propojení s databází. Databáze nám slouží k uchovávání dat a následnou práci s nimi.

Jako první je důležité si uvědomit, jaká data chceme uchovávat v databázi. Proto musíme vytvořit kompletní analýzu a shromáždit všechna dostupná data a informace. Při návrhu nám může velmi pomoci ER- Diagram. Zachycuje hlavní nároky kladené na databázi a slouží ke zpětné vazbě od zákazníka.

3.1 Základní pojmy databází

Entita je základní stavební prvek databáze. Každá entita zastupuje širší množinu informací, které mají podobný význam. Každá entita má své vlastnosti (atributy). Atribut představuje konkrétní vlastnost entity což, může být např. pro entitu člověk atribut rodné číslo, které zároveň determinuje každou instanci entity „člověk“.

3.2 Kardinalita je násobnost vztahu

1:1 Instance entity odpovídá právě jedné instanci druhé entity

1:N Instance entity odpovídá N instancí druhé entity

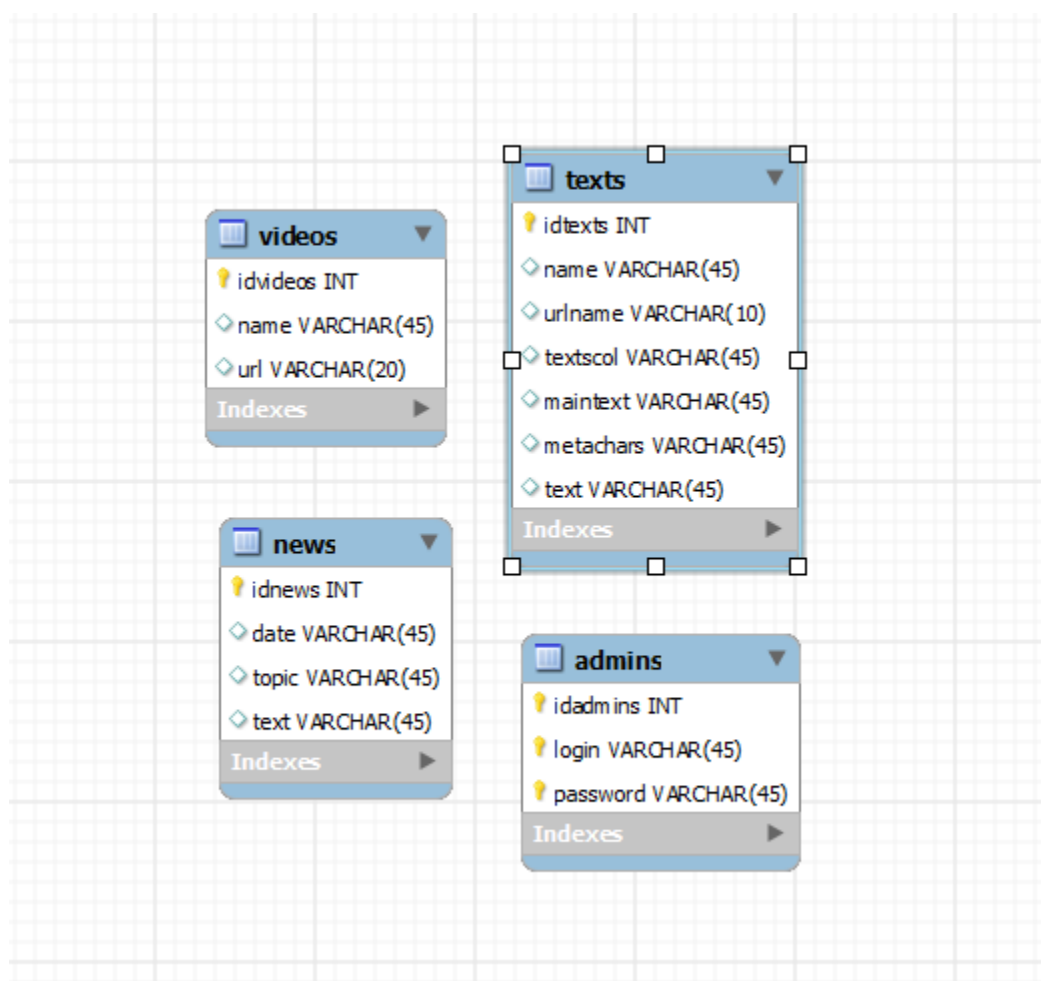
M:N M entit odpovídá N instancí druhé entity

Druhou důležitou vlastností je též parcialita, která definuje, zda se musí entita nacházet v nějakém vztahu. Rozlišujeme dva druhy vztahu totální a parciální.

3.3 MySQL

je databázový open-source server, který se rozšířil zejména při vytváření webových aplikací. Jeho hlavní výhodou je vysoká stabilita a hlavně nenáročnost na zdroje počítače.





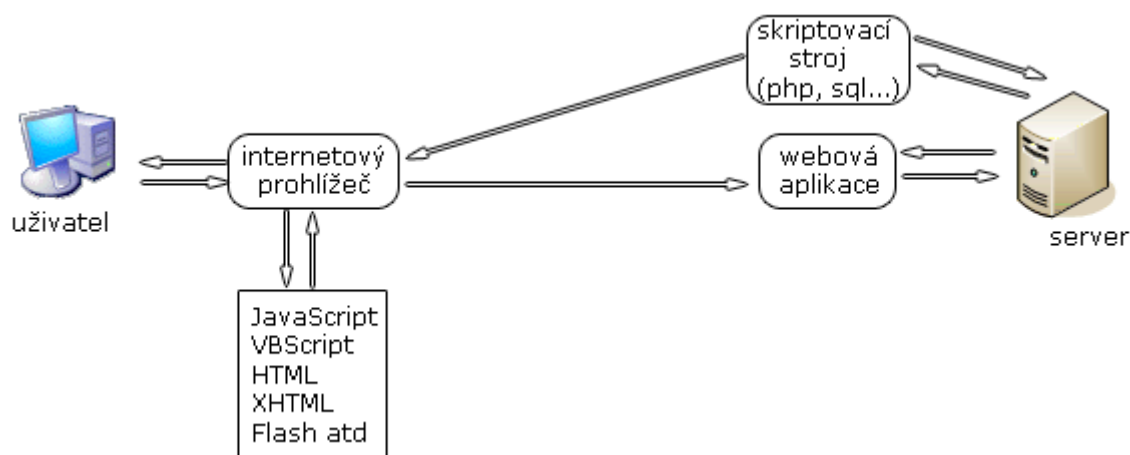
Obrázek 2 Databáze Portálu



4 Webový server a PHP

4.1 Klient/Server

Webové aplikace fungují na principu klient/server. Klient zasílá požadavky (request) na server a následně na ně server odpovídá v podobě odpovědi (response).



Obrázek 3 Klient/Server

4.2 LocalHost

Při vývoji web aplikací je velmi užitečné používat tzv. lokální servery, které běží na počítači vývojáře, tzn, že lokální počítač se chová přesně jako server. Requesty se posílají na lokální počítač a tím se tak simuluje reálná situace v podobě klient/server. Vývojář je tedy schopný si ihned zobrazit své výsledky bez nutnosti aktualizace stránek na webový hosting.

4.3 PHP

Je nejrozšířenější serverový skriptovací jazyk, který umožňuje tvořit dynamické webové aplikace.



5 Realizace

Celá aplikace je v podstatě rozdělena do dílčích funkcí, které řeší jednotlivé úlohy aplikace.

5.1 MVC

MVC je softwarová šablona pro řešení rozsáhlejších projektů. MVC rozděluje aplikaci na tři celistvé vrstvy

5.1.1 Model

Model obsahuje logiku a vše, co do ní spadá. Mohou to být výpočty, databázové dotazy, validace a podobně. Model vůbec neví o výstupu. Jeho funkce spočívá v přijetí parametrů zvenku a vydání dat ven.

5.1.2 View

Pohled se stará o zobrazení výstupu uživateli. Nejčastěji se jedná o html šablonu, obsahující HTML stránku a tagy značkovacího jazyka.

5.1.3 Controler

Jedná se o jakéhosi prostředníka, se kterým komunikuje uživatel, model i view. Drží tedy celý systém pohromadě a komponenty propojuje.

Velmi důležité je též oddělení HTML kódu od kódu PHP, výsledný kód je pak lépe čitelný a mnohem snáze se ladí a hledají chyby.

5.2 Obsah

Portál používá pouze jednu jedinou HTML šablonu, která zůstává stejná pro všechny podstránky webu. Dynamická je pouze část s obsahem, která se mění na základě získaného url odkazu. PHP skript získá informaci o zadané URL a zavolá funkci, která změní obsah dané stránky.

`localhost/projekt/index.php?page=kontakt`

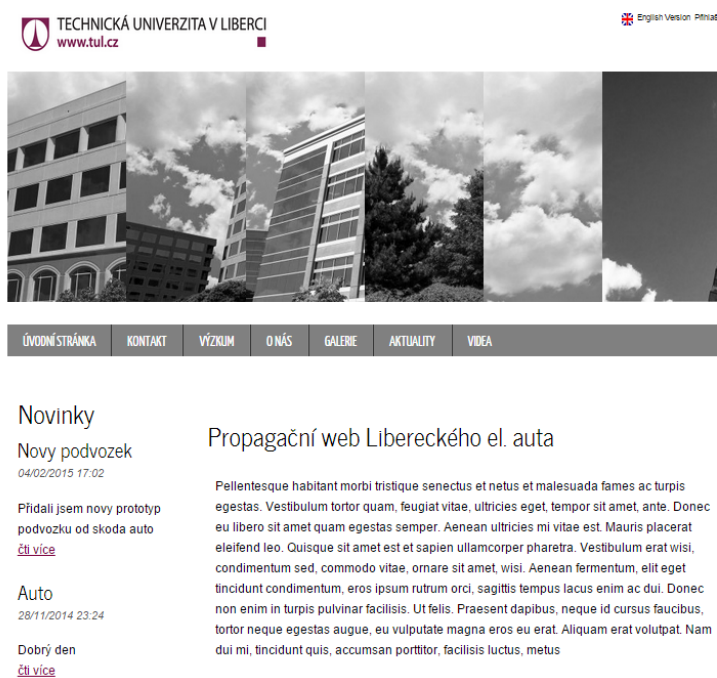
Pro ukládání textu na stránkách jsou použity textové soubory, nejsou však uloženy v databázi. Jelikož se jedná o prezentační web jehož texty budou takřka statické, je zbytečné je ukládat do databáze a tím i zpomalit celou aplikaci. Texty jsou načítány pomocí skriptu, který podle url adresy naplní proměnou požadovaným textem a ten je posléze na stránce zobrazen. Texty lze rovněž v administrativní zóně měnit mazat a různě upravovat.



5.3 Design

Další nedílnou součástí webových aplikací je jejich vzhled. Pokud se uživatel rozhodne kliknout na Vaši stránku, první věc na kterou narazí je právě design. Špatný Design totiž může docílit toho, že se uživatel na Vaši stránku už nikdy nevrátí. Proto je velmi důležité si vymezit hlavní funkci stránek a na základě toho vytvořit design, který by přesně vystihl funkci dané stránky.

5.3.1 Šablona



Obrázek 4 Design/šablona

Po konzultaci bylo rozhodnuto o střídavém vzhledu v barvách technické univerzity ale v důsledku unifikace a dodržení nového formátu TUL webových stránek, byla šablona změněna na stávající vzhled, který přesně vystihuje danou funkci stránek a navíc přidává stránce větší dojem, že se nachází v celém konceptu webových stránek TUL.



5.4 Komunikace PHP s MySQL

Při práci s databází vznikají čtyři základní operace.

- 1) připojení k databázi
- 2) zaslání příkazu databázovému serveru
- 3) převzetí dat
- 4) odpojení od databáze

Pro připojení k databázi se používá mnoho rozhraní. V aplikaci je použito PDO. V případě jeho správného použití rovněž zabraňuje útoku SQL injection.

5.4.1 Ukázka připojení k databázi pomocí PDO

Pomocí php skriptu musíme specifikovat konkrétní databázový server. Musíme určit jeho adresu, typ, název databáze, popřípadě heslo.

5.4.2 Zaslání příkazu do databázového serveru

Při se celé řadě operací je nutno skripty zasílat dotazy na db. server. Hlavně však při vytváření redakčního systému, kde se neustále pracuje s databází.

```
$sql = "INSERT INTO aktuality (datum,nazev,text) VALUES (:datum,:nazev,:obsah)";
```

ukázka vložení aktuality do databáze

5.4.3 Převzetí dat

Druhů převzetí dat je hned několik. MySQL vrací výsledek v podobě tzv. object result. Samotný object result neobsahuje požadovaná data, ale je jakýmsi prostředníkem jak data získat. Pokud chceme získat konkrétní data, můžeme například použít funkci fetch assoc, ta nám vrátí výsledek v podobě asociativního pole.

```
$zaznam = $ok->fetch(PDO::FETCH_ASSOC)
```

Ke konkrétním datům se následně můžeme dostat takto:

`$zaznam['nazev']` parametr název pak představuje konkrétní sloupec z tabulky.



6 Vytvoření administrační zóny

Srdcem dynamických webových stránek je vytvoření administrační zóny, která primárně slouží administrátorovi webových stránek k jeho editaci a správě. Na internetu existuje spousta redakčních systémů, které už poskytují nepřeberné množství funkcí i metod. Z těch nejznámějších stojí za zmínku např. WordPress a Drupal. Jejich rozšíření a obliba z nich dělají jedny z nejpoužívanějších redakčních systémů používaných při tvorbě webových stránek.

Redakční systémy jsou velmi užitečné, zejména pak pro uživatele, kteří nejsou seznámeni s tvorbou webových stránek. Umožňují jim spravovat webové stránky pohodlně bez zásahu do kódu stránek.

Cílem však bylo implementovat si vlastní jednoduchý redakční systém, který měl umožnit:

- Správu aktualit
- Editace hlavního menu
- Změna hesla
- Editace a správa galerie
- Správu obsahu webu

The screenshot shows a web administration interface titled "Administrace". At the top, there is a navigation bar with the following links: Home, Aktuality, Texty, Změna hesla, Video, Galerie, and Logout. The main content area contains a form for adding a new news item. The form has two labels: "Název zprávy:" followed by a text input field, and "Obsah zprávy:" followed by a large text area. Below the text area is a button labeled "Přidat". At the bottom of the interface, there is a footer bar with the word "Administrace" centered.

Obrázek 5 Administrační rozhraní



7 Zabezpečení

Dalším důležitým aspektem webových stránek je jejich zabezpečení. Pod pojmem zabezpečení webových stránek se skrývá zejména ošetření všech možných dat, která přijímáme.

7.1 Relace(Session)

Je speciální druh superglobální proměnné (\$_SESSION), která vzniká vždy při spojení se serverem, na kterém jsou naše stránky umístěny. Proměnná Session je typu pole podobně jako proměnná (\$_POST). Session nám umožňuje překonat slabiny bezstavovosti protokolu http tzn. že si dokážeme například uchovat data, zda je již uživatel přihlášen a nebudeme znovu vyžadovat heslo při přechodu na další podstránku.

7.2 Ošetření dat z formulářů.

Hlavní důležitou součástí zabezpečení je ošetření všech vstupů, které jsou zadávány od uživatele, nejčastěji však z webových formulářů. Uživatel může úmyslně nebo omylem zadat chybová data, která při špatném zpracování mohou způsobit pád celé aplikace.

7.3 Základní funkce pro ošetření formulářů:

```
function html($string){  
    return htmlspecialchars($string);  
}
```

Funkce htmlspecialchars() převádí jakýkoliv text, který začíná „<“ na obyčejný string, tím docílíme toho, že se např. do naší databáze neuloží kus HTML kódu, který by mohl ovlivnit zobrazení našich stránek.

```
function trimm($string){  
    return trim($string);  
}
```

Funkce trim() ořízne z řetězce všechny bílé znaky.



7.4 Ošetření přístupu do databáze

K této problematice se váže pojem „SQL injection“ což je druh napadení webových a databázových aplikací. Jedná se o vložení části SQL kódu do aplikace a tím získat přístup k databázi nebo v horším případě ji celou odstranit.

Ochranou pro tento druh útoku jsou tzv. „Prepared Statements“. Doslova si data si připravíme, zkontrolujeme a následně uložíme do databáze. Tím zabráníme potenciálnímu útočníkovi vložit nebezpečný kód do naší aplikace.

Prepared Statements je řešení, při kterém se nejprve vytvoří SQL dotaz, ve kterém nejsou přímo data od uživatele ale zástupné parametry, databázový systém poté už očekává konkrétní datové typy a znemožní uložit jiný datový typ, než databázový stroj očekává.

```
INSERT INTO PRODUCT (name, price) VALUES (?, ?)
```

Otazníky v SQL dotazu představují zástupné parametry. Do kterých se až po kontrole vloží data, se kterými chceme pracovat.

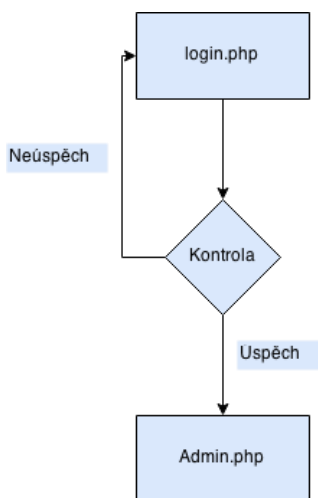
```
$sql = "SELECT id, login, heslo FROM admini WHERE login = :login AND heslo = :heslo LIMIT 1;";
```

```
$ok = $db->prepare($sql);
```

```
$ok->bindParam(':login', htmlentities($_POST['login']));
```

```
$ok->bindParam(':heslo', $mirekheslo);
```

```
$ok->execute();
```



Obrázek 6 Kontrola uživatele



Závěr

Hlavním úkolem této práce bylo vytvoření webového portálu, který by měl hlavně studenty ale i širokou veřejnost informovat o běžících projektech týkající se elektromobility TUL. Představit elektrické auto, které škola vyvíjí a právě touto cestou ho prezentovat.

Ve zprávě jsou uvedeny zásadní kroky, které jsou nezbytné pro vytvoření správného webu. Je zde poukázáno na kroky, které jsou považovány za nejdůležitější a každý vývojář webových aplikací by se jimi měl pečlivě zabývat.

V první řadě je důležité získávat průběžně veškeré, informace dostupné od zadavatele, ujasnit si podmínky, vzhled a rozsah celé aplikace. Též je velmi důležité pokládat zadavateli co nejvíce otázek a stále si ověřovat, zda se celá aplikace vyvíjí požadovaným směrem, předejdeme tím nečekaným závěrům a zbytečným chybám.

Hlavním pilířem práce však byl redakční systém. Systém se neopírá o žádné již vytvořené redakční systémy a je navržen tak aby přesně vyhovoval požadavkům portálu. Je však na tolik obecný, že jej lze použít jako Framework pro další webové projekty.



Ukázka Zdrojových kódů

