

Odesíláme tweety na server

V této kapitole:

- Přidání nové stránky s formulářem
- Získání a ošetření vstupu od uživatele
- Odeslání dat a získání odpovědi

Abychom v naší aplikaci mohli používat i přidávání tweetů skrze náš účet, je třeba vytvořit funkci, která nám tuto funkcionalitu umožní. Prozatím to bude funkce jednoduchá, která přidá jeden tweet po stisknutí tlačítka **odeslat**. V následující kapitole věnované práci s GPS si ale funkcionalitu rozšíříme o ukládání GPS lokace ke každému našemu tweetu a zobrazování těchto tweetů na mapě.

Přidání nové stránky s formulářem

Nejprve ze všeho si v našem souboru `index.html` vytvoříme stránku, která bude obsahovat několik primárních položek. Budou to následující:

- textarea čili box, do kterého budeme vkládat náš text,
- počet napsaných znaků, jelikož jeden Twitter status může mít maximálně 140 znaků,
- tlačítko Odeslat.

Kód, který dosadíme do našeho souboru `index.html`, bude tedy vypadat takto:

```
<!-- STRANKA S ID addNewtweet, SLOUZI JAKO STRANKA OBSAHUJICI FORMULAR
PRO ODESLANI TWEETU -->
<div data-role="page" id="addNewtweet">
  <!-- HEADER S NADPISEM -->
  <div data-theme="a" data-role="header">
    <h3>
```

```

    Nový tweet
</h3>
<!-- HORIZONTALNI MENU APLIKACE -->
<div data-role="navbar" data-iconpos="bottom">
  <ul>
    <!-- POLOZKA MENU -->
    <li>
      <a href="#tweetsFromOthers" onClick="gettweetsFromOthers();"
        data-theme="b" data-icon="refresh" class="ui-btn-active">
        Výpis
      </a>
    </li>
    <li>
      <a href="#addNewtweet" data-theme="b" data-icon="plus">
        Nový tweet
      </a>
    </li>
    <li>
      <a href="#mytweets" data-theme="b" data-icon="home">
        Mé tweety
      </a>
    </li>
    <li>
      <a href="#settings" data-theme="" data-icon="gear">
        Nastavení
      </a>
    </li>
  </ul>
</div>
</div>
<!-- OBSAH STRANKY, V TOMTO PRIPADE VYPIS NOVYCH TWEETU -->
<div data-role="content">
  <!-- POLOZKA TEXTAREA, VIZ
  http://jquerymobile.com/demos/1.1.0/docs/forms/textinputs/ -->
  <div data-role="fieldcontain">
    <label for="textarea">tweet:</label>
    <textarea name="textarea" id="tweetContent"></textarea>
  </div>

  <div>
    Napsáno znaků: <div id="counter">0</div>
  </div>

  <!-- POLOZKA TLACITKA, VIZ http://jquerymobile.com/demos/1.1.0/docs/
  buttons/buttons-types.html -->
  <a href="#" id="sendtweetBtn" onClick="postNewtweet();"

```

```

data-role="button" data-icon="plus">Odeslat tweet</a>

</div>
<!-- PATICKA STRANKY -->
<div data-theme="a" data-role="footer" data-position="fixed">
  <h3>
    <a onClick="openInChildBrowser('www.jirivavru.com');">
      &copy; jirivavru.com
    </a>
  </h3>
</div>
</div>

```

Poprvé se zde setkáváme s formulářovou položkou `textarea`. Jedná se o klasický HTML tag, ovšem jQuery Mobile mu přidává výchozí vzhled optimalizovaný pro mobilní zařízení.

Vše ostatní jsou již nám známé elementy a kromě popisu v rámci kódu k nim není třeba dalšího výkladu.

Více o textových polích v jQuery Mobile se lze dočíst na oficiálních stránkách <http://jquerymobile.com/demos/1.1.0/docs/forms/textinputs/>.

Pokud jste stránku přidali do našeho `index.html` souboru a spustíme emulátor, zobrazí se nám po kliknutí na tlačítko s názvem **Nový tweet** následující stránka obsahující všechny potřebné prvky, viz obrázek 39.



Obrázek 39: Obrazovka s formulářem pro zadání textu nového tweetu

Získání a ošetření vstupu od uživatele

Samotný formulář v tuto chvíli nemá žádnou funkcionalitu. K tomu, abychom mohli data odeslat, potřebujeme funkci pro odesílání. Jak již bylo zmiňováno na začátku této kapitoly, post zasílaný na Twitter může mít maximálně 140 znaků. Vytvoříme proto nejprve funkci, která tento počet ohlídá a nedovolí poslat zprávu prázdnou ani delší než zmiňovaných 140 znaků.

Nejprve přidáme událost na naši položku `textarea`:

```
<textarea name="textarea" id="tweetContent"
  onkeyup="watchLengthOftweet();">
</textarea>
```

Jedná se o událost `onkeyup`, která je spuštěna pokaždé, když je zadán znak z klávesnice (ať už klasické hardwarové, nebo softwarové na displeji telefonu).



Poznámka: Zde je potřeba věnovat zvýšenou pozornost těmto událostem, jelikož ne všechny klasické javascriptové události na virtuální klávesnici u zařízení s Androidem fungují. Je to dáno použitím jádra prohlížeče Webkit, tento prohlížeč má i vlastní události. Níže uvedená tabulka by vám ale měla poskytnout základní přehled o dostupných událostech v mobilních prohlížečích Android a iPhone, nikoliv jen v Safari.

Událost-název	Safari	iPhone	Vysvětlení
onabort	1.0	1.0	Pokud je zrušeno načítání obrázků (platí pro <code></code> elementy)
onfocus	1.0	1.0	Pokud element obdrží fokus, např. po kliknutí do formulářového políčka
ongesturechange		2.0	Pokud je prsty pohnuto v rámci gesta (pouze pro iPhone)
ongestureend		2.0	Pokud je gesto ukončeno (povrchu displeje se dotýká 1 nebo 0 prstů) (pouze pro iPhone)
ongesturestart		2.0	Pokud se dva nebo více prstů dotýkají povrchu displeje (pouze pro iPhone)
oninput	1.3	1.0	Pokud je text vložen do elementu
onkeydown	1.0	1.0	Pokud je stisknuta klávesa na klávesnici
onkeypress	1.0	1.0	Pokud je stisknuta a uvolněna klávesa na klávesnici
onkeyup	1.0	1.0	Pokud je uvolněna klávesa na klávesnici
onload	1.0	1.0	Pokud je dokončeno načítání elementu
onmousedown	1.0	1.0	Pokud je stisknuto tlačítko myši nad daným elementem
onmousemove	1.0	1.0	Pokud je myš pohybováno uvnitř elementu
onmouseout	1.0	1.0	Pokud myš přejíždíme mimo element

Událost-název	Safari	iPhone	Vysvětlení
onmouseover	1.0	1.0	Pokud myš přejedeme mimo element
onmouseup	1.0	1.0	Pokud je tlačítko myši puštěno ze stisku nad daným elementem
onmousewheel	1.0	1.0	Pokud je zarolováno kolečkem myši
onorientationchange		1.1	Pokud je změněna orientace displeje (naležato, nastojato)
onreset	1.0	1.0	Pokud je formulář resetován
onresize	1.0	1.0	Pokud je element zvětšen
onscroll	1.2	1.0	Pokud je elementem srolováno
onselect	1.0	1.0	Pokud je text uvnitř elementu označen
onsubmit	1.0	1.0	Pokud je element (např. Formulář) odeslán
ontouchcancel		2.0	Pokud systém zruší trackování dotyku na displeji (pouze pro iPhone)
ontouchend		2.0	Pokud se daná událost ukončí na povrchu displeje (pouze pro iPhone)
ontouchmove		2.0	Pokud se prst pro danou událost pohybuje po povrchu (pouze pro iPhone)
ontouchstart		2.0	Pokud se prst pro danou událost dotýká povrchu (pouze pro iPhone)
onunload		2.1	Pokud je element uvolněn ze stránky

Na událost `onkeyup`, která funguje bez problému v i v prohlížeči s webkitem, tedy zavoláme nově vytvořenou funkci s názvem `watchLengthOftweet`, kterou doplníme do našeho souboru `main.js`. Funkce bude vypadat následovně:

```
//Funkce pro ohlídní délky tweetu na max. 140 znaků
function watchLengthOftweet()
{
  //Celkový počet znaků
  var countOfChars = $('#tweetContent').val().length;
  console.log(countOfChars);
  //Aktualizujeme vizuální počítadlo
  $('#counter').html(countOfChars);
  //Pokud je počet znaků větší než 140 (1 tweet) nebo je roven 0,
  //zablokujeme tlačítko pro odeslání
  if(countOfChars > 140 || countOfChars == 0)
  {
    console.log("Text je dlouhý nebo příliš krátký");
  }
}
```

```

//Změníme atribut onClick na prázdný, takže po kliknutí se neprovede
//zavolání žádné funkce
$('#sendtweetBtn').attr("onClick","");

}
//Pokud je vše v pořádku
else
{
//Změníme atribut onClick na zavolání funkce postNewtweet
$('#sendtweetBtn').attr("onClick","postNewtweet()");
}
}

```

Pokud je text v `textarea` delší než 140 znaků, změníme atribut `onClick` na prázdný, takže po kliknutí se neprovede zavolání žádné funkce. Pokud jsou hodnoty menší než 140 znaků, uvedeme do atributu `onClick` opět název funkce, která zpracuje odeslání tweetu na server.

Odeslání dat a získání odpovědi

Nyní je třeba vytvořit funkci, která bude provádět následující úkony:

1. Převezme text z `textarea`.
2. Ověří, zda text není příliš dlouhý (jedno ověření již máme, ale je dobré kontrolovat text ještě přímo před odesláním na jeho validnost).
3. Pokud text nevyhovuje našim parametrům, zobrazíme chybovou zprávu a vrátíme hodnotu `false`.
4. Pokud je vše v pořádku, dosadíme text z `textarea` jako parametr URL.
5. Provedeme odeslání.
6. V závislosti na úspěšnosti či neúspěšnosti odeslání zobrazíme stavovou zprávu a provedeme vynulování počítadla a vymazání textu v `textarea`, aby byl formulář připraven pro odeslání dalšího tweetu.

Funkci pojmenujeme `postNewtweet` a bude volána událostí `onClick` na tlačítku s ID `sendtweetBtn`.

Zdrojový kód funkce `postNewtweet`:

```

/*Funkce pro odesílání nového tweetu*/
function postNewtweet()
{
  console.log("odesílám nový tweet");
  //Celkový počet znaků
  var countOfChars = $('#tweetContent').val().length;
  //Pokud je počet znaků větší než 140 (1 tweet) nebo je roven 0,

```

```

//zablokujeme tlačítko pro odeslání
if(countOfChars > 140 || countOfChars == 0)
{
    console.log("Text je dlouhý nebo příliš krátký");
    showMessage("Text je příliš dlouhý nebo krátký, upravte jej prosím");
    hideMessageOrPreloader();
    return false;
}
else{
    //Provedeme odeslání tweetu
    var contentOftweet = $('#tweetContent').val()
    console.log("Odesílám");
    showMessage("Odesílám tweet");
    //Použijeme POST metodu, viz
    //https://dev.twitter.com/docs/api/1/post/statuses/update
    OAuth.post('http://api.twitter.com/1/statuses/update.JSON',
    //parametry, nyní pouze text tweetu, nemusíme použít URL encode
    //jsOAuth ho provede za nás
    {'status':contentOftweet },
    function(data) {
        //Odpověď ze serveru
        var entries = JSON.parse(data.text);
        console.log(entries);

        //Zobrazíme informační zprávu o úspěšném odeslání
        showMessage("tweet úspěšně odeslán");
        //A zprávu skryjeme
        hideMessageOrPreloader();
        //Provedeme vymazání textu v textarea
        $('#tweetContent').val('');
        //A vynulování počítadla
        $('#counter').html('0');
    },
    //Pokud dojde k chybě při odesílání tweetu, zobrazíme chybovou //hlášku
    function(x, e) {
        console.log("AppLaudLog: Error " + data);
        showMessage("Chyba při načítání tweetů");
        hideMessageOrPreloader();
    }
    );
}
}

```

Pokud kód doplníme do našeho souboru `main.js` a spustíme, měli bychom po odeslání našeho prvního tweetu spatřit na displeji telefonu či emulátoru hlášku *tweet úspěšně odeslán*.

Ve spodní části debug okna v našem Eclipse bychom měli také vidět výpis odpovědi ze serveru po přijetí našeho tweetu.

Tím máme první verzi odesílání tweetu na server hotovou. V následující kapitole se podíváme na to, jak ji obohatit o odesílání GPS souřadnic.