



Pedagogická
fakulta
Faculty
of Education

Jihočeská univerzita
v Českých Budějovicích
University of South Bohemia
in České Budějovice

Jihočeská univerzita v Českých Budějovicích

Pedagogická fakulta

Katedra Informatiky

Bakalářská práce

Webové formuláře v HTML5 a Web Forms 2.0

Vypracoval: Václav Vlk, DiS.

Vedoucí práce: PaedDr. Petr Pexa, Ph.D.

České Budějovice 2013

JIHOČESKÁ UNIVERZITA V ČESKÝCH BUDĚJOVICÍCH
Fakulta pedagogická
Akademický rok: 2011/2012

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Václav VLK**
Osobní číslo: **P10382**
Studijní program: **B7507 Specializace v pedagogice**
Studijní obor: **Informační technologie ve vzdělávání**
Název tématu: **Webové formuláře v HTML5 a Web Forms 2.0**
Zadávající katedra: **Katedra informatiky**

Z á s a d y p r o v y p r a c o v á n í :

Cílem bakalářské práce bude rozbor aktuálního stavu vývoje nové a perspektivní technologie Web Forms 2.0 pro tvorbu webových formulářů v HTML5 webových aplikacích. Bude provedeno porovnání s tradičními technikami tvorby webových formulářů (HTML4, XHTML1) i technikami paralelně vyvíjenými (XForms 1.0). Výstupem práce bude sada praktických příkladů, popis výhod a možností této nové technologie a otestování podpory v aktuálních verzích webových prohlížečů. V praktické aplikaci bude kladen důraz na validitu a použitelnost se zaměřením na běžného uživatele, bude provedeno i testování použití v extrémních případech (přetěžování).

Rozsah grafických prací: CD ROM

Rozsah pracovní zprávy: 60

Forma zpracování bakalářské práce: tištěná

Seznam odborné literatury:

1. HICKSON, Ian. Web Forms 2.0. In: Web Forms 2.0 [online]. 27.6.2004 [cit. 2012-03-25]. Dostupné z: <http://www.whatwg.org/specs/web-forms/2004-06-27-call-for-comments/>
2. HUNT, Lachlan. Seznámení s HTML 5. In: Seznámení s HTML 5 [online]. 24. 12. 2007 [cit. 2012-03-25]. Dostupné z: <http://interval.cz/clanky/seznameni-s-html-5/>
3. KJAR, Olav Junker. Web Forms 2. In: Web Forms 2 [online]. 24. 12. 2007 [cit. 2012-03-25]. Dostupné z: <http://olav.dk/wf2/demo/>
4. SATROM, Brandon. Better Web Forms with HTML5 Forms: Building HTML5 Applications. In: Better Web Forms with HTML5 Forms: Building HTML5 Applications [online]. 11.2011 [cit. 2012-03-25]. Dostupné z: <http://msdn.microsoft.com/en-us/magazine/hh547102.aspx>
5. WEYL, Estelle. Web Forms 2.0: New form elements & Attributes in HTML5. In: Web Forms 2.0: New form elements & Attributes in HTML5 [online]. [cit. 2012-03-25]. Dostupné z: <http://standardista.com/forms/#slide1>
6. RUTER, Weston. Web Forms 2.0 test suite. In: Web Forms 2.0 test suite [online]. 29.11.2007 [cit. 2012-03-25]. Dostupné z: <http://webforms2.googlecode.com/svn/trunk/testsuite/index.html>

Vedoucí bakalářské práce: PaedDr. Petr Pexa, Ph.D.
Katedra informatiky

Datum zadání bakalářské práce: 12. dubna 2012

Termín odevzdání bakalářské práce: 26. dubna 2013


Mgr. Michal Vančura, Ph.D.

děkan




doc. PaedDr. Jiří Vaníček, Ph.D.
vedoucí katedry

V Českých Budějovicích dne 12. dubna 2012

Prohlášení

Prohlašuji, že svoji bakalářskou práci jsem vypracoval/a samostatně pouze s použitím pramenů a literatury uvedených v seznamu citované literatury.

Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své bakalářské práce, a to v nezkrácené podobě fakultou elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách, a to se zachováním mého autorského práva k odevzdanému textu této kvalifikační práce. Souhlasím dále s tím, aby toutéž elektronickou cestou byly v souladu s uvedeným ustanovením zákona č. 111/1998 Sb. zveřejněny posudky školitele a oponentů práce i záznam o průběhu a výsledku obhajoby kvalifikační práce. Rovněž souhlasím s porovnáním textu mé kvalifikační práce s databází kvalifikačních prací Theses.cz provozovanou Národním registrem vysokoškolských kvalifikačních prací a systémem na odhalování plagiátů.

V Českých Budějovicích dne

Anotace

Cílem bakalářské práce bude detailní rozbor nové technologie pro webové formuláře v HTML5 Web Forms 2.0. Každá syntaktická část bude popsána a vysvětlena a bude uveden její praktický příklad. Výstupem práce, bude popis této nové technologie s velkým důrazem na praktické využití v běžné praxi. Bakalářská práce bude obsahovat vzorový webový formulář s použitím technologie Web Forms 2.0. Funkčnost tohoto vzorového formuláře bude testována v několika webových prohlížečích a jednotlivé chyby a odlišnosti budou podrobně popsány. Dále tento webový formulář bude testován i z hlediska validity.

Klíčová slova

HTML5, autocomplete, autofocus, form, formaction, formenctype, formmethod, formnovalidate, formtarget, max, min, multiple, pattern, placeholder, required, step, date, tel, time, url, week, range.

Abstract

The paper deals with the detailed analysis of new technologies for web form in HTML 5 Web Forms 2.0. The first part is devoted to HTML, from his first introduction to the novelty HTML5. The next section will be analyzed in detail the language HTML5, which will be described in different syntax. Next and last section provides a detailed analysis of the syntax for Web Forms Web Forms 2.0. This section will be described as news, which brings the technology, support for various search engines, etc..

Keywords

HTML5, autocomplete, autofocus, form, formaction, formenctype, formmethod, formnovalidate, formtarget, max, min, multiple, pattern, placeholder, required, step, date, tel, time, url, week, range.

Poděkování

Tímto bych rád poděkoval vedoucímu mé bakalářské práce PaedDr. Petru Pexovi za odborné vedení, přínosné informace, podnětné připomínky, ochotu a čas, který mi věnoval.

Obsah

1	ÚVOD.....	10
2	TEORETICKÁ ČÁST	11
2.1	HISTORIE FORMULÁŘŮ HTML5	11
2.2	ROZDÍLY V XHTML A HTML5	11
2.3	VALIDACE FORMULÁŘŮ POMOCÍ PŘEDCHOZÍCH VERZÍ HTML	11
2.3.1	<i>Ukázka validace formulářového pole EMAIL</i>	<i>12</i>
2.4	VALIDACE FORMULÁŘŮ POMOCÍ HTML5.....	13
3	PRAKTICKÁ ČÁST.....	14
3.1	FORMULÁŘE.....	14
3.1.1	<i>Element FORM.....</i>	<i>14</i>
3.1.2	<i>Element FIELDSET</i>	<i>17</i>
3.1.3	<i>Element LEGEND.....</i>	<i>19</i>
3.1.4	<i>Element LABEL</i>	<i>21</i>
3.1.5	<i>Elementy INPUT.....</i>	<i>22</i>
3.1.6	<i>DATALIST.....</i>	<i>74</i>
3.1.7	<i>OPTGROUP.....</i>	<i>76</i>
3.1.8	<i>OPTION</i>	<i>77</i>
3.1.9	<i>TEXTAREA</i>	<i>79</i>
3.1.10	<i>KEYGEN</i>	<i>81</i>
3.1.11	<i>OUTPUT</i>	<i>83</i>
3.1.12	<i>PROGRESS.....</i>	<i>84</i>
3.1.13	<i>METER.....</i>	<i>85</i>
3.2	GLOBALNÍ ATRIBUTY V HTML5.....	86
3.2.1	<i>ACCESSKEY</i>	<i>86</i>
3.2.2	<i>CLASS</i>	<i>87</i>
3.2.3	<i>CONTENTEDITABLE.....</i>	<i>87</i>
3.2.4	<i>CONTEXTMENU.....</i>	<i>87</i>
3.2.5	<i>DIR</i>	<i>87</i>
3.2.6	<i>DRAGABLE.....</i>	<i>87</i>
3.2.7	<i>HIDDEN</i>	<i>87</i>

3.2.8	<i>ID</i>	88
3.2.9	<i>LANG</i>	88
3.2.10	<i>SPELLCHECK</i>	88
3.2.11	<i>TABINDEX</i>	88
3.2.12	<i>TITLE</i>	88
4	UKÁZKOVÝ FORMULÁŘ	89
4.1	VALIDACE FORMULÁŘE	90
4.1.1	<i>Jméno a příjmení</i>	90
4.1.2	<i>Telefon</i>	90
4.1.3	<i>Email</i>	91
4.1.4	<i>Datum</i>	91
4.1.5	<i>Počet ubytovaných osob</i>	91
4.1.6	<i>Vyberte pokoj</i>	92
4.1.7	<i>Vyberte penzi</i>	92
4.1.8	<i>Checkbox</i>	92
4.1.9	<i>Čas příjezdu a odjezdu</i>	92
4.1.10	<i>Způsob platby</i>	93
4.1.11	<i>Poznámky</i>	93
5	ZÁVĚR	94
6	PRAKTICKÁ APLIKACE	95

1 Úvod

Webové formuláře jsou základním prvkem tvorby webových prezentací a fungování internetu jako takového. Každá stránka obsahuje nějaký formulář, ať už se jedná o e-shop s formulářem na registraci uživatele nebo jednoduchou webovou prezentaci na malý penzion, kde webový formulář bude použit na rezervaci ubytování.

Proč můžeme považovat webové formuláře za základní kámen internetu? Odpověď je snadná. Internet dělá svoji výjimečnost jeho dynamičnost. Veškerá aktualizace webů, a blogů je tvořena pomocí formulářů. Dnešní doba přináší mnoho možností, jak webové formuláře tvořit.

Cílem této bakalářské práce bude seznámení s novou možností, jak vytvářet webové formuláře v HTML 5 tedy Web Forms 2.0. Jelikož tato nová technologie obsahuje nové příkazy, pro tvorbu formulářů bude obsahovat teoretickou a praktickou část.

V teoretické části této práce bude nastíněn nový jazyk HTML 5, kde budou popsány rozdíly mezi HTML a HTML 5. Další kapitola bude věnována webovým formulářům a validací dat, které do daného formuláře vkládáme.

Praktická část bude popisovat jednotlivé nové příkazy na tvorbu webových formulářů. Každý příkaz bude podrobně popsán a jeho funkčnost bude otestována v několika nejpoužívanějších prohlížečích a to (Opera, Internet Explorer 8, 9 a 10, Mozilla Firefox a Google Chrome) na závěr tohoto testování bude popsána funkčnost.

2 Teoretická část

V teoretické části jsou popsány rozdíly mezi formuláři v HTML5 a předchozích verzích. Hlavní důraz je kladen na validaci formulářů. Díky validaci jsme schopni získat data, která jsou pro nás důležitá v takové podobě, jak potřebujeme.

2.1 Historie formulářů HTML5

Historie formulář HTML5 sahá do roku 2005, v tomto roce se nedá mluvit o formulářích v HTML5, ale vznikla samostatná sekce pod názvem Web Forms 2.0. Za touto novou specifikací stojí Ian Hackson, v té době zaměstnanec Ope-ry. Takto navržená specifikace webových formulářů byla tedy v roce 2005 pře- daná skupině W3C, která převzala vývoj. Později se Web Forms 2.0 stává sou- částí specifikace HTML5 a nadále jsou webové formuláře doplněné o nové prvky (atributy, elementy, atd.) součástí HTML5.

2.2 Rozdíly v XHTML a HTML5

Hlavními rozdíly v XHTML a HTML5 jsou nové atributy a nové typy vstupních prvků. Které slouží pro validaci formulářů, která je díky těmto no- vým prvkům uživatelsky příznivější a přehlednější, tak díky novým vstupním prvkům se objevuje implementován například kalendář, který se XHTML, mu- sel programovat pomocí JavaScriptu. Těchto nových prvků v HTML5 najdeme hned několik a všechny jsou popsány v této bakalářské práci.

2.3 Validace formulářů pomocí předchozích verzí HTML

Předchozí verze formulářů, tedy formuláře programované pomocí jazyka HTML a později pomocí XHTML jsou validovány pomocí JavaScriptu. Vali- dace byla dříve velice obtížná, nepřehledná a zdlouhavá. Pro z validování jed- noho pole, musel programátor napsat dlouhý a nepřehledný JavaScript, který dané pole ošetřoval. Pro ošetření celého formuláře, muselo být napsáno tedy

několik desítek řádů kódu. Další část této kapitoly se bude věnovat pouze XHTML. Jazyk HTML, je už velice zastaralý a žádný programátor formulářů jej nepoužívá.

2.3.1 Ukázka validace formulářového pole EMAIL

Pro ukázkou, jak v dřívější verzi HTML, tedy XHTML docházelo k validování, jsem použil formulářové pole pro email. V HTML5 je pro to vytvořen input typu EMAIL, který vše zařídí za nás. V XHTML musela být validace zajištěna pomocí JavaScriptu, tento skript poté pomocí regulárního výrazu ošetřoval zadaný řetězec.

2.3.1.1 Zdrojový kód pro validaci emailu

- **Zdrojový kód JavaScript**

```
<script>
<!-- Begin function test(formular)
{
if (/^\w+([\.-]?\w+)*@\w+([\.-]
]?\w+)*(\.\w{2,3})+$/ .test(formular.email.value))
{
return (true)
}
alert("Špatně zadaná emailová adresa!")
return (false)
}
</script>
```

- **Zdrojový kód XHTML**

```
<form onSubmit="return test(this)">
  <label>Zadejte emailovou adresu:</label>
  <input type="text" name="email">
  <input type="submit" value="Submit">
</form>
```

2.4 Validace formulářů pomocí HTML5

Validace formulářů v HTML5 je o dost lehčí. O veškeré ošetření polí se stará prohlížeč. Vše se děje pomocí správně zvolených inputů, pokud tedy budeme potřebovat od uživatele, aby zadal email, musíme zvolit input typu email. Prohlížeč už pak na své straně ošetří správné zadání. Pokud by se uživatel snažil zadat email ve špatném tvaru, bude ihned na tuto chybu upozorněn. Veškeré další ošetření například velikost (délka) emailové adresy, zda je pole nutné vyplnit před odesláním formuláře k dalšímu zpracování atd. se provádí pomocí atributů, které jsou popsány v této bakalářské práci. Pro ukázkou je použito stejně jako v předchozím případě pole pro zadání emailu.

- **Zdrojový kód emailu v HTML5**

```
E-mail: <input type="email" name="email" required>
```

Takto vytvořené pole, tedy input typu email, bude ošetřovat správnost zadání emailové adresy a pomocí atributu required prohlížeči říkáme, že toto pole musí být pro odeslání formuláře vyplněno. V kapitolách jednotlivých inputů jsou popsány jejich povolené atributy, které slouží pro validaci.

3 Praktická část

V této části bakalářské práce dojde k popsání jednotlivých částí zdrojového kódu pro práci s formuláři. Jednotlivé syntaxe a jejich atributy budou popsány a otestovány v nejpoužívanějších prohlížečích. Vše bude zprostředkováno na ukázkách zdrojového kódu a na praktické ukázce bude popsána funkčnost jednotlivých syntaxí.

3.1 Formuláře

Formuláře jsou nedílnou součástí většiny možná i všech moderní webů. Obsahují mnoho ovládacích prvků, jako jsou tlačítka, textová pole kalendáře, zaškrtnávací políčka a mnoho dalších prvků. A právě tyto základní prvky budou teď popsány. Veškeré zdrojové kódy jsou testovány v těchto prohlížečích.

- Google Chrome 25
- Internet Explorer 8
- Internet Explorer 9
- Internet Explorer 10
- Opera 12.02
- Mozilla Firefox 18.0.2

Každou aktualizací některého z prohlížečů může dojít k případné opravě. A prvek může být podporován v plném rozsahu. Proto uvádím přesnou testovanou verzi.

3.1.1 Element FORM

Form je základní prvek formulářů, jedná se o příkaz, který prohlížeči říká, že se bude jednat o data, která budou dále zpracovávána. Tyto data, se dále zpracovávají pomocí skriptu. Form má několik atributů, pomocí nichž dochází ke zpracování formuláře. Základním atributem je action, který zde slouží

k odesílání (zpracovávání) dat formuláře, nabývá dvou hodnot a to GET, což je výchozí hodnota a slouží k odeslání dat pomocí URL adresy. Další hodnotou je POST, zde jsou data zabalena a šifrována do těla http v tomto případě nemají vstupní data omezenou velikost. Dalším ze základních atributů form je action, který obsahuje URL adresu soboru, který bude zpracovávat daný formulář nejčastěji pomocí PHP.

3.1.1.1 Možné atributy elementu FORM

- **accept-charset** - tento atribut určuje, jakým datovým typem bude formulář zpracováván. Nejčastěji UTF-8, windows-1250, atd...
- **action** - tento atribut použijeme, pokud chceme, aby formulář zpracovával nějaký jiný skript na nějaké jiné adrese parametr atributu action je tedy adresa URL
- **autocomplete** - atribut autocomplete je defaultně zapnutý vždy, výchozí hodnota je tedy nastavena na "ON", pokud nechceme použít takzvaný našeptávač musíme tuto hodnotu vypnutou, tím že použijeme tento atribut a nastavíme její hodnotu na "OFF"
- **enctype** - tímto atributem určujeme způsob kódování formuláře. Výchozí hodnota je nastavena application/x-www-form-urlencoded, v tomto případě jsou mezery nahrazeny +, speciální znaky jsou předělány na hexadecimální hodnoty. Další možností je multipart/form-data a poslední možností je text/plain.
- **method** - určuje jakým způsobem budou data odeslána. Tento atribut může nabýt dvou hodnot GET a POST. Pokud tento atribut nevyplníme, je jeho výchozí hodnota nastavena na GET. GET odesílá data pomocí URL adresy. POST data jsou zabalena v těle http a jsou šifrována, tato metoda je pro odeslání dat doporučována.



















- **name** - pojmenovává daný formulář k dalšímu zpracování. Slouží převážně v tom případě, pokud je na stránce více formulářů.
- **novalidate** - slouží k validaci dat před odesláním. Pokud použijeme tento atribut, nebude hodnota validována.
- **target** - atribut nabývající hodnot `_blank` (odpověď v novém okně), `_top` (vyvolá odpověď v hlavním rámu), `_parent` (odpověď v nadřazeném rámu), `_self` (odpověď v aktuálním rámu). Hodnota `_self` je nastavena defaultně.

3.1.1.2 Zdrojový kód FORM

```
<form action="odesli.php" method="post" accept-charset="UTF-8"
autocomplete="on" novalidate target="_self" name="formulář"
ectype=" application/x-www-form-urlencoded ">
```

3.1.1.3 Tabulka s podporou elementu FORM a jeho atributů

Element form je starý východní element, který je podporován a převzat z předchozích verzí HTML, není tedy potřeba testovat samotný form, ale pouze jeho nové atributy. Čímž jsou pouze `autocomplete` a `novalidate`.

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
<code>autocomplete</code>						
<code>novalidate</code>						

Tabulka 1: Podpora elementu FORM a jeho atributů v prohlížečích

3.1.2 Element FIELDSET

Fieldset je jednoduchý nástroj, jak seskupovat části formuláře se stejnou tematikou a tím celý formulář zpřehlednit. V příkladu níže, lze vidět krátký a jednoduchý formulář, který je rozdělen do dvou částí a to osobní údaje fakturační údaje.

3.1.2.1 Možné atributy elementu FIELDSET

- **Disabled** – atribut kterým říkáme, že nepůjde měnit hodnoty elementů v tomto fieldsetu. To znamená, že veškeré elementy, které jsou v této části, budou zašedlé a hodnota nepůjde upravit.
- **Form** – element můžeme také definovat mimo tento formulář, to znamená, že data mohou být připojena z jiného formuláře a to za pomoci tohoto atributu. Je nutné uvést jako parametr id formuláře, ze kterého se mají data připojit.
- **Name** – jak už název napovídá, tento atribut slouží k pojmenování tohoto prvku. To slouží k případnému dalšímu zpracování pomocí JavaScriptu.

3.1.2.2 Zdrojový kód FIELDSET

```
<fieldset name="udaje_zkl" disabled form="formular">
Jméno a příjmení: <input type="text" value="Václav Vlk"/><br />
Ulice: <input type="text" value="Komenského" /><br />
ČP: <input type="number" value="45"/><br />
Město: <input type="text" value="České Budějovice"/><br />
PSČ: <input type="number" value="370 01"/><br />
</fieldset>

<fieldset>

<legend>Doručovací adresa:</legend>
    Jméno a příjmení: <input type="text" /><br />
    Ulice: <input type="text" /><br />
    ČP: <input type="number" /><br />
    Město: <input type="text" /><br />
    PSČ: <input type="number" /><br />
</fieldset>
```

























3.1.2.3 Ukázka zobrazení FIELDSET

The image shows two examples of HTML FIELDSET elements rendered in a browser. The top example is a disabled fieldset with the name 'udaje_zkl' and form 'formular'. It contains five input fields: a text field for 'Jméno a příjmení' with the value 'Václav Vlk', a text field for 'Ulice' with the value 'Komenského', a number field for 'ČP' with the value '45', a text field for 'Město' with the value 'České Budějovice', and a number field for 'PSČ' with the value '370 01'. The bottom example is an active fieldset with an empty legend 'Doručovací adresa:'. It contains five empty input fields for the same labels: 'Jméno a příjmení', 'Ulice', 'ČP', 'Město', and 'PSČ'.

Obrázek 1: Zobrazení FIELDSET Google Chrome

3.1.2.4 Tabulka s podporou elementu FIELDSET a jeho atributů

Samotný element Fieldset je podporován ve všech prohlížečích. V této tabulce jsou uvedeny pouze nové atributy.

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
Disabled						
Form						
Name						

Tabulka 2: Podpora Elementu FIELDSET a jeho atributů v prohlížečích

3.1.3 Element LEGEND

Legend slouží jako nadpis pro fieldset, tedy pomocí legendu dáváme název seskupenému poli.

3.1.3.1 Možné atributy elementu LEGEND

Jelikož se jedná pouze o element, který slouží dalšímu elementu fieldset nemá tento element žádné možné atributy.

3.1.3.2 Zdrojový kód LEGEND













```
<fieldset>
  <legend>Doručovací adresa:</legend>
  Jméno a příjmení: <input type="text" /><br />
  Ulice: <input type="text" /><br />
  ČP: <input type="number" /><br />
  Město: <input type="text" /><br />
  PSČ: <input type="number" /><br />
</fieldset>
```

3.1.3.3 Ukázka zobrazení LEGEND



Obrázek 2: Ukázka zobrazení LEGEND v Opeře

3.1.3.4 Tabulka s podporou elementu LEGEND

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
legend						

Tabulka 3: Podpora Elementu LEGEN v prohlížečích

3.1.4 Element LABEL

Label je nedílnou součástí všech formulářů. Mohlo by se zdát, že tento prvek, lze nahradit textem vedle nějakého inputu a tím uživateli říci jaká data po něm požadujeme. Bohužel tomu tak není. Text sice lze vedle jakéhokoli inputu použít, ale problém nastane, pokud budeme mít input typu checkbox nebo radio. Pokud u těchto typů použijeme label, pro jeho výběr můžeme použít i text, tedy klikneme na text a checkbox se zaškrtně. Pokud, ale použijeme prostý text, nic se po kliknutí na text nestane a výběr budeme muset přesně definovat kliknutím na malý čtvereček.

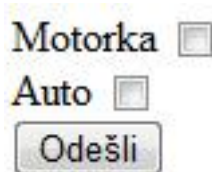
3.1.4.1 Možné atributy elementu LABEL

- **For** - jako parametr je uveden id prvku, ke kterému tento label patří.
- **Form** – element může být uveden i mimo formulář. Tento atribut slouží k připojení s daným formuláře. Připojení se provádí pomocí id formuláře, který chceme připojit.

3.1.4.2 Zdrojový kód LABEL

```
<form action="odesli.php" id="label">
<label>Motorka</label>
<input type="checkbox" name="vozidlo" id="auto" value="auto"><br />
<label for="zena">Auto</label>
<input type="checkbox" name="vozidlo" id="motorka" value="motorka"><br />
<input type="submit" value="Odešli">
</form>
```













3.1.4.3 Ukázka zobrazení LABEL



Obrázek 3: Ukázka zobrazení LABEL v Opeře

3.1.4.4 Tabulka s podporou elementu LABEL

V tomto případě se jedná také o starý známý element. Nový atribut, který je součástí HTML5 je pouze form.

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
form						

Tabulka 4: : Podpora Elementu LABEL v prohlížečích

3.1.5 Elementy INPUT

Element input se používá všude tam, kde ve formuláři chceme po uživateli nějaká vstupní data. Tento element uživateli dovolí vkládat data do formuláře, které dále můžeme zpracovávat. Element input má mnoho pomocných prvků, které můžeme rozlišit do dvou základních kategorií a to Input Type a Input Form. Jednotlivé příklady použití jsou popsány v dalších kapitolách.

3.1.5.1 Nové Input Atributy v HTML5

Input jak již bylo řečeno má mnoho pomocných prvků, tato část je věnovaná pouze těm, které jsou podporovány v HTML5. Ostatní budou stručně popsány pod touto kapitolou.

3.1.5.1.1 AUTOCOMPLETE

Tento atribut je velice hojně používaný. Pokud nastavíme autocomplete na hodnotu "on" u nějakého elementu například jako je v ukázce elementu text, bude nám prohlížeč dávat na výběr podobné hodnoty, které už jednou byly zadány. To velice usnadňuje a zrychluje práci s formulářem. Je možné autocompleten nastavit celému formuláři a to tím, že jej umístíme přímo do formu, pokud je potřeba nějakému elementu zakázat tuto volbu vložíme k tomuto elementu atribut s hodnotou autocomplete="off".

3.1.5.1.1.1 Zdrojový kód AUTOCOMPLETE

```
<input type="text" autocomplete="on">
```

3.1.5.1.1.2 Ukázka zobrazení AUTOCOMPLETE



Obrázek 4: Ukázka zobrazení AUTOCOMPLETE v Google Chrome

3.1.5.1.1.3 Tabulka s podporou AUTOCOMPLETE

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
autocomplete						

Tabulka 5: Podpora atributu AUTOCOMPLETE v prohlížečích

3.1.5.1.2 AUTOFOCUS

Tento prvek použijeme všude tam, kde potřebujeme nebo kde je vyžadováno okamžité vkládání nějakého textu. Pokud nějakému prvku dáme atribut autofocus, ihned po načtení stránky se tento atribut stává aktivním, to znamená, že do něj ihned můžeme psát text, aniž by byl na tento vyhledán a označen. To se může hodit například pro element search.

3.1.5.1.2.1 Zdrojový kód AUTOFOCUS













```
<input type="search" autofocus>
```

3.1.5.1.2.2 Ukázka zobrazení AUTOFOCUS



Obrázek 5: Ukázka zobrazení AUTOFOCUS v Google Chrome

3.1.5.1.2.3 Tabulka s podporou AUTOFOCUS

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
autofocus						

Tabulka 6: Podpora atributu AUTOFOCUS v prohlížečích













3.1.5.1.3 FORM

Tímto atributem říkáme, že data se nacházejí mimo formulář. Tento atribut slouží k jejich následnému připojení s formulářem. Pro připojení se používá id formuláře.

3.1.5.1.3.1 Zdrojový kód FORM

```
<form action="odesli.php" id="form">  
Křestní jméno: <input type="text" name="jmeno"><br>  
<input type="submit" value="Odešli">  
</form>  
Příjmení: <input type="text" name="prijmeni" form="form">
```

3.1.5.1.3.2 Tabulka s podporou FORM

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
form						

Tabulka 7: Podpora atributu FORM v prohlížečích

3.1.5.1.4 FORMACTION

Tímto atributem říkáme, který soubor bude zpracovávat vstupní data. Automaticky jej zpracovává action u prvku <form>, pokud potřebujeme určitý input zpracovat pomocí nějakého jiného souboru, použijeme u něj atribut formaction, který potlačuje výchozí atribut u prvku form.

3.1.5.1.4.1 Zdrojový kód FORMACTION

```
<form action="odesli.php" id="form">  
Křestní jméno: <input type="text" name="jmeno">  
Příjmení: <input type="text" name="prijmeni" form="form">  
<input type="submit" value="Odešli">  
<input type="submit" formaction="odesli_admin.php" value="Odešli admin">  
</form>
```












3.1.5.1.4.2 Ukázka zobrazení FORMACTION

Křestní jméno:

Příjmení:

Obrázek 6: Ukázka zobrazení FORMACTION v Google Chrome

3.1.5.1.4.3 Tabulka s podporou FORMACTION

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
formaction						

Tabulka 8: Podpora atributu FORMACTION v prohlížečích

3.1.5.1.5 FORMETYPE

Tímto atributem specifikujeme kódování dat, která se budou odesílat. Form-type funguje pouze na odeslání pomocí metody POST. Použit, lze pouze na input type submit nebo na tlačítko, které je vytvořeno pomocí obrázku. Tento atribut nabývá hodnot:

- application/x-www-form-urlencoded
- multipart/form-data
- text/plain

3.1.5.1.5.1 Zdrojový kód FORMETYPE













```
<form action="odesli.php" id="form">
Křestní jméno: <input type="text" name="jmeno">
Příjmení: <input type="text" name="prijmeni" form="form">
<input type="submit" value="Odešli">
<input type="submit" formtype="multipart/form-data" value="admin send">
</form>
```

3.1.5.1.5.2 Ukázka zobrazení FORMETYPE



Obrázek 7: Ukázka zobrazení FORMETYPE v Opeře

3.1.5.1.5.3 Tabulka s podporou FORMETYPE

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
formetype						

Tabulka 9: Podpora atributu FORMETYPE v prohlížečích

3.1.5.1.6 FORMMETHOD

Tímto atributem říkáme, pomocí které http metody se bude formulář odesílat. Funkčnost je stejná jako u atributu method u elementu form. Stejně jako v předchozím případě jej lze použít pouze na tlačítko submit nebo na tlačítko tvořené pomocí image.

Nabývá hodnot POST a GET.

3.1.5.1.6.1 Zdrojový kód FORMMETHOD





```
<form action="odesli.php" id="form" method="get">  
Křestní jméno: <input type="text" name="jmeno">  
Příjmení: <input type="text" name="prijmeni" form="form">  
<input type="submit" value="Odešli">  
<input type="submit" formmethod="post" formacti-  
on="odesli_admin.php" value="admin send">  
</form>
```

3.1.5.1.6.2 Ukázka zobrazení FORMEMETHOD



Obrázek 8: Ukázka zobrazení FORMEMETHOD v Opeře

3.1.5.1.6.3 Tabulka s podporou FORMMETHOD

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
formmethod						

Tabulka 10: Podpora atributu FORMMETHOD v prohlížečích

3.1.5.1.7 FORMNOVALIDATE

Jedná se o atribut, pomocí kterého prohlížeči říkáme, zda daná data budou validovaná či nikoli. V některých případech je validace nežádoucí a data, která

uživatel zadá, potřebujeme získat, tak jak je zadá. Tento atribut lze použít stejně jako v u předchozího prvku pouze na tlačítko submit nebo na tlačítko které je tvořené pomocí image.

3.1.5.1.8 Zdrojový kód FORMNOVALIDATE













```
<form action="odesli.php">
Email:<input type="text" name="email">
<input type="submit" formnovalidate value="Odešli">
</form>
```

3.1.5.1.8.1 Ukázka zobrazení FORMNOVALIDATE



Obrázek 9: Ukázka zobrazení FORMNOVALIDATE v Google Chrome

3.1.5.1.8.2 Tabulka s podporou FORMNOVALIDATE

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
formnovalidate						

Tabulka 11: Podpora atributu FORMNOVALIDATE v prohlížečích

3.1.5.1.8.3 FORMTARGET

Tento atribut udává, do jakého okna se má zobrazit výsledek zpracování formuláře. Funkce je stejná jako u atributu target na elementu form. Nabývá tedy hodnot:

- **_self** - výsledek se zobrazí ve stejném okně, jako se zadává. Jedná se o výchozí hodnotu. To znamená, že pokud není uvede-

ný atribut použit, výsledek se bude zobrazovat v aktuálním okně.

- **_blank** - výsledek se zobrazí v novém okně
- **_parent** - výsledek je zobrazen v nadřazeném rámu
- **_top** - výsledek se zobrazí v nejhlavnějším rámu
- **framename** - výsledek zpracování se zobrazí ve formu, který má stejný název

3.1.5.1.8.4 Zdrojový kód FORMTARGET

```
<form action="odesli.php">
Email:<input type="text" name="email">
<input type="submit" formtarget="_top" value="Odešli">
</form>
```

3.1.5.1.8.5 Tabulka s podporou FORMTARGET

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
formtarget						

Tabulka 12: Podpora atributu FORMTARGET v prohlížečích

3.1.5.1.9 HEIGHT a WIDTH

Tyto prvky jsou povoleny pouze u inputu typu image a definují požadovanou velikost obrázku v pixelech. Vždy je nutné uvádět šířku a výšku daného prvku. Takto stylovat tlačítko lze i pomocí css. A můj názor je takový, že veškeré stylování by mělo být ve zvláštním souboru s kaskádovými styly. Ale HTML5 nám dává možnost nastavit to přímo v samotném kódu.

3.1.5.1.9.1 Zdrojový kód HEIGHT a WIDTH













```
<form action="odesli.php">
Email:<input type="text" name="email">
<input type="image" src="tlacitko.gif" alt="ODESLAT"
width="128" height="128">
</form>
```

3.1.5.1.9.2 Ukázka zobrazení HEIGHT a WIDTH



Obrázek 10: Ukázka zobrazení HEIGHT a WIDTH v Google Chrome

3.1.5.1.9.3 Tabulka s podporou HEIGHT a WIDTH

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
formtarget						

Tabulka 13: Podpora atributu HEIGHT a WIDTH v prohlížečích

3.1.5.1.10 LIST

Tento atribut slouží pro rychlý výběr z nepoužívanějších možností. V ukázce vidíme seznam automobilů. Pokud se budeme dotazovat uživatele jaký automobil vlastní, můžeme mu nabídnout z nejčastějších značek. Pokud mu daná hodnota nevyhovuje, může vložit vlastní značku auta.

3.1.5.1.10.1 Zdrojový kód LIST













```
<input type="text" list="seznam">
  <datalist id="seznam">
    <option value="Škoda"></option>
    <option value="Ford"></option>
    <option value="Volvo"></option>
    <option value="Honda"></option>
  </datalist>
```

3.1.5.1.10.2 Ukázka zobrazení LIST



Obrázek 11: Ukázka zobrazení LIST v Mozilla Firefox

3.1.5.1.10.3 Tabulka s podporou LIST

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
list						

Tabulka 14: Podpora atributu LIST v prohlížečích

3.1.5.1.11 MAX a MIN

Atribut min nám udává minimální možnou hodnotu, která bude moci být zadána. Pro opačnou hodnotu, tedy maximální je potřeba použít atribut max.

Tím říkáme prohlížeči, jakou maximální možnou hodnotu budeme moci do formulářového pole zadat. K ověření zadané hodnoty dochází ještě před odesláním formuláře. Pokud je zadaná hodnota, která neodpovídá danému rozmezí, budete na to prohlížečem upozorněni.

3.1.5.1.11.1 Zdrojový kód MAX a MIN













```
<input type="number" max="10" min="-10">
```

3.1.5.1.11.2 Ukázka zobrazení MIN a MAX



Obrázek 12: Ukázka zobrazení MIN a MAX v Google Chrome

3.1.5.1.11.3 Tabulka s podporou MAX a MIN

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
Max a min						

Tabulka 15: Podpora atributu MIN a MAX v prohlížečích

3.1.5.1.12 MULTIPLE

Tento atribut je možné použít všude tam, kde potřebujete do jednoho pole zadat více hodnot. Nejlepší příklad užití je na formuláři, který je určený pro odeslání emailu. Pokud použijeme tento atribut na pole email, je možné zaslat email na více adres například jako v ukázce ve skryté kopii.

Multiple má poměrně dobrou podporu všech novější prohlížečů. Pokud chcete zadat více příjemců, stačí zvolit nějakého z nabídky, použít jako oddělovač "," a ihned se nabídka s možnými příjemci ukáže znova.

3.1.5.1.12.1 Zdrojový kód MULTIPLE







```
<input type="email" multiple name="cc" list="kontakt">
  <datalist id="kontakt">
    <option value="v.vlk@hornistropnice.cz">
    <option value="top09@hornistropnice.cz">
    <option value="sportovnihala@hornistropnice.cz">
    <option value="obec@hornistropnice.cz">
  </datalist>
```

3.1.5.1.12.2 Ukázka zobrazení MULTIPLE



Obrázek 13: Ukázka zobrazení MULTIPLE v Mozila Firefox

3.1.5.1.12.3 Tabulka s podporou MULTIPLE

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
multiple						

Tabulka 16: Podpora atributu MULTIPLE v prohlížečích

3.1.5.1.13 PATTERN

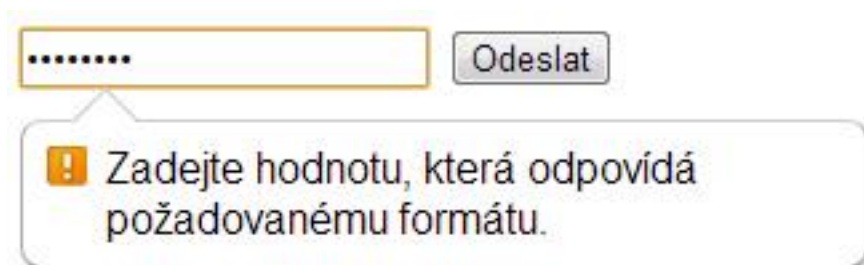
Asi jeden z nepoužívanějších atributů, které HTML5 přináší. Tento atribut má plnit funkci tzv. validace vstupních dat bez použití Java Scriptu. Pokud použijeme tento příkaz a jako parametr použijeme jednoduchý regulární výraz, máme snadno ošetřena vstupní data například po použití s elementem password nebo tel. V příkladu je tento atribut použit s elementem password, který musí být naplněn pouze písmeny, malá nebo velká a délka zadaného řetězce je přesně 4.

Tato funkce má však i možná pro někoho drobný já si myslím, že zadní nedostatek. Prohlížeče nereagují na písmena s diakritikou a automaticky je definují jako špatně zadanou hodnotu, což přináší velké omezení například při validaci inputu text, do kterého chceme zadávat pouze příjmení.

3.1.5.1.13.1 Zdrojový kód PATTERN












```
<input type="password" pattern="[A-z]{4}">
```

3.1.5.1.13.2 Ukázka zobrazení PATTERN



Obrázek 14: Ukázka zobrazení PATTERN v Google Chrome

3.1.5.1.13.3 Tabulka s podporou PATTERN

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
pattern						

Tabulka 17: Podpora atributu PATTERN v prohlížečích

3.1.5.1.14 PLACEHOLDER

Placeholder plní funkci nápovědy pro zadávání. Lze použít všude tam, kde potřebujeme uživateli říci, v jakém formátu požadujeme daná data. V ukázce je placeholder použit na inputu email. Na zobrazení lze vidět, jak se tato nápověda zobrazuje. Jedná se o šedý text, který ihned po vepsání určitého znaku zmizí. Pokud smažeme veškerá zadaná data, text nápovědy se zase objeví.

3.1.5.1.14.1 Zdrojový kód PLACEHOLDER













```
<input type="email" placeholder="Zadejte email">
```

3.1.5.1.14.2 Ukázka zobrazení PLACEHOLDER



Obrázek 15: Ukázka zobrazení PLACEHOLDER v Google Chrome

3.1.5.1.14.3 Tabulka s podporou PLACEHOLDER

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
placeholder						

Tabulka 18: Podpora atributu PLACEHOLDER v prohlížečích

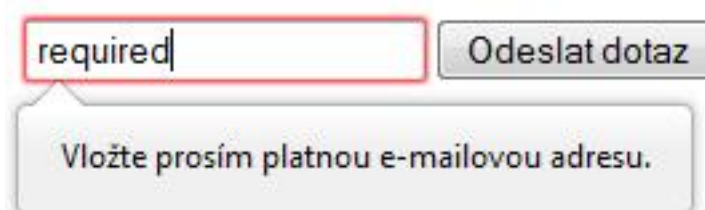
3.1.5.1.15 REQUIRED

Jedná se o jednoduché a přitom efektivní ošetření formuláře. Tímto příkazem, říkáme, že zadaná hodnota nesmí být prázdná. To znamená, že se formulář neodešle, pokud element, který bude mít nastaveno required, bude prázdný. To může sloužit nejen pro nutnost zadání emailu, jako je uvedeno v příkladu, ale také pro mnoho dalších inputů.

3.1.5.1.15.1 Zdrojový kód REQUIRED

```
<input type="email" required>
```

3.1.5.1.15.2 Ukázka zobrazení REQUIRED



Obrázek 16: Zobrazení REQUIRED v Mozilla Firefox

3.1.5.1.15.3 Tabulka s podporou REQUIRED

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
required						

Tabulka 19: Podpora atributu REQUIRED v prohlížečích

3.1.5.1.16 STEP

Tento atribut složí k tzv. krokování. Pokud jej použijeme v elementu number, bude tento element zvyšovat nebo snižovat hodnotu právě o námi zadané krokování. Standardně je toto krokování nastaveno na hodnotu 1. Pokud však potřebujeme krokovat na desetinné místo, musíme použít právě atribut step. Malinký háček je v tom, že hodnota musí být zadaná s desetinou tečkou. Námi používaná desetinná čárka není podporována.

3.1.5.1.16.1 Zdrojový kód STEP













```
<input type="number" step="0.2">
```

3.1.5.1.16.2 Ukázka zobrazení STEP



Obrázek 17: Ukázka zobrazení STEP v Google Chrome

3.1.5.1.16.3 Tabulka s podporou STEP

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
step						

Tabulka 20: Podpora atributu STEP v prohlížečích

3.1.5.2 Starší Input Atributy v předchozích verzích HTML

3.1.5.2.1 ACCEPT

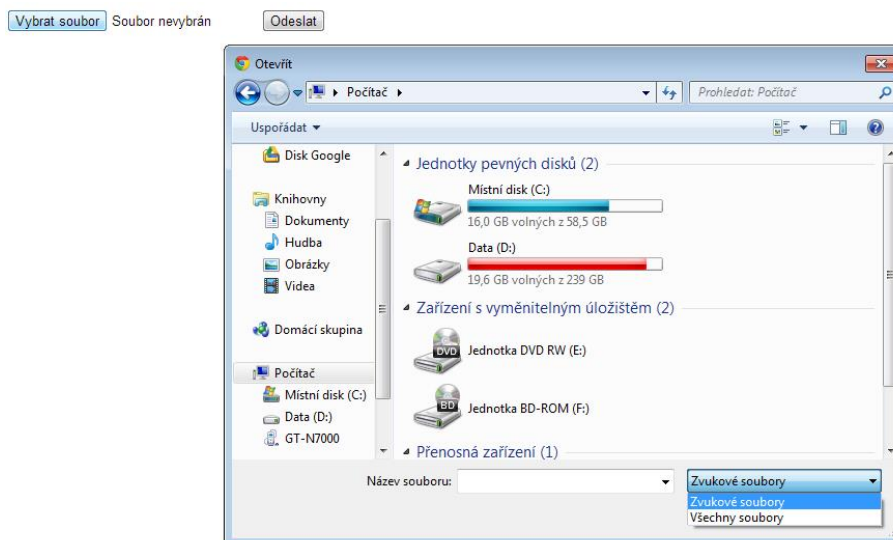
Tento atribut slouží jako atribut k `<input type="file">`, k žádnému jinému typu nelze použít a slouží, jako pomůcka k tomu jaká data mohou být při načítání akceptovaná. Nabývá hodnot:

- **audio/*** - veškeré zvukové soubory
- **video/*** - veškeré video soubory
- **image/*** - veškeré obrázkové soubory

3.1.5.2.1.1 Zdrojový kód ACCEPT











```
<input type="file" name="img" accept="audio/*">
```

3.1.5.2.1.2 Ukázka zobrazení ACCEPT



Obrázek 18: Ukázka zobrazení ACCEPT v Google Chrome

3.1.5.2.1.3 Tabulka s podporou ACCEPT

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
accept						

Tabulka 21: Podpora atributu ACCEPT v prohlížečích

3.1.5.2.2 DISABLED

Pokud uvedeme u nějakého inputu atribut disabled, znamená to, že hodnota která je uvedena popřípadě je input prázdný, uživatel nebude moci upravovat toto pole.

3.1.5.2.2.1 Zdrojový kód DISABLED





```
<input type="number" disabled>
```


3.1.5.2.2 Ukázka zobrazení DISABLED



Obrázek 19: Ukázka zobrazení DISABLED v Internet Explorer 10

3.1.5.2.3 Tabulka s podporou DISABLED

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
disabled						

Tabulka 22: Podpora atributu DISABLED v prohlížečích

3.1.5.2.3 SIZE

Atribut size určuje počet zobrazených hodnot v určitém poli. V příkladu lze vidět, jak tento atribut funguje. Máme select o 7 options, tedy máme na výběr ze sedmi možností, pokud nebude zvolena hodnota size, defaultně je nastavena hodnota na 1, zobrazí se tedy pouze první město a pro zvolení ostatních musíme rozkliknout box. Pokud je hodnota jako v případě nastavena na 4 zobrazí se ihned 4 města a ostatní můžeme vybrat pomocí rolování.

3.1.5.2.3.1 Zdrojový kód SIZE

```
<select size="4" name="města">  
  <option value="brno">Brno</option>  
  <option value="praha">Praha</option>  
  <option value="ostrava">Ostrava</option>  
  <option value="Teplice">Teplice</option>  
  <option value="tabor">Tábor</option>  
  <option value="plzen">Plzeň</option>
```

```

<option value="opava">Opava</option>
</select>













```

3.1.5.2.3.2 Ukázka zobrazení SIZE



Obrázek 20: Ukázka zobrazení SIZE v Internet Explorer 10

3.1.5.2.3.3 Tabulka s podporou SIZE

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
size						

Tabulka 23: Podpora atributu SIZE v prohlížečích

3.1.5.2.4 SRC

Tento atribut určuje cestu, která vede k obrázku, který je použit jako tlačítko, z toho tedy vyplývá, že tento atribut lze použít pouze u inputu typu image.

3.1.5.2.4.1 Zdrojový kód SRC

```

<form action="odesli.php">
  Jméno: <input type="text" name="jmeno">
  <input type="image" src="odesli.gif" alt="Odeslat">
</form>













```

3.1.5.2.4.2 Ukázka zobrazení SRC



Obrázek 21: Ukázka zobrazení SRC v Mozilla Firefox

3.1.5.2.4.3 Tabulka s podporou SRC

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
src						

Tabulka 24: Podpora atributu SRC v prohlížečích










3.1.5.2.5 VALUE

Tento základní atribut slouží, jak už možná název napovídá k nastavení výchozí hodnoty. Do hodnoty value, lze zadat jakoukoli hodnotu, kterou chceme, aby uživatel nemusel zadávat avšak je důležitá pro další zpracování. Takto zadanou hodnotu může uživatel libovolně upravit. V některých případech je hodnotu value udávat vždy. Například u vstupního typu checkbox a radio. Jediný případ kdy hodnotu value nelze použít je `<input type="file">`.

3.1.5.2.5.1 Zdrojový kód VALUE

```
<input value="text">
```

3.1.5.2.5.2 Tabulka s podporou VALUE

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
value						

Tabulka 25: Podpora atributu VALUE v prohlížečích

3.1.5.3 Nové Input type v HTML5

3.1.5.3.1 COLOR

Tento input je zatím podporován pouze prohlížečem Google Chrome a Opera. V ostatních prohlížečích se prezentuje jako input typu text. Pokud si tedy chcete otestovat možnosti tohoto nového prvku, musíte jej spustit právě v těchto dvou prohlížečích. Po kliknutí se zobrazí nové dialogové okno, ve kterém lze vybrat z několika základních barev, nebo pomocí kurzoru můžeme vybrat svou barvu ve spektru barev a poslední řadě lze barvu vybrat přímo pomocí kódu barvy.

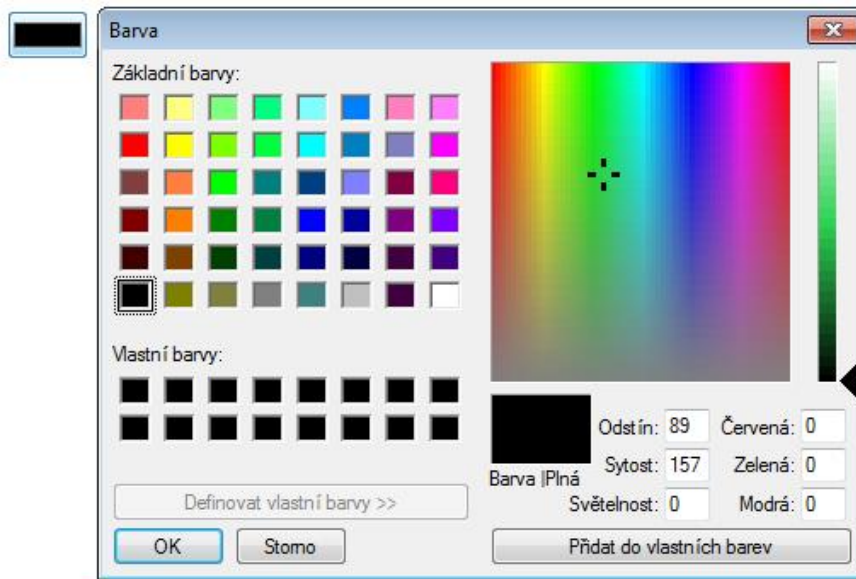
3.1.5.3.1.1 Možné atributy COLOR

- **Autocomplete** – automaticky nám dává nabídku již jednou zadaných hodnot
- **Value** – výchozí hodnota. V tomto případě výchozí barva.
- **List** – Nastavením dáváme na výběr z několika možností, které jsou nejčastěji zadávány.

3.1.5.3.1.2 Zdrojový kód COLOR








```
<input type="color">
```

3.1.5.3.1.3 Ukázka zobrazení COLOR



Obrázek 22: Ukázka zobrazení COLOR v Google Chrome

3.1.5.3.1.4 Tabulka s podporou COLOR

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
color						

Tabulka 26: Podpora input type COLOR v prohlížečích

3.1.5.3.2 DATE

Tento prvek je velmi často využívá, především tam, kde je potřeba získat od uživatele určitý datum nebo čas. Před příchodem HTML5, bylo pro zobrazení takového kalendáře zapotřebí JavaScriptu. S příchodem HTML5 je tato funkce implementována. Avšak většina prohlížečů má stále problémy s jeho zobrazením.

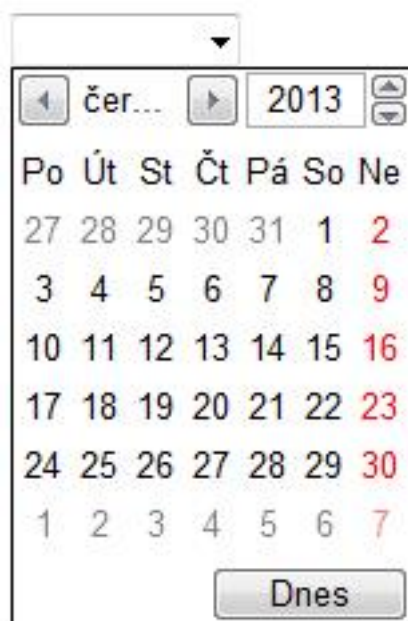
3.1.5.3.2.1 Možné atributy DATE

- **Min a Max** – je možné nastavit rozmezí, ze kterého budeme moci datum vybrat. Pokud nastavíme hodnotu min na 2000-1-1 a hodnotu na 2010-1-1. Automaticky bude definováno, že do pole date budeme zvolit pouze datum v tomto rozmezí.
- **Step** – použitím tohoto atributu specifikujeme, o kolik se má datum krokovat.
- **Value** – nastavíme výchozí hodnotu (datum)
- **Autocomplete** – automaticky nám dává nabídku již jednou zadaných hodnot
- **List** – Nastavením dáváme na výběr z několika možností, které jsou nejčastěji zadávány.
- **Readonly** – znemožní vyplnit formulářové pole
- **Required** – formulář se neodešle, pokud není pole vyplněno. Nelze použít s Value, v takovém případě je hodnota nastavena a required postrádá smysl.

3.1.5.3.2.2 Zdrojový kód DATE









```
<input type="date">
```

3.1.5.3.3 Ukázka zobrazení DATE



Obrázek 23: Ukázka zobrazení DATE v Opere

3.1.5.3.4 Tabulka s podporou DATE

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
date						

Tabulka 27: Podpora input type DATE v prohlížečích

3.1.5.3.3 DATETIME

Možností, jak zadat čas je v HTML5 hned několik, tento input se věnuje převážně zadávání času ve světovém formátu. Bohužel jeho podpora není moc dobrá. Dá se říci, že jediným prohlížečem, který jej zobrazí je Opera. Ostatní prohlížeče jej zobrazují jako pole pro zadávání textu.

3.1.5.3.3.1 Možné atributy DATETIME

- **Min a Max** – je možné nastavit rozmezí, ze kterého budeme moci datum vybrat. Pokud nastavíme hodnotu min na 2000-1-1 a hodnotu na 2010-1-1. Automaticky bude definováno, že do pole date budeme zvolit pouze datum v tomto rozmezí.
- **Step** – použitím tohoto atributu specifikujeme, o kolik se má datum krokovat.
- **Value** – nastavíme výchozí hodnotu (datum)
- **Autocomplete** – automaticky nám dává nabídku již jednou zadaných hodnot
- **List** – Nastavením dáváme na výběr z několika možností, které jsou nejčastěji zadávány.
- **Readonly** – znemožní vyplnit formulářové pole
- **Required** – formulář se neodešle, pokud není pole vyplněno. Nelze použít s Value, v takovém případě je hodnota nastavena a required postrádá smysl.

3.1.5.3.3.2 Zdrojový kód DATETIME

```
<input type="datetime">
```


3.1.5.3.3 Ukázka zobrazení DATETIME



Obrázek 24: Ukázka zobrazení DATETIME

3.1.5.3.4 Tabulka s podporou DATETIME

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
datetime						

Tabulka 28: Podpora input type DATETIME v prohlížečích

3.1.5.3.4 DATETIME-LOCAL

Poslední možností jak nastavit datum a čas je lokální nastavení, tedy nastavení bez časových zón. Ale jak již je zvykem pokud nefunguje základný typ tedy date nebude fungovat ani jeho odnož. I v tomto případě s e musíme spoko-

jit s dvěma prohlížeči a to Google Chrome nebo Opera. Ostatní prohlížeče zobrazí tento input jako text.

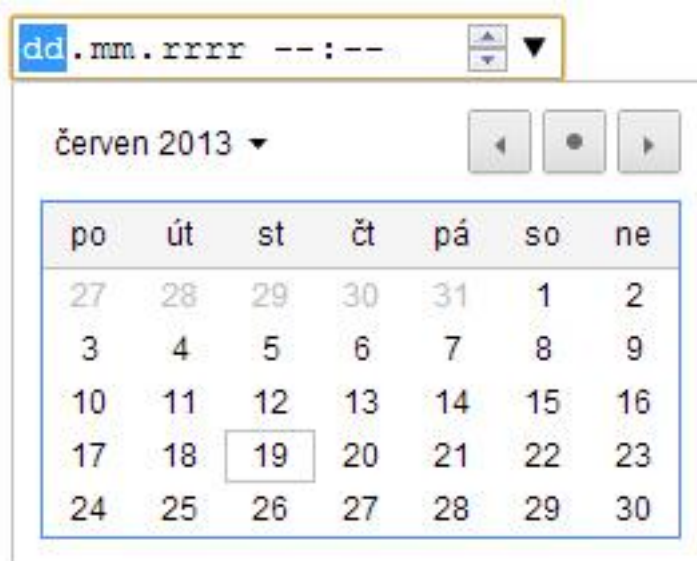
3.1.5.3.4.1 Možné atributy DATETIME-LOCAL

- **Min a Max** – je možné nastavit rozmezí, ze kterého budeme moci datum vybrat. Pokud nastavíme hodnotu min na 2000-1-1 a hodnotu na 2010-1-1. Automaticky bude definováno, že do pole date budeme zvolit pouze datum v tomto rozmezí.
- **Step** – použitím tohoto atributu specifikujeme, o kolik se má datum krokovat.
- **Value** – nastavíme výchozí hodnotu (datum)
- **Autocomplete** – automaticky nám dává nabídku již jednou zadaných hodnot
- **List** – Nastavením dáváme na výběr z několika možností, které jsou nejčastěji zadávány.
- **Readonly** – znemožní vyplnit formulářové pole
- **Required** – formulář se neodešle, pokud není pole vyplněno. Nelze použít s Value, v takovém případě je hodnota nastavena a required postrádá smysl.

3.1.5.3.4.2 Zdrojový kód DATETIME-LOCAL













```
<input type="datetime-local">
```

3.1.5.3.4.3 Ukázka zobrazení DATETIME-LOCAL



Obrázek 25: Ukázka zobrazení DATETIME-LOCAL v Google Chrome

3.1.5.3.4.4 Tabulka s podporou DATETIME-LOCAL

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
datetime-local						

Tabulka 29: Podpora input type DATETIME-LOCAL v prohlížečích

3.1.5.3.5 EMAIL

Pokud jsme v dřívějších verzích HTML chtěli, aby uživatel zadal email. Museli jsme použít složitý JavaScriptový kód, který zajistil, že zadaná hodnota bude odpovídat emailové adrese. HTML5 přináší element typu email, který už v sobě má vše implementované. Veškeré ošetření si prohlížeč kontroluje sám.

Prohlížeč kontroluje, zda je adresa zapsaná syntakticky správně, už však neřeší, zda emailová adresa je funkční popřípadě, zda vůbec existuje.

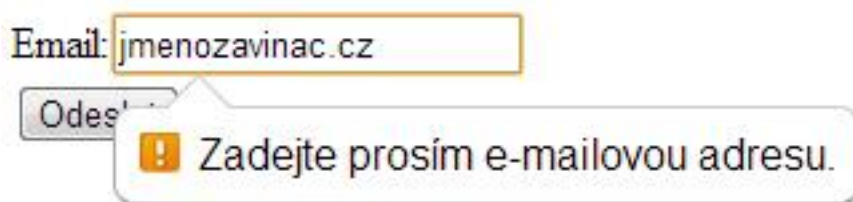
3.1.5.3.5.1 Možné atributy EMAIL

- **Value** – nastavíme výchozí hodnotu (email), která se bude odesílat, aniž by došlo k vyplnění pole formuláře. Postrádá smysl s použitím atributu required.
- **Autocomplete** – automaticky nám dává nabídku již jednou zadaných hodnot
- **List** – Nastavením dáváme na výběr z několika možností, které jsou nejčastěji zadávány.
- **Readonly** – znemožní vyplnit formulářové pole
- **Required** – formulář se neodešle, pokud není pole vyplněno. Nelze použít s Value, v takovém případě je hodnota nastavena a required postrádá smysl.
- **Maxlength** – tímto atributem říkáme, jaká maximální hodnota tedy počet znaků může být to pole email zadán.
- **Multiple** – atribut, který nám dává možnost, zadat více emailových adres do jednoho pole. Použijeme tehdy, pokud formulář bude odesílat emaily na více adres popřípadě jako skryté kopie.
- **Pattern** – tímto atributem určujeme, jak zadaná data mají vypadat. To se provádí pomocí regulárních výrazů. Například pokud budeme chtít zadávat pouze čísla a to přesně 3 zvolíme tento kód: `pattern="[0-9]{3}"`.
- **Placeholder** – slouží jako nápověda, která se zobrazuje, přímo ve formulářovém poli na kterém je aplikovaná. Zobrazuje se šedý text, který ihned po aktivaci pole a zadávání hodnoty mizí.

3.1.5.3.5.2 Zdrojový kód EMAIL













```
<input type="email">
```

3.1.5.3.3 Ukázka zobrazení EMAIL



Obrázek 26: Ukázka zobrazení email v Google Chrome

3.1.5.3.4 Tabulka s podporou EMAIL

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
email						

Tabulka 30: Podpora input type EMAIL v prohlížečích

3.1.5.3.6 MONTH

Dalším způsobem, jak vytvořit v HTML5 kalendář, je pomocí `type=month`. Tímto příkazem získáme kalendář, ze kterého budeme moci získat pouze měsíc a rok. Jak je tento prvek podporován v prohlížečích se můžeme přesvědčit na testování níže.

3.1.5.3.6.1 Možné atributy MONTH

- **Min a Max** – je možné nastavit rozmezí, ze kterého budeme moci datum vybrat. Pokud nastavíme hodnotu min na 2000-1-1 a hodnotu na 2010-1-1. Automaticky bude definováno, že do pole date budeme zvolit pouze datum v tomto rozmezí.

- **Step** – použitím tohoto atributu specifikujeme, o kolik se má datum krokovat.
- **Value** – nastavíme výchozí hodnotu (datum)
- **Autocomplete** – automaticky nám dává nabídku již jednou zadaných hodnot
- **List** – Nastavením dáváme na výběr z několika možností, které jsou nejčastěji zadávány.
- **Readonly** – znemožní vyplnit formulářové pole
- **Required** – formulář se neodešle, pokud není pole vyplněno. Nelze použít s Value, v takovém případě je hodnota nastavena a required postrádá smysl.

3.1.5.3.6.2 Zdrojový kód MONTH









```
<input type="month">
```

3.1.5.3.6.3 Ukázka zobrazení MONTH



Obrázek 27: Ukázka zobrazení MONTH v Google Chrome

3.1.5.3.6.4 Tabulka s podporou MONTH

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
month						

Tabulka 31: Podpora input type MONTH v prohlížečích

3.1.5.3.7 NUMBER

Pokud potřebujeme ve formuláři získávat čísla, je tu užitečný element number. Tento element má několik pomocných atributů, které mohou upravit funkčnost, popřípadě přesněji specifikovat o jaká čísla se bude jednat, jak velký rozsah budeme moci zadat, popřípadě jaká bude nastaven defaultní hodnota. Nyní už se podívejme na jednotlivé funkčnosti.

3.1.5.3.7.1 Možné atributy NUMBER

- **Value** – nastavíme výchozí hodnotu (číslo)
- **Autocomplete** – automaticky nám dává nabídku již jednou zadaných hodnot
- **List** – Nastavením dáváme na výběr z několika možností, které jsou nejčastěji zadávány.
- **Readonly** – znemožní vyplnit formulářové pole
- **Required** – formulář se neodešle, pokud není pole vyplněno. Nelze použít s Value, v takovém případě je hodnota nastavena a required postrádá smysl.

- **Placeholder** – slouží jako nápověda, která se zobrazuje, přímo ve formulářovém poli na kterém je aplikovaná. Zobrazuje se šedý text, který ihned po aktivaci pole a zadávání hodnoty mizí.
- **Min a Max** – je možné nastavit rozmezí, ze kterého budeme moci číslo vybrat. Pokud nastavíme hodnotu min na -10 a hodnotu na 10. Automaticky bude definováno, že do pole number budeme zvolit pouze číslo v tomto rozmezí.
- **Step** – použitím tohoto atributu specifikujeme, o kolik se má číslo krokovat. Defaultní hodnota, je nastaven na 1.

3.1.5.3.7.2 Zdrojový kód NUMBER













```
<input type="number">
```

3.1.5.3.7.3 Ukázka zobrazení NUMBER



Obrázek 28: Ukázka zobrazení NUMBER v Opeře

3.1.5.3.7.4 Tabulka s podporou NUMBER

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
number						

Tabulka 32: Podpora input type NUMBER v prohlížečích

3.1.5.3.8 RANGE

I když by se mohlo zdát, že se jedná o neužitečný prvek, opak je pravdou. Tento input má velké spektrum využití. Jednak se dá použít při různém nastavování hodnot. Například pro použití rezervace počtu osob nebo nastavení času. Ale dá se použít jako přepínač dvou hodnot ON/OFF nebo 0/1. Bohužel podpora prohlížečů je omezena, tudíž praktické využití je zatím v nedohlednu.

3.1.5.3.8.1 Možné atributy RANGE

- **Value** – nastavíme výchozí hodnotu (číslo)
- **Autocomplete** – automaticky nám dává nabídku již jednou zadaných hodnot
- **List** – Nastavením dáváme na výběr z několika možností, které jsou nejčastěji zadávány.
- **Min a Max** – je možné nastavit rozmezí, ze kterého budeme moci číslo vybrat. Pokud nastavíme hodnotu min na -10 a hodnotu na 10. Automaticky bude definováno, že Range budeme moci posouvat pouze od hodnot -10 po hodnotu 10.
- **Step** – použitím tohoto atributu specifikujeme, o kolik se má číslo krocovat. Defaultní hodnota, je nastaven na 1.

3.1.5.3.8.2 Zdrojový kód RANGE











```
<input type="range">
```

3.1.5.3.8.3 Ukázka zobrazení RANGE



Obrázek 29: Ukázka zobrazení RANGE v Google Chrome

3.1.5.3.8.4 Tabulka s podporou RANGE

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
range						

Tabulka 33: Podpora input type RANGE v prohlížečích

3.1.5.3.9 SEARCH

Tento input je skoro k nerozlišení od typu text, jediná změna, která se u některých prohlížečů zobrazí je nepatrný křížek, který se aktivuje po zapsání nějakého textu do pole.

3.1.5.3.9.1 Možné atributy SEARCH

- **Value** – nastavíme výchozí hodnotu
- **Autocomplete** – automaticky nám dává nabídku již jednou zadaných hodnot
- **List** – Nastavením dáváme na výběr z několika možností, které jsou nejčastěji zadávány.
- **Readonly** – znemožní vyplnit formulářové pole
- **Required** – formulář se neodešle, pokud není pole vyplněno. Nelze použít s Value, v takovém případě je hodnota nastavena a required postrádá smysl.
- **Placeholder** – slouží jako nápověda, která se zobrazuje, přímo ve formulářovém poli na kterém je aplikovaná. Zobrazuje se šedý text, který ihned po aktivaci pole a zadávání hodnoty mizí.

- **Maxlength** – při použití je zadávané pole omezeno délkou řetězce. Pokud nastavíme hodnotu na 5, délka zadaného řetězce bude maximálně 5znaků.
- **Pattern** – tímto atributem určujeme, jak zadaná data mají vypadat. To se provádí pomocí regulárních výrazů. Například pokud budeme chtít zadávat pouze velká písmena a to přesně o délce 5, zvolíme tento kód: `pattern="[A-Z]{5}"`.
- **Placeholder** – slouží jako nápověda, která se zobrazuje, přímo ve formulářovém poli na kterém je aplikovaná. Zobrazuje se šedý text, který ihned po aktivaci pole a zadávání hodnoty mizí.

3.1.5.3.2 Zdrojový kód SEARCH











```
<input type="search">
```

3.1.5.3.3 Ukázka zobrazení SEARCH



Obrázek 30: Ukázka zobrazení SEARCH v Opěře

3.1.5.3.4 Tabulka s podporou SEARCH

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
search						

Tabulka 34: Podpora input type SEARCH v prohlížečích

3.1.5.3.10 TEL

Telefonní číslo, je jedním ze základních prvků snad většiny formulářů. HTML5 přináší speciální input typu tel, který je určený speciálně pro zadávání telefonních čísel.

3.1.5.3.10.1 Možné atributy TEL

- **Value** – nastavíme výchozí hodnotu
- **Autocomplete** – automaticky nám dává nabídku již jednou zadaných hodnot
- **List** – Nastavením dáváme na výběr z několika možností, které jsou nejčastěji zadávány.
- **Readonly** – znemožní vyplnit formulářové pole
- **Required** – formulář se neodešle, pokud není pole vyplněno. Nelze použít s Value, v takovém případě je hodnota nastavena a required postrádá smysl.
- **Placeholder** – slouží jako nápověda, která se zobrazuje, přímo ve formulářovém poli na kterém je aplikovaná. Zobrazuje se šedý text, který ihned po aktivaci pole a zadávání hodnoty mizí.
- **Maxlength** – při použití je zadávané pole omezeno délkou řetězce. Pokud nastavíme hodnotu na 5, délka zadaného řetězce bude maximálně 5znaků.
- **Pattern** – tímto atributem určujeme, jak zadaná data mají vypadat. To se provádí pomocí regulárních výrazů. Například pokud budeme chtít zadávat pouze velká písmena a to přesně o délce 5, zvolíme tento kód: `pattern="[A-Z]{5}"`.
- **Placeholder** – slouží jako nápověda, která se zobrazuje, přímo ve formulářovém poli na kterém je aplikovaná. Zobrazuje se šedý text, který ihned po aktivaci pole a zadávání hodnoty mizí.

3.1.5.3.10.2 Zdrojový kód TEL







```
<input type="tel">
```

3.1.5.3.10.3 Ukázka zobrazení TEL

Telefon:

Obrázek 31: Ukázka zobrazení TEL v Google Chrome

3.1.5.3.10.4 Tabulka s podporou TEL

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
tel						

Tabulka 35: Podpora input type TEL v prohlížečích

3.1.5.3.11 TIME

Získávání času není příliš populární ve formulářích, ale nějaké využití si určitě najde. Bohužel podpora většiny prohlížečů není příliš dobrá. Pouze dva prohlížeče tento typ podporují.

3.1.5.3.11.1 Možné atributy TIME

- **Min a Max** – je možné nastavit rozmezí, ze kterého budeme moci čas vybrat. Pokud nastavíme hodnotu min na 02:00:00 a hodnotu na 12:00:00. Automaticky bude definováno, že do pole time budeme zvolit pouze čas v tomto rozmezí.
- **Step** – použitím tohoto atributu specifikujeme, o kolik se má čas krokovat.

- **Value** – nastavíme výchozí hodnotu (čas)
- **Autocomplete** – automaticky nám dává nabídku již jednou zadaných hodnot
- **List** – Nastavením dáváme na výběr z několika možností, které jsou nejčastěji zadávány.
- **Readonly** – znemožní vyplnit formulářové pole
- **Required** – formulář se neodešle, pokud není pole vyplněno. Nelze použít s Value, v takovém případě je hodnota nastavena a required postrádá smysl.

3.1.5.3.11.2 Zdrojový kód TIME













```
<input type="time">
```

3.1.5.3.11.3 Ukázka zobrazení TIME



Obrázek 32: Ukázka zobrazení TIME v Opeře

3.1.5.3.11.4 Tabulka s podporou TIME

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
time						

Tabulka 36: Podpora input type TIME v prohlížečích

3.1.5.3.12 URL

Jak už název napovídá, tento input bude sloužit pro zadávání URL adres.

3.1.5.3.12.1 Možné atributy URL

- **Value** – nastavíme výchozí hodnotu
- **Autocomplete** – automaticky nám dává nabídku již jednou zadaných hodnot
- **List** – Nastavením dáváme na výběr z několika možností, které jsou nejčastěji zadávány.
- **Readonly** – znemožní vyplnit formulářové pole
- **Required** – formulář se neodešle, pokud není pole vyplněno. Nelze použít s Value, v takovém případě je hodnota nastavena a required postrádá smysl.
- **Placeholder** – slouží jako nápověda, která se zobrazuje, přímo ve formulářovém poli na kterém je aplikovaná. Zobrazuje se šedý text, který ihned po aktivaci pole a zadávání hodnoty mizí.
- **Maxlength** – při použití je zadávané pole omezeno délkou řetězce. Pokud nastavíme hodnotu na 5, délka zadaného řetězce bude maximálně 5znaků.
- **Pattern** – tímto atributem určujeme, jak zadaná data mají vypadat. To se provádí pomocí regulárních výrazů. Například pokud budeme chtít zadávat pouze velká písmena a to přesně o délce 5, zvolíme tento kód: `pattern="[A-Z]{5}"`.
- **Placeholder** – slouží jako nápověda, která se zobrazuje, přímo ve formulářovém poli na kterém je aplikovaná. Zobrazuje se šedý text, který ihned po aktivaci pole a zadávání hodnoty mizí.

3.1.5.3.12.2 Zdrojový kód URL






```
<input type="url">
```

3.1.5.3.12.3 Ukázka zobrazení URL



Obrázek 33: Ukázka zobrazení URL v Mozilla Firefox

3.1.5.3.12.4 Tabulka s podporou URL

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
url						

Tabulka 37: Podpora input type URL v prohlížečích

3.1.5.3.13 WEEK

Poslední možností jak získat určitý datum je type=week, tento typ elementu input nám získá informaci o požadovaném týdnu v roce. Bohužel v některých prohlížečích je stále nepodporován a data získaná z tohoto kalendáře nejsou žádným způsobem ošetřena.

3.1.5.3.13.1 Možné atributy WEEK

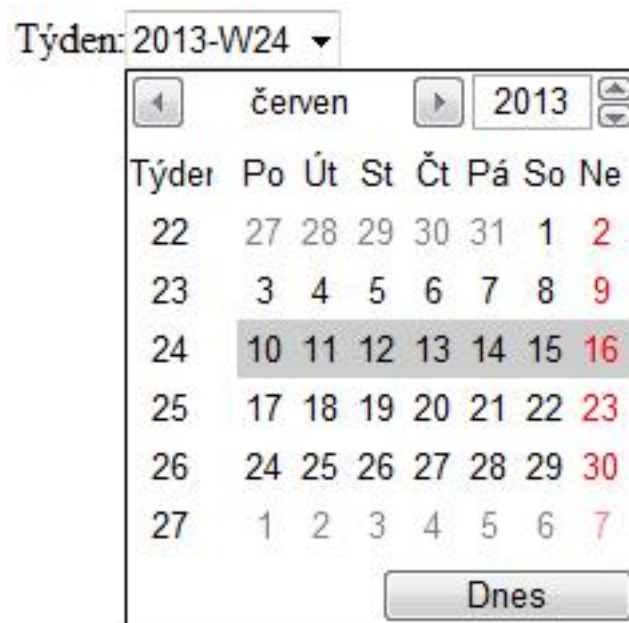
- **Min a Max** – je možné nastavit rozmezí, ze kterého budeme moci datum vybrat. Pokud nastavíme hodnotu min na 2000-1-1 a hodnotu na 2010-1-1. Automaticky bude definováno, že do pole week budeme zvolit pouze datum v tomto rozmezí.
- **Step** – použitím tohoto atributu specifikujeme, o kolik se má datum krokovat.
- **Value** – nastavíme výchozí hodnotu (datum)

- **Autocomplete** – automaticky nám dává nabídku již jednou zadaných hodnot
- **List** – Nastavením dáváme na výběr z několika možností, které jsou nejčastěji zadávány.
- **Readonly** – znemožní vyplnit formulářové pole
- **Required** – formulář se neodešle, pokud není pole vyplněno. Nelze použít s Value, v takovém případě je hodnota nastavena a required postrádá smysl.

3.1.5.3.13.2 Zdrojový kód WEEK













```
<input type="week">
```

3.1.5.3.13.3 Ukázka zobrazení WEEK



Obrázek 34: Ukázka zobrazení WEEK v Opeře

3.1.5.3.13.4 Tabulka s podporou WEEK

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
week						

Tabulka 38: Podpora input type WEEK v prohlížečích

3.1.5.4 Starší Input type v předchozích verzích HTML

3.1.5.4.1 BUTTON

Nejpoužívanější prvek každého formuláře, avšak tento prvek není nutný. Formulář lze odeslat i pomocí jiných prvků.

3.1.5.4.1.1 Možné atributy BUTTON

- **Value** – jediný povolený a vyžadovaný atribut

3.1.5.4.1.2 Zdrojový kód BUTTON













```
<button type="button">Odeslat</button>
```

3.1.5.4.1.3 Ukázka zobrazení BUTTON



Obrázek 35: Ukázka zobrazení BUTTON v Internet Explorer 10

3.1.5.4.1.4 Tabulka s podporou BUTTON

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
button						

Tabulka 39: Podpora input type BUTTON v prohlížečích

3.1.5.4.2 SELECT

Podobný typ jako datalist, na rozdíl od datalistu nelze vkládat vlastní hodnotu. Ta je omezena pouze na předepsaný výběr, je možné vybrat jednu nebo více možností. Příklad zobrazuje dny v týdnu.

3.1.5.4.3 Možné atributy SELECT

- **Required** – ošetření odeslání formuláře. Pokud je použit tento atribut formuláře se neodešle, pokud nebude daný prvek vyplněn.
- **Name** – pojmenování daného prvku
- **Multiple** – je možné vybrat více možností.
- **Size** – parametr tohoto atributu je číslo a určuje velikost zobrazených položek.
- **Disable** – atribut který z neaktivní daný prvek.
- **Form** – parametr je id formu pod který daný prvek patří. Díky tomuto atributu může být prvek umístěn mimo formulář, ke kterému patří.
- **Autofocus** – pokud použijeme tento atribut po načtení ky, je ihned kurzor aktivní na daném prvku.

3.1.5.4.3.1 Zdrojový kód SELECT













```
<select name="den">
  <option value="1">Pondělí</option>
  <option value="2">Úterý</option>
  <option value="3">Středa</option>
  <option value="4">Čtvrtek</option>
  <option value="5">Pátek</option>
  <option value="6">Sobota</option>
  <option value="7">Neděle</option>
</select>
```

3.1.5.4.3.2 Ukázka zobrazení SELECT



Obrázek 36: Ukázka zobrazení SELECT v Google Chrome

3.1.5.4.3.3 Tabulka s podporou SELECT

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
select						

Tabulka 40: Podpora input type SELECT v prohlížečích

3.1.5.4.4 CHECKBOX

Checkbox použijeme všude tam, kde potřebujeme od uživatele zjistit nějaký stav nebo výběr nějaké možnosti. Dole na příkladu je checkbox, použit jako zaškrťávací pole, které získává od uživatele, kolik služeb objednává.

3.1.5.4.4.1 Možné atributy CHECKBOX

- **Value** – povinný atribut pro checkbox. Určuje hodnotu, která se odešle při zaškrtnutí checkboxu.
- **Required** – ošetření odeslání formuláře. Pokud je použit tento atribut formuláře se neodešle, dokud nebude vybrána nějaká hodnota checkboxu.

3.1.5.4.4.2 Zdrojový kód CHECKBOX

```
<label>Sauna</label><input type="checkbox" value="sauna">  
<label>Bazén</label><input type="checkbox" value="bazen">  
<label>Masáž</label><input type="checkbox" value="masaz">
```

3.1.5.4.4.3 Ukázka zobrazení CHECKBOX

Sauna Bazén Masáž

Obrázek 37: Ukázka zobrazení CHECKBOX v Mozilla Firefox

3.1.5.4.4.4 Tabulka s podporou CHECKBOX

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
checkbox						

Tabulka 41: Podpora input type CHECKBOX v prohlížečích

3.1.5.4.5 FILE

Známí a velice používaný typ file, využijeme všude tam, kde potřebujeme spolu s formulářem odeslat ještě nějaká jiná data, nejčastěji nějaký obrázek či dokument.

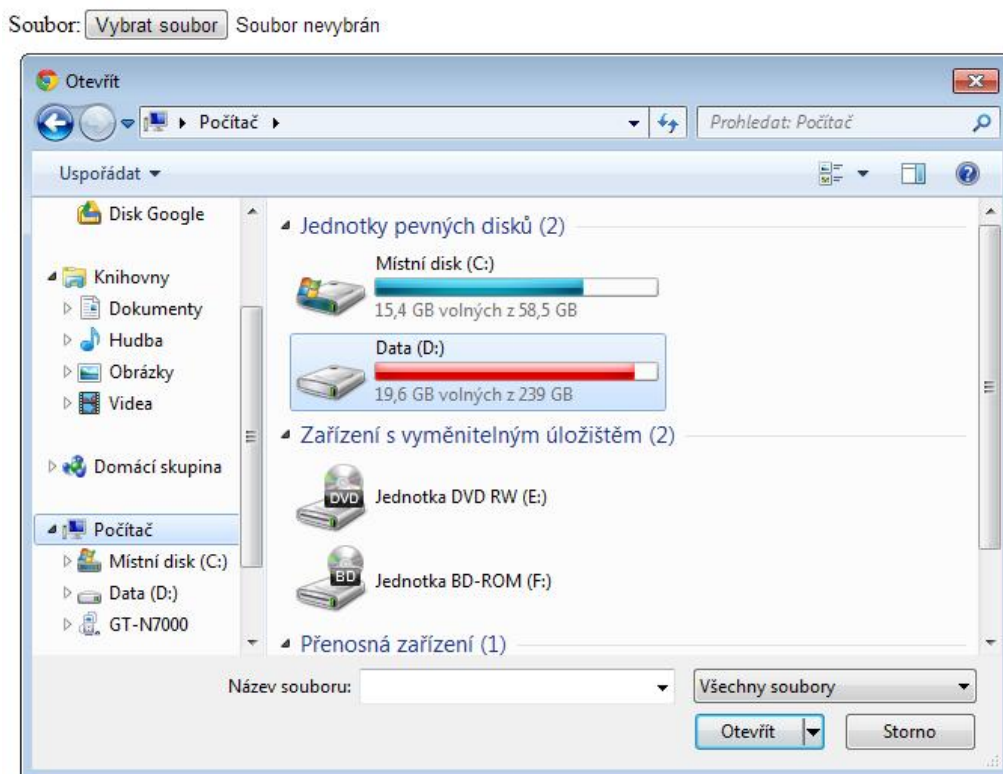
3.1.5.4.5.1 Možné atributy FILE

- **Value** – povinný atribut pro checkbox. Určuje hodnotu, která se odešle při zaškrtnutí checkboxu.
- **Required** – ošetření prvku file, pokud nedojde k vybrání nějakého zvoleného souboru, formulář se neodešle.
- **Accept** – specifikuje načtená data, více v kapitole Atribut Accept.
- **Multiple** – je možné vybrat více souborů.

3.1.5.4.5.2 Zdrojový kód FILE













```
<label>Vyberte soubor:</label>  
<input type="file" name="soubor">
```

3.1.5.4.5.3 Ukázka zobrazení FILE



Obrázek 38: Ukázka zobrazení FILE v Google Chrome

3.1.5.4.5.4 Tabulka s podporou FILE

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
file						

Tabulka 42: Podpora input type FILE v prohlížečích

3.1.5.4.6 HIDDEN

Tímto příkazem můžeme skrýt jakýkoli jiný prvek. To znamená, že bude neviditelný a uživatel ho nebude moci upravovat ani žádným jiným způsobem nastavovat jeho hodnotu. Na příkladu níže vidíme, že obsahuje dva inputy typu text, jeden z inputů má nastavenou hodnotu hidden, je tedy neviditelný a jeho obsah nepodléhá validaci.

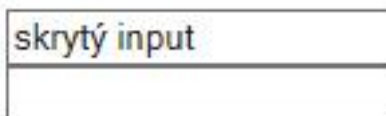
3.1.5.4.6.1 Možné atributy HIDDEN

- **Value** – povinný atribut pro hidden. Určuje hodnotu, která se odešle u tohoto skrytého prvku.

3.1.5.4.6.2 Zdrojový kód HIDDEN













```
<input type="text" value="skrytý input" hidden>  
<input type="text" >
```

3.1.5.4.6.3 Ukázka zobrazení HIDDEN



Obrázek 39: Ukázka špatného zobrazení v Internet Explorer10

3.1.5.4.6.4 Tabulka s podporou HIDDEN

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
hidden						

Tabulka 43: Podpora input type HIDDEN v prohlížečích

3.1.5.4.7 PASSWORD

Tento input je určen pro část formuláře, kde uživatel vkládá citlivá data, popřípadě data typu heslo. Funkce je stejná jako u inputu typu text, avšak rozdíl je v tom, že zadávaná hodnota je skryta, místo zadávaných znaků se zobrazuje zástupný znak (nejčastěji tečka).

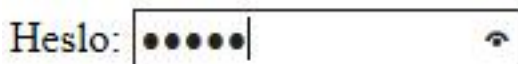
3.1.5.4.7.1 Možné atributy PASSWORD

- **Value** – nastavíme výchozí hodnotu
- **Autocomplete** – automaticky nám dává nabídku již jednou zadaných hodnot
- **Readonly** – znemožní vyplnit formulářové pole
- **Required** – formulář se neodešle, pokud není pole vyplněno. Nelze použít s Value, v takovém případě je hodnota nastavena a required postrádá smysl.
- **Placeholder** – slouží jako nápověda, která se zobrazuje, přímo ve formulářovém poli na kterém je aplikovaná. Zobrazuje se šedý text, který ihned po aktivaci pole a zadávání hodnoty mizí.
- **Maxlength** – při použití je zadávané pole omezeno délkou řetězce. Pokud nastavíme hodnotu na 5, délka zadaného řetězce bude maximálně 5znaků.
- **Pattern** – tímto atributem určujeme, jak zadaná data mají vypadat. To se provádí pomocí regulárních výrazů. Například pokud budeme chtít zadávat pouze velká písmena a to přesně o délce 5, zvolíme tento kód: `pattern="[A-Z]{5}"`.

3.1.5.4.7.2 Zdrojový kód PASSWORD









```
<input type="password"/>
```

3.1.5.4.7.3 Ukázka zobrazení PASSWORD



Obrázek 40: Ukázka zobrazení PASSWORD v Internet Explorer 10

3.1.5.4.7.4 Tabulka s podporou PASSWORD

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
password						

Tabulka 44: Podpora input type PASSWORD v prohlížečích

3.1.6 DATALIST

Datalist je takový malý pomocník. Pokud potřebujeme od uživatele zjistit nějakou hodnotu, která může nabývat mnoha hodnot, ale ve většině případů každý zadá podobnou, můžeme těmto uživatelům pomoci datalistem, ten nám dá na výběr nejpoužívanější možnosti a zároveň pokud se hodnota nebude ukazovat v listu, může uživatel zadat svou vlastní. Příklad uvedený níže reprezentuje list nejčastějších vyhledávačů.

3.1.6.1 Možné atributy elementu DATALIST

Datalist nemá žádné možné atributy. Lze použít pouze globální atributy. Viz kapitola o Globálních atributech.

3.1.6.2 Zdrojový kód DATALIST

```
<input list="vyhledavace">
<datalist id="vyhledavace">
  <option value="Opera">
```

```

<option value="Google Chrome">
<option value="Internet Explorer 7">
<option value="Internet Explorer 8">
<option value="Internet Explorer 9">
<option value="Internet Explorer 10">
</datalist>
<input type="submit">

```

3.1.6.3 Ukázka zobrazení DATALIST



Obrázek 41: Ukázka zobrazení DATALIST v Google Chrome

3.1.6.4 Tabulka s podporou DATALIST

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
datalist						

Tabulka 45: Podpora input type DATALIST v prohlížečích

3.1.7 OPTGROUP

Tento prvek je součástí dvou prvků a to select a datalist, jedná se o tzv. seskupování možností option. Viz příklad níže.

3.1.7.1 Možné atributy elementu OPTGROUP

- **Disable** - pokud zadáme tento atribut, bude možnost zadávání vypnuta.
- **Label** – Povinný atribut, který slouží k tzv. seskupování určitých možností pod stejné typy.

3.1.7.2 Zdrojový kód OPTGROUP

```
<select name="Motorek">
  <optgroup label="Super Sport">
    <option value="1">Honda CBR</option>
    <option value="2">Yamaha YZF</option>
    <option value="3">Suzuki GSX-R</option>
    <option value="4">Kawasaki Ninja</option>
  </optgroup>
  <optgroup label="Cestovní">
    <option value="6">Honda VFR</option>
    <option value="7">Yamaha XV</option>
    <option value="6">Suzuki GSR</option>
    <option value="7">Kawasaki Z</option>
  </optgroup>
</select>
```

3.1.7.3 Ukázka zobrazení OPTGROUP



Obrázek 42: Ukázka zobrazení OPTGROUP v Opeře

3.1.7.4 Tabulka s podporou OPTGROUP

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
outgroup						

Tabulka 46: Podpora input type OPTGROUP v prohlížečích

3.1.8 OPTION

Option je součástí dvou prvků, select a datalist. Jedná se o možnosti, které nabízí tyto dva prvky.

3.1.8.1 Možné atributy elementu OPTION

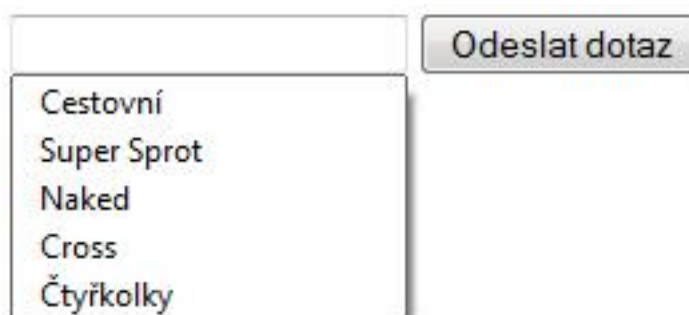
- **Disable** - pokud zadáme tento atribut, bude možnost vypnuta

- **Label** – Povinný atribut, který slouží k tzv. seskupování určitých možností pod stejné typy.
- **Value** – nastaví výchozí hodnotu. Jedná se o povinný atribut.
- **Selected** – Pokud je tento atribut u určitého prvku je tento prvek vybrán. A uživatel jej nemusí vybírat ručně.

3.1.8.2 Zdrojový kód OPTION



```
<input list="motoriky">  
<datalist id="motoriky">  
  <option value="Cestovní">  
  <option value="Super Sprot">  
  <option value="Naked">  
  <option value="Cross">  
  <option value="Čtyřkolky">  
</datalist>
```

3.1.8.3 Ukázka zobrazení OPTION



Obrázek 43: Ukázka zobrazení OPTION v Mozilla Firefox

3.1.8.4 Tabulka s podporou OPTION

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
option						

Tabulka 47: Podpora input type OPTION v prohlížečích

3.1.9 TEXTAREA

Tento formulářový prvek zná každý programátor, jedná se o pole, které je určené pro psaní delšího řetězce textu. Takovým hlavním příkladem jsou formulářové poznámky. Které obsahuje snad každý formulář.

3.1.9.1 Možné atributy elementu TEXTAREA

- **Autofocus** – po načtení formuláře je ihned aktivní kurzor na tomto poli.
- **Cols** – Počet sloupců jedná se tedy o počet znaků na řádek. Toto nastavení je sice možné udělat pomocí tohoto atributu, avšak je lepší to řešit pomocí CSS.
- **Disabled** – Pokud je tento atribut uveden je pole neaktivní a nelze do něj nic vkládat.
- **Form** – Tento element může být definován mimo formulář a následně pomocí tohoto atributu přidělen zpět.
- **Maxlength** – maximální možný řetězec znaků, který může být do pole zadán.
- **Name** – Určuje jméno elementu. Tento atribut musí nabývat vždy unikátních hodnot.

- **Placeholder** – Nápopvěda, která se zobrazí jako šedý text přímo v tomto elementu.
- **Readonly** – pokud je uveden u elementu nelze tento element upravovat.
- **Required** – Určuje podmínku, která neodešle formulář, pokud nebude pole vyplněno.
- **Rows** – Podobně jako Cols určuje Rows počet řádků. Lepší řešení jak element upravit je pomocí CSS.
- **Wrap** – tento atribut nabývá dvou hodnot hard a soft, kde soft udává, že se text zalomuje pouze pro zobrazení a po odeslání je text nezalomený. Hard zalomuje text i pro odeslání.

3.1.9.2 Zdrojový kód TEXTAREA

```
<textarea name="komentar" rows="10" cols="50"></textarea>
```

3.1.9.3 Ukázka zobrazení TEXTAREA



Obrázek 44: Ukázka zobrazení TEXTAREA v Internet Explorer 10

3.1.9.4 Tabulka s podporou TEXTAREA

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
textarea						

Tabulka 48: Podpora input type TEXTAREA v prohlížečích

3.1.10 KEYGEN

Velice užitečný tag, který bude nedílnou součástí každého e-shopu. Pomocí tohoto tagu odesíláme data šifrovaně. Požadované pole vygeneruje dva klíče, jeden privátní, který je uložen lokálně a druhý veřejný, který se odesílá na server. Velice dobré například pro ukládání a odesílání hesel. Viz ukázka níže.

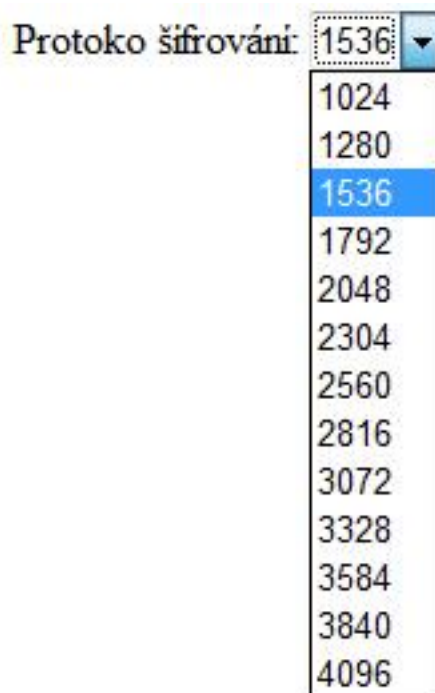
3.1.10.1 Možné atributy elementu KEYGEN

- **Autofocus** – po načtení formuláře je ihned aktivní kurzor na tomto poli.
- **Disabled** – Pokud je tento atribut uveden je pole neaktivní a nelze do něj nic vkládat.
- **Form** – Tento element může být definován mimo formulář a následně pomocí tohoto atributu přidělen zpět.
- **Name** – Určuje jméno elementu. Tento atribut musí nabývat vždy unikátních hodnot.
- **Keytype** – Jedná se o atribut, který určuje algoritmus k šifrování. Nabývá hodnot rsa, dsa, ec.

3.1.10.2 Zdrojový kód KEYGEN





```
<keygen name="key">
```

3.1.10.3 Ukázka zobrazení KEYGEN



Obrázek 45: Ukázka zobrazení KEYGEN v Opeře

3.1.10.4 Tabulka s podporou KEYGEN

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
keygen						

Tabulka 49: Podpora input type KEYGEN v prohlížečích

3.1.11 OUTPUT

Output slouží k aktuálnímu výpočtu hodnot ve formuláři. V příkladu níže je ukázka outputu, který představuje sčítání dvou čísel. Čísla se zadávají do dvou number typů a výsledek je právě prvek output.

3.1.11.1 Možné atributy elementu OUTPUT


- **For** – nepovinný atribut, který určuje id elementu, ke kterému output patří.
- **Form** – Tento element může být definován mimo formulář a následně pomocí tohoto atributu přidělen zpět.
- **Name** – Určuje jméno elementu. Tento atribut musí nabývat vždy unikátních hodnot.

3.1.11.2 Zdrojový kód OUTPUT

```
<form
oninput="x.value=parseInt(a.value)+parseInt(b.value)">
  <input type="number" id="a" value="50">+
  <input type="number" id="b" value="50">=
  <output name="x"></output>

</form>
```












3.1.11.3 Ukázka zobrazení OUTPUT



20 + 33 = 53

Obrázek 46: Ukázka zobrazení OUTPUT v Mozilla Firefox

3.1.11.4 Tabulka s podporou OUTPUT

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
output						

Tabulka 50: Podpora input type OUTPUT v prohlížečích

3.1.12 PROGRESS

Co to vlastně progress je? Jedná se o ukazatel průběhu. Určitě jej všichni známe z webových stránek, kdy čekáme na určité načtení nějakého stránky prvku nebo aplikace. Jedná se o ukazatel zpracování. Tento prvek se musel dříve složitě programovat pomocí JavaScriptu. Dnes v HTML5 stačí použít jednoduchý příkaz a progress bar máme ihned. Tento prvek nabývá dvou hodnot. Max, tedy maximální hodnota pro zpracování převážně se bude používat hodnota 100, jako procenta, a dále hodnota value, tedy aktuální hodnota zpracování.

3.1.12.1 Možné atributy elementu PROGRESS

- **Value** – Výchozí hodnota.
- **Max** – Maximální možná zadaná hodnota.

3.1.12.2 Zdrojový kód PROGRESS

```
<progress value="22" max="100"></progress>
```

3.1.12.3 Ukázka zobrazení PROGRESS



Obrázek 47: Ukázka zobrazení PROGRESS v Google Chrome

3.1.12.4 Tabulka s podporou PROGRESS

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
progress						

Tabulka 51: Podpora input type PROGRESS v prohlížečích

3.1.13 METER

Tento prvek můžeme použít všude tam, kde potřebujeme na jednoduchém grafu ukázat například naplnění nějakého kurzu. Víme, že do kurzu, se vejde maximálně 20 lidí tedy max hodnota je rovna 20. minimální počet lidí je 0 a hodnotou value, udáváme aktuální stav kurzu.

3.1.13.1 Možné atributy elementu METER

- **Value** – Výchozí hodnota.
- **Max** – Maximální možná zadaná hodnota.
- **Min** – Minimální možná hodnota.
- **Optimum** – optimální hodnota
- **Form** – Nepovinný atribut, tímto atributem určujeme formulář, do kterého tento element patří.
- **High** – Určuje mez, od které je hodnota již brána jako vysoká
- **Low** – Určuje mez, od které je již hodnota brána jako nízká

3.1.13.2 Zdrojový kód METER













```
<meter min="0" value="8" max="20" high="15"> </meter>
```

3.1.13.3 Ukázka zobrazení METER



Obrázek 48: Ukázka zobrazení METER v Mozilla Firefox

3.1.13.4 Tabulka s podporou METER

						
	Google Chrome	Opera	Mozilla Firefox	Internet Explorer 8	Internet Explorer 9	Internet Explorer 10
meter						

Tabulka 52: Podpora input type METER v prohlížečích

3.2 Globální atributy v HTML5

Globální neboli obecné atributy lze použít u všech elementů bez omezení.

3.2.1 ACCESSKEY

Tento atribut definuje klávesovou zkratku pro určitý element a je velice dobře znám z dřívějších verzí HTML, usnadňuje práci lidem s handicapem. Pokud určitému elementu nastavíme accesskey a jeho parametr bude ="1". Zdrojový kód by vypadal následovně:

```
accesskey="1"
```

Říkáme tím prohlížeči, že na klávesové zkratce ALT+1 aktivuje daný element.

3.2.2 CLASS

Class je znám všem programátorům HTML a nejen HTML. Definuje třídu, do které element patří. V HTML se nejčastěji atributu class používá k stylování pomocí CSS. Veškerá data jsou uložena v externím souboru a pomocí class se odkazujeme na určitou třídu, která obsahuje kaskádový styl pro daný element. Odkaz na třídu by mohl vypadat následovně.

```
class="label_jmeno"
```

3.2.3 CONTENTEDITABLE

Tento atribut má za úkol specifikovat, zda daný element můžeme dále editovat. Nabývá dvou hodnot true a false. Kód vypadá následovně:

```
contenteditable="true"
```

3.2.4 CONTEXTMENU

Umožňuje vytvořit vlastní nabídku v menu, které vyskočí na pravé tlačítko myši. Pokud to použijeme na nějaký element a uživatel klikne na pravé tlačítko myši, zobrazí se v této nabídce požadovaná možnost.

3.2.5 DIR

Tento atribut určuje směr textu. Nabývá dvou hodnot ltr a rtl. Buďto je text zleva doprava nebo zprava doleva.

3.2.6 DRAGABLE

Atribut definuje, zda je možné určitý element, který tento atribut obsahuje přesouvat (přemísťovat). Nabývá tedy dvou hodnot true a false.

3.2.7 HIDDEN

Tento atribut nenabývá žádné hodnoty. Definuje, zda je daný prvek aktivní (viditelný).

3.2.8 ID

Tímto atributem definujeme unikátní označení (klíč) určitému elementu. Nesmí být prázdný řetězec.

3.2.9 LANG

Atribut, pomocí kterého je možné definovat jazyk elementu a atributů. Jedná se o neprázdný řetězec, který nabývá několika hodnot. Možné hodnoty, lze najít na tomto odkazu:

<http://www.w3.org/TR/html5/references.html#refsBCP47>

3.2.10 SPELLCHECK

Velice zajímavý atribut, který slouží ke kontrole gramatiky určitého elementu. Velice užívaný například u formulářů. Nabývá hodnot true, false a default.

3.2.11 TABINDEX

Určuje pořadí focusu, tedy v jakém pořadí bude docházet k přecházení ne určitý element pomocí tabulátoru. Hodnoty tabindexu jsou vždy kladná celá čísla. Pokud zadáme zápornou hodnotu říkáme tím prohlížeči, že zadaný element nemůže získat focus pomocí tabulátoru.

3.2.12 TITLE

Atribut, pomocí něhož dochází k přesnějšímu specifikování elementu, přesněji lze říci, že tento prvek rozšiřuje označení elementu.

4 Ukázkový formulář

Ukázkový formulář je vytvořen pro rezervaci ubytování v malém penzionu. V tomto formuláři jsou použity základní avšak nejpoužívanější prvky webových formulářů. Při získávání informací si musíme uvědomit, jaká data jsou pro nás nejdůležitější a jaká data o potenciálním zákazníkovi potřebujeme znát. Není důležité znát vše. Velké a obsáhlé formuláře mnohdy zákazníka od vyplňování odradí a raději se poohlédnou jinde. Proto je potřeba formulář vytvořit tak, aby zákazník nebyl otráven vyplňováním, a nám získaná data postačí pro přesnou identifikaci klienta. Ukázkový formulář má tedy následující strukturu:

- Jméno a Příjmení
- Telefonní číslo (mobil nebo pevná linka)
- Email
- Datum příjezdu
- Datum odjezdu
- Počet ubytovaných dospělých osob
- Počet ubytovaných dětí
- ListBox pro volbu pokoje (apartmán, 1- lůžkový, atd.)
- Checkbox pro volbu domácího mazlíčka
- Čas příjezdu
- Čas odjezdu
- RadioButton pro volbu platby (hotově, platba převodem, PayPal, atd.)
- ListBox pro volbu penze (plná penze, polopenze)
- Poznámky

4.1 Validace formuláře

Díky novým formulářovým prvkům nebylo pro validování vzorového formuláře použito žádného JavaScriptového kódu. Každý prvek je ošetřen pomocí svých atributů.

4.1.1 Jméno a příjmení

Toto pole by mělo obsahovat pouze křestní jméno a příjmení, je nežádoucí, aby uživatel zadával nějaké jiné znaky. Jméno a příjmení může obsahovat pouze písmena. Dále bylo vhodné omezit délku možných znaků zadaných v poli, aby nedocházelo k zatěžování databáze. A jelikož jméno je jeden z nejdůležitějších údajů, formulář by se neměl odeslat bez vyplněného pole.

4.1.1.1 Použité atributy

- **Maxlength** – atribut pro ošetření délky řetězce
- **Pattern** – atribut s parametrem "[a-Z]", kterým ošetřujeme zadání pouze malých a velkých písmen.
- **Required** – díky tomuto atributu, se formulář neodešle, pokud bude pole prázdné.

4.1.2 Telefon

Pole telefon, by se nemuselo už dále upravovat. Pokud použijeme správný typ, tedy input typu tel, nemusíme použít žádné atributy a pole je ošetřené pro telefonní čísla. Bohužel formát telefonů, se v různých zemích stále liší. Bude me tedy předpokládat, že do tohoto formuláře budou zadávat údaje pouze lidé z České republiky.

4.1.2.1 Použité atributy

- **Pattern** – díky regulárním výrazům ošetříme pole pro zadání telefonu v mezinárodním formátu.

- **Placeholder** – atribut nápovědy dává uživateli nápovědu, že má zadat telefon v mezinárodním formátu.
- **Maxlength** – maximální délka telefonního čísla v ČR je 13 znaků, tento atribut tedy ošetří maximum zadaných znaků
- **Required** – v tomto případě je také telefon nezbytný pro odeslání formuláře.

4.1.3 Email

Email je pole, které je pro všechny země stejné a přímo pro toto pole je určen input typu email. Proto jsou ve formuláři použity pouze dva atributy,

4.1.3.1 Použité atributy

- **Placeholder** – nápověda, která uživateli říká, že má zadat email.
- **Required** – i tento prvek je nezbytný pro odeslání formuláře.

4.1.4 Datum

V tomto formuláři jsou použity dva input typu date. Jeden pro zvolení data příjezdu a druhý na zvolení data odjezdu. Atribut obsahují oba a to pouze jeden.

4.1.4.1 Použité atributy

- **Min a max** – atribut min je použit u inputu pro zadání data příjezdu a max je použit pro zvolení data odjezdu.

4.1.5 Počet ubytovaných osob

Počet ubytovaných osob nemůže přesáhnout celkovou kapacitu penzionu, proto musí být nastavena maximální možná hodnota reprezentující maximální počet osob. Druhý input obsahuje počet dětí, jelikož děti nemůžou být ubytovány samotné, musí zde být alespoň jedna dospělá osoba, bude hodnota max o 1 menší než u počtu osob. Pole pro zadání počtu osob bude obsahovat i hod-

notu min, který bude nastavena na 1. Rezervace nemůže být provedena s hodnotou 0 osob.

4.1.5.1 Použité atributy

- **Max a min** – atribut použit pro zadání maximálního počtu ubytovaných osob, která nesmí přesáhnout kapacitu penzionu. A minimální hodnota v případě ubytovaných osob.

4.1.6 Vyberte pokoj

V tomto případě je použit input typu text s atributem list. Kde si uživatel bude moci vybrat z několika přednastavených variant.

4.1.7 Vyberte penzi

Pro zvolení penze je také použit input typu text s atributem list, který má díky datalistu přednastavené možnosti, ze kterých si zákazník může vybírat.

4.1.8 Checkbox

Checkbox pro zvolení domácího mazlíčka nemusí být nijak ošetřen je zde pouze jeden a plní funkci ANO/NE.

4.1.9 Čas příjezdu a odjezdu

Penzion je otevřen s obsluhou pouze v určitém čase, proto musíme uživatele omezit ve výběru času. A to hodnotou min a max. Kdy hodnota min reprezentuje hodinu příchodu obsluhy a hodnota max čas odchodu obsluhy.

4.1.9.1 Použité atributy

- **Max a min** – atributy používáme pro mezní čas (příjezdu a odjezdu)

4.1.10 Způsob platby

V tomto případě jsou ve formuláři tři možnosti způsobu platby (bankovním převodem, složenkou a online kartou). Pro zvolení jsou použity inputy typu radio. V takovém případě je nutné nastavit hodnota value, je tedy nutné použít atribut value, který obsahuje parametr, právě té platby, kterou reprezentuje, tato hodnota se následně odesílá s formulářem k dalšímu zpracování. Dále můžeme zvolit pouze jednu možnost, to by nešlo, pokud nezvolíme atribut name, který má v každých třech případech stejný parametr, to pak prohlížeči říká, že půjde vybrat pouze jedna možnost.

4.1.10.1 Použité atributy

- **Value** – hodnota která je nezbytná při použití inputů typu radio
- **Name** – Další nezbytný atribut k tomuto inputu. Zajišťuje, že půjde zvolit pouze jedna možnost.

4.1.11 Poznámky

Pro poznámky ve vzorovém formuláři je použit prvek textarea.

4.1.11.1 Použité atributy

- **Name** – pojmenování daného prvku
- **Rows** – počet řádků
- **Cols** – počet sloupců

5 Závěr

Webové formuláře jsou nedílnou součástí webových stránek už od prvních verzí HTML. Největší rozmach však nastal s příchodem a hromadným rozšířením e-shopů, které vznesli nové požadavky na formuláře. Dnes jsou formuláře nedílnou součástí snad každého vytvořeného webu. Může se jednat o malé formuláře s rychlým kontaktem, až po složité a rozsáhlé formuláře například s dotazníky.

Díky stále větším nárokům na webové formuláře došlo k rozšíření formulářů o nové prvky, které se nakonec staly nedílnou součástí nové verze jazyka HTML5.

Bakalářská práce se věnuje HTML5 a to právě pouze webovým formulářům. Teoretická část obsahuje stručný popis starší verze webových formulářů v XHTML. Stručný popis validace v XHTML a HTML5. Praktická část je pak věnovaná jednotlivým syntaxím. Jednak jsou popsány nové prvky, ale nebylo zapomenuto i na ty staré, které jsou, známe z předchozí verze HTML.

Výstupem této práce je vytvoření manuálu pro tvorbu webových formulářů pomocí nového jazyka HTML5. Součástí práce je i testování jednotlivých prvků a jejich podpora v nejpoužívanějších prohlížečích. Každý programátor má zájem na tom, aby jeho stránka byla zobrazována stejně na každém počítači, bez ohledu na to, jaký prohlížeč uživatel používá a jelikož se jedná o nové prvky, ne každý prohlížeč je dokáže správně zobrazit, popřípadě je funkčnost daného prvku omezena.

Součástí této práce je tedy webová stránka s ukázkami použití jednotlivých prvků v praxi. Tato webová stránka je uložena na přiloženém CD nebo je jí možné shlédnout na doméně www.webforms.xf.cz a díky které si můžete ověřit podporu určitého prvku i ve vašem prohlížeči.

6 Praktická aplikace

V rámci této práce jsem vypracoval webovou aplikaci, na které jsou uvedeny všechny příklady jazyka HTML5 pro tvorbu webových formulářů. Každá syntaxe je uvedena na vlastní stránce s krátkým popisem, je zde uveden zdrojový kód, následuje praktická ukázka a jako poslední bod je tabulka s podporou prohlížečů pro daný prvek.

Praktická aplikace je také dostupná online na doméně:
www.webforms.xf.cz.

Seznam obrázků

Obrázek 1: Zobrazení <i>FIELDSET</i> Google Chrome	18
Obrázek 2: Ukázka zobrazení <i>LEGEND</i> v Opeře.....	20
Obrázek 3: Ukázka zobrazení <i>LABEL</i> v Opeře.....	22
Obrázek 4: Ukázka zobrazení <i>AUTOCOMPLETE</i> v Google Chrome.....	23
Obrázek 5: Ukázka zobrazení <i>AUTOFOCUS</i> v Google Chrome	24
Obrázek 6: Ukázka zobrazení <i>FORMACTION</i> v Google Chrome	26
Obrázek 7: Ukázka zobrazení <i>FORMETYPE</i> v Opeře.....	27
Obrázek 8: Ukázka zobrazení <i>FORMEMETHOD</i> v Opeře	28
Obrázek 9: Ukázka zobrazení <i>FORMNOVALIDATE</i> v Google Chrome.....	29
Obrázek 10: Ukázka zobrazení <i>HEIGHT</i> a <i>WIDTH</i> v Google Chrome	31
Obrázek 11: Ukázka zobrazení <i>LIST</i> v Mozilla Firefox.....	32
Obrázek 12: Ukázka zobrazení <i>MIN</i> a <i>MAX</i> v Google Chrome	33
Obrázek 13: Ukázka zobrazení <i>MULTIPLE</i> v Mozila Firefox	34
Obrázek 14: Ukázka zobrazení <i>PATTERN</i> v Google Chrome	35
Obrázek 15: Ukázka zobrazení <i>PLACEHOLDER</i> v Google Chrome.....	36
Obrázek 16: Zobrazení <i>REQUIRED</i> v Mozilla Firefox	37
Obrázek 17: Ukázka zobrazení <i>STEP</i> v Google Chrome.....	38
Obrázek 18: Ukázka zobrazení <i>ACCEPT</i> v Google Chrome	40
Obrázek 19: Ukázka zobrazení <i>DISABLED</i> v Internet Explorer 10.....	41
Obrázek 20: Ukázka zobrazení <i>SIZE</i> v Internet Explorer 10.....	42
Obrázek 21: Ukázka zobrazení <i>SRC</i> v Mozilla Firefox	43
Obrázek 22: Ukázka zobrazení <i>COLOR</i> v Google Chrome.....	45

Obrázek 23: Ukázka zobrazení DATE v Opeře	47
Obrázek 24: Ukázka zobrazení DATETIME	49
Obrázek 25: Ukázka zobrazení DATETIME-LOCAL v Google Chrome	51
Obrázek 26: Ukázka zobrazení email v Google Chrome	53
Obrázek 27: Ukázka zobrazení MONTH v Google Chrome	54
Obrázek 28: Ukázka zobrazení NUMBER v Opeře	56
Obrázek 29: Ukázka zobrazení RANGE v Google Chrome	57
Obrázek 30: Ukázka zobrazení SEARCH v Opeře	59
Obrázek 31: Ukázka zobrazení TEL v Google Chrome	61
Obrázek 32: Ukázka zobrazení TIME v Opeře	62
Obrázek 33: Ukázka zobrazení URL v Mozilla Firefox	64
Obrázek 34: Ukázka zobrazení WEKK v Opeře	65
Obrázek 35: Ukázka zobrazení BUTTON v Internet Explorer 10	66
Obrázek 35: Ukázka zobrazení SELECT v Google Chrome	68
Obrázek 36: Ukázka zobrazení CHECKBOX v Mozilla Firefox	69
Obrázek 37: Ukázka zobrazení FILE v Google Chrome	71
Obrázek 38: Ukázka špatného zobrazení v Internet Explorer10	72
Obrázek 39: Ukázka zobrazení PASSWORD v Interent Explorer 10	74
Obrázek 40: Ukázka zobrazení DATALIST v Google Chrome	75
Obrázek 41: Ukázka zobrazení OPTGROUP v Opeře	77
Obrázek 42: Ukázka zobrazení OPTION v Mozilla Firefox	78
Obrázek 43: Ukázka zobrazení TEXTAREA v Internet Explorer 10	80
Obrázek 44: Ukázka zobrazení KEYGEN v Opeře	82

Obrázek 45: Ukázka zobrazení OUTPUT v Mozilla Firefox	83
Obrázek 46: Ukázka zobrazení PROGRESS v Google Chrome	85
Obrázek 47: Ukázka zobrazení METER v Mozilla Firefox.....	86

Seznam tabulek

Tabulka 1: Podpora elementu FORM a jeho atributů v prohlížečích	16
Tabulka 2: Podpora Elementu FIELDSET a jeho atributů v prohlížečích.....	19
Tabulka 3: Podpora Elementu LEGEN v prohlížečích	20
Tabulka 4: : Podpora Elementu LABEL v prohlížečích.....	22
Tabulka 5: Podpora atributu AUTOCOMPLETE v prohlížečích.....	23
Tabulka 6: Podpora atributu AUTOFOCUS v prohlížečích	24
Tabulka 7: Podpora atributu FORM v prohlížečích.....	25
Tabulka 8: Podpora atributu FORMACTION v prohlížečích	26
Tabulka 9: Podpora atributu FORMETYPE v prohlížečích	27
Tabulka 10: Podpora atributu FORMMETHOD v prohlížečích	28
Tabulka 11: Podpora atributu FORMNOVALIDATE v prohlížečích.....	29
Tabulka 12: Podpora atributu FORMTARGET v prohlížečích	30
Tabulka 13: Podpora atributu HEIGHT a WIDTH v prohlížečích	31
Tabulka 14: Podpora atributu LIST v prohlížečích	32
Tabulka 15: Podpora atributu MIN a MAX v prohlížečích	33
Tabulka 16: Podpora atributu MULTIPLE v prohlížečích.....	34
Tabulka 17: Podpora atributu PATTERN v prohlížečích	36
Tabulka 18: Podpora atributu PLACEHOLDER v prohlížečích	37
Tabulka 19: Podpora atributu REQUIRED v prohlížečích	38

Tabulka 20: Podpora atributu <i>STEP</i> v prohlížečích.....	39
Tabulka 21: Podpora atributu <i>ACCEPT</i> v prohlížečích.....	40
Tabulka 22: Podpora atributu <i>DISABLED</i> v prohlížečích.....	41
Tabulka 23: Podpora atributu <i>SIZE</i> v prohlížečích.....	42
Tabulka 24: Podpora atributu <i>SRC</i> v prohlížečích.....	43
Tabulka 25: Podpora atributu <i>VALUE</i> v prohlížečích.....	44
Tabulka 26: Podpora input type <i>COLOR</i> v prohlížečích.....	45
Tabulka 27: Podpora input type <i>DATE</i> v prohlížečích.....	47
Tabulka 28: Podpora input type <i>DATETIME</i> v prohlížečích.....	49
Tabulka 29: Podpora input type <i>DATETIME-LOCAL</i> v prohlížečích.....	51
Tabulka 30: Podpora input type <i>EMAIL</i> v prohlížečích.....	53
Tabulka 31: Podpora input type <i>MONTH</i> v prohlížečích.....	55
Tabulka 32: Podpora input type <i>NUMBER</i> v prohlížečích.....	56
Tabulka 33: Podpora input type <i>RANGE</i> v prohlížečích.....	58
Tabulka 34: Podpora input type <i>SEARCH</i> v prohlížečích.....	59
Tabulka 35: Podpora input type <i>TEL</i> v prohlížečích.....	61
Tabulka 36: Podpora input type <i>TIME</i> v prohlížečích.....	62
Tabulka 37: Podpora input type <i>URL</i> v prohlížečích.....	64
Tabulka 38: Podpora input type <i>WEEK</i> v prohlížečích.....	66
Tabulka 39: Podpora input type <i>BUTTON</i> v prohlížečích.....	67
Tabulka 39: Podpora input type <i>SELECT</i> v prohlížečích.....	68
Tabulka 40: Podpora input type <i>CHECKBOX</i> v prohlížečích.....	69
Tabulka 41: Podpora input type <i>FILE</i> v prohlížečích.....	71

Tabulka 42: Podpora input type <i>HIDDEN</i> v prohlížečích.....	72
Tabulka 43: Podpora input type <i>PASSWORD</i> v prohlížečích	74
Tabulka 44: Podpora input type <i>DATALIST</i> v prohlížečích	75
Tabulka 45: Podpora input type <i>OPTGROUP</i> v prohlížečích.....	77
Tabulka 46: Podpora input type <i>OPTION</i> v prohlížečích	79
Tabulka 47: Podpora input type <i>TEXTAREA</i> v prohlížečích.....	81
Tabulka 48: Podpora input type <i>KEYGEN</i> v prohlížečích	82
Tabulka 49: Podpora input type <i>OUTPUP</i> v prohlížečích.....	84
Tabulka 50: Podpora input type <i>PROGRESS</i> v prohlížečích	85
Tabulka 51: Podpora input type <i>METER</i> v prohlížečích	86

Seznam použité literatury a zdrojů

- [1] *HTML5 Tutorial: Web form 2.0* [online]. 2013 [cit. 2013-06-18]. Dostupné z: <http://www.html5tutorial.info/html5-webform2.php>
- [2] *W3Schools: HTML Global Attributes* [online]. 2013 [cit. 2013-06-18]. Dostupné z: http://www.w3schools.com/tags/ref_standardattributes.asp
- [3] *W3Schools: HTML Forms and Input* [online]. 2013 [cit. 2013-06-18]. Dostupné z: http://www.w3schools.com/html/html_forms.asp
- [4] *W3Schools: HTML5 New Input Types* [online]. 2013 [cit. 2013-06-18]. Dostupné z: http://www.w3schools.com/html/html5_form_input_types.asp
- [5] *W3Schools: HTML5 Form Elements* [online]. 2013 [cit. 2013-06-18]. Dostupné z: http://www.w3schools.com/html/html5_form_elements.asp
- [6] *W3Schools: HTML5 Form Attributes* [online]. 2013 [cit. 2013-06-18]. Dostupné z: http://www.w3schools.com/html/html5_form_attributes.asp
- [7] HAWRYLUK, Zoltan. CSS3 Pseudo-Classes and HTML5 Forms: Their Limitations and a Possible Solution. In: [online]. 2012, 5.11.2012 [cit. 2013-06-18]. Dostupné z: <http://www.useragentman.com/blog/2012/11/05/css3-pseudo-classes-and-html5-forms-their-limitations-and-a-possible-solution/#more-4898>
- [8] HAWRYLUK, Zoltan. HTML5 & CSS3 Support: Web forms 2.0. In: *HTML5 & CSS3 Support*[online]. 2013 [cit. 2013-06-18]. Dostupné z: <http://fmbip.com/litmus/>
- [9] VETEŠNÍK, Oldřich. Formuláře v HTML5 a nové atributy. [online]. 2013, 25.3.2013 [cit. 2013-06-18]. DOI: 1803-5620. Dostupné z: <http://www.zdrojak.cz/clanky/formulare-v-html5-a-nove-atributy/>
- [10] HTML5: Formuláře II – element input. [online]. 2010, 26.11.2010 [cit. 2013-06-18]. Dostupné z: http://www.webnt.cz/28-formulare-ii-element-input/#type_tel__9

- [11] HTML5: A vocabulary and associated APIs for HTML and XHTML. In: [online]. W3C, 2012, 17.12.2012 [cit. 2013-06-18]. Dostupné z: <http://www.w3.org/TR/html5/forms.html#forms>
- [12] *Internet trochu jinak: Atributy formulářových elementů podle HTML5* [online]. 2011, 25.3.2011 [cit. 2013-06-18]. Dostupné z: <http://swatelier.info/at/formulare/HTML5attrib.asp>
- [13] *Internet trochu jinak: Validace webového formuláře* [online]. 2007, 8.12.2007 [cit. 2013-06-18]. Dostupné z: <http://swatelier.info/at/formulare/validate.htm>
- [14] *Internet trochu jinak: Atributy formulářových elementů podle HTML5* [online]. 2011, 25.3.2011 [cit. 2013-06-18]. Dostupné z: <http://swatelier.info/at/formulare/validate.htm>
- [15] CLARK, Richard. *HTML Doctor: HTML5 forms introduction and new attributes* [online]. Media Temple, 2013, 26.10.2013 [cit. 2013-06-18]. Dostupné z: <http://html5doctor.com/html5-forms-introduction-and-new-attributes/>
- [16] DVOŘÁK, Jakub. *HTML 5: nová generace webů* [online]. Mladá fronta a. s., 2009, 9.7.2009 [cit. 2013-06-18]. Dostupné z: <http://www.zive.cz/clanky/html-5-nova-generace-webu/sc-3-a-147815/default.aspx>
- [17] KLEINERT, Jan. *Making Forms Fabulous with HTML5* [online]. 2011, 2.6.2011 [cit. 2013-06-18]. Dostupné z: <http://www.html5rocks.com/en/tutorials/forms/html5forms/>
- [18] *Český HTML 5 manuál* [online]. devbook.cz, 2013 [cit. 2013-06-18]. Dostupné z: <http://www.devbook.cz/cesky-html-5-manual>