LTAT. 05.004 Veebirakenduste loomine

XMLHttpRequest, Autentimine

Siim Karus

siim.karus@ut.ee

kevad 2018

Tänases loengus

- XMLHttpRequest
- Autentimine
Ajax isn’t a technology. It’s really several technologies, each flourishing in its own right, coming together in powerful new ways. Ajax incorporates:

- **standards-based presentation** using XHTML and CSS;
- **dynamic display and interaction** using the Document Object Model;
- **data interchange and manipulation** using XML and XSLT;
- **asynchronous data retrieval** using XMLHttpRequest;
- and **JavaScript** binding everything together.

**XMLHttpRequest DOM**

- **Methods**
  - `open(method, url, async, user, password)`
  - `setRequestHeader(header, value)`
  - `send(data)`
  - `abort()`

- **Attributes**
  - `timeout` – aegumine millisekundites
  - `withCredentials`
  - `upload`

[http://www.w3.org/TR/XMLHttpRequest2/](http://www.w3.org/TR/XMLHttpRequest2/)

Pole standard!
XMLHttpRequest DOM

- Sündmused
  - onreadystatechange
  - loadstart
  - progress
  - abort
  - error
  - timeout
  - loadend

http://www.w3.org/TR/XMLHttpRequest2/

Pole standard!
XMLHttpRequest vastuse DOM

- **Methodid**
  - `getResponseHeader(header)`
  - `getAllResponseHeaders()`

- **Omadused**
  - `status`
  - `statusText`
  - `responseType`
  - `response`
  - `responseText`
  - `responseXML`

**responseType:**

"arraybuffer", "blob", "document", "json", and "text"
XMLHttpRequest IE-s

if (typeof XMLHttpRequest == "undefined")
    XMLHttpRequest = function () {
        try { return new ActiveXObject("Msxml2.XMLHTTP.6.0"); } catch (e) {} 
        try { return new ActiveXObject("Msxml2.XMLHTTP.3.0"); } catch (e) {} 
        try { return new ActiveXObject("Microsoft.XMLHTTP"); } catch (e) {} 
    //Microsoft.XMLHTTP points to Msxml2.XMLHTTP and is redundant 
        throw new Error("This browser does not support XMLHttpRequest."); 
    };
AJAX näide

- Failid Coursees: VC_demo2.zip.
<DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>VL2012 demo2</title>
  <script type="text/javascript" src="VL_demo2.js"></script>
</head>
<body>
  <p>
    <a href="VL2012_demo.xml" title="XML" id="loadXML">Load XML</a>
    <a href="VL2012_demo.json" title="JSON" id="loadJSON">Load JSON</a>
  </p>
  <div id="courseData"></div>
</body>
</html>
AJAX näide: kuularid

```javascript
window.addEventListener("load", load_Document);
if (typeof window.opera != "undefined")
    window.opera.addEventListener("load", load_Document);

// set click handlers for links
function load_Document() {
    var loadXML = window.document.getElementById("loadXML");
    if (loadXML != null)
        loadXML.addEventListener("click", click_loadXML);
    var loadJSON = window.document.getElementById("loadJSON");
    if (loadJSON != null)
        loadJSON.addEventListener("click", click_loadJSON);
    loadXML;
}"
```
```javascript
var ROOT_URI = "http://.../";
var xsl = null;

function loadXSL() {
    var req = getRequestObject();
    req.onreadystatechange = function () {
        if (req.readyState == 4 && req.status == 200 && (req.responseType == "xml"))
            xsl = req.responseXML;
    }
    req.open("GET", ROOT_URI + "VL2012_demo.xslt", true); // async
    req.send("\n");
}
```
function getRequestObject() {
    if (window.XMLHttpRequest)
        return new window.XMLHttpRequest();
    if (typeof XMLHttpRequest == "undefined") { // old IE
        XMLHttpRequest = function () {
            try { return new ActiveXObject("Msxml2.XMLHTTP.6.0"); } catch (e) { }
            try { return new ActiveXObject("Msxml2.XMLHTTP.3.0"); } catch (e) { }
            try { return new ActiveXObject("Microsoft.XMLHTTP"); } catch (e) { }
            throw new Error("This browser does not support XMLHttpRequest.");
        };
    }
    else
        return new XMLHttpRequest();
function runXSLProcessor(xml) {
    if (xsl == null)
        throw new Error("Could not load XSLT.");
    if (typeof XSLTProcessor == "undefined") {
        if (typeof xml.transformNode != "undefined") { // old IE
            return xml.transformNode(xsl);
        }
        throw new Error("This browser does not support processing XSLT.");
    }
    else {
        xsltProcessor = new XSLTProcessor();
        xsltProcessor.importStylesheet(xsl);
        return xsltProcessor.transformToDocument(xml);
    }
}
function replaceCourseData(newdatael) {
    var cd = window.document.getElementById("courseData");
    while (cd.hasChildNodes()) {
        cd.removeChild(cd.firstChild);
    }
    cd.appendChild(newdatael);
}
AJAX näide: XML sõnum

<?xml version="1.0" encoding="utf-8"?>
<ut:course xmlns:ut="http://www.ut.ee" kood="MTAT.03.230" nimi="Veebirakenduste loomine">
  <ut:teemad>
    <ut:teema>Sissejuhatus</ut:teema>
    <ut:teema>HTML, CSS</ut:teema>
    <ut:teema>Veebiserverid, HTTP</ut:teema>
    <ut:teema>JavaScript, DOM</ut:teema>
  </ut:teemad>
  <ut:loeng päev="Neljapäev" kell="10.15 - 12.00" ruum="111"/>
</ut:course>
AJAX näide: XSLT

```xml
<?xml version="1.0" encoding="utf-8"?>
  <xsl:output method="xml" indent="yes"/>
  <xsl:template match="/">
    <xsl:apply-templates select="/*[1]"/>
  </xsl:template>
  <xsl:template match="/*[1]"/>
  ...
</xsl:template>
</xsl:stylesheet>
```

18.03.2018     MTAT.03.230 Veebirakenduste loomine
AJAX näide: XSLT (mallid)

```xml
<xsl:template match="/*[1]">
  <div id="courseinfo">
    <h2><xsl:value-of select="concat(./kood, ' ', ./nimi)"/></h2>
    <ol>
      <xsl:apply-templates select="./teemad/ut:teema"/>
    </ol>
    <p>
      <xsl:text>Loengu toimumine: </xsl:text>
      <xsl:value-of select="concat(./loeng/@päev, ' ', ./loeng/@kell)"/>
      <br/>
      <xsl:text>Ruum: </xsl:text> <xsl:value-of select="./loeng/@ruum"/> <br/>
      <xsl:text>Allikas: XML</xsl:text>
    </p>
  </div>
</xsl:template>
```

18.03.2018  MTAT.03.230 Veebirakenduste loomine  17
```javascript
function click_loadXML(evt) {
    if (xsl == null) loadXSL();
    var req = getRequestMethod();
    req.open("GET", ROOT_URI + "VL2012_demo.xml", false); // synchronous request
    req.send("" );
    if (req.responseType || req.responseType == "xml") {
        var trdoc = runXSLProcessor(req.responseXML);
        ...
        var ci = trdoc.getElementById("courseinfo");
        if (ci) {
            replaceCourseData(window.document.importNode(ci, true));
        }
    } else throw new Error("Response was not an XML document! Response " + req.responseText + " " + req.responseText);
function click_loadJSON(evt) {
  var req = getRequestMethod();
  req.open("GET", ROOT_URI + "VL2012_demo.json", false); // synchronous request
  req.send("");
  if (req.responseType || req.responseType == "json") {
    var ci = eval("(" + req.responseText + ")"); // Firefox requires()
    if (ci) {
      var cid = window.document.createElement("div");
      ...
      replaceCourseData(cid);
    }
  }
  else
    throw new Error("Response was not an JSON document! Response " + req.responseText + ");
}
AJAX näide: JSON vastus

{
  kood: "MTAT.03.230",
  nimi: "Veebirakenduste loomine",
  teemad: [
    "Sissejuhatus",
    "HTML, CSS",
    "Veebiserverid, HTTP",
    "Javascripit, DOM",
  ],
  loeng: {
    paev: "Neiljapäev",
    kell: "10.15 - 12.00",
    ruum: 111
  }
}

18.03.2018
var cid = window.document.createElement("div");
cid.setAttribute("id", "courseinfo");
var cih = window.document.createElement("h2");
cih.appendChild(window.document.createTextNode(ci.kood + " " + ci.nimi));
cid.appendChild(cih);
var cil = window.document.createElement("ol");
cid.appendChild(cil);
ci.teemad.forEach(function (it) {
    var cit = window.document.createElement("li");
cit.appendChild(cit);
cit.appendChild(window.document.createTextNode(it));
});
var cip = window.document.createElement("p");
cid.appendChild(cip);
cip.appendChild(window.document.createTextNode("Loengu toimumine: " + ci.loeng.paev + ci.loeng.kell));
Näide: andmemehud

<table>
<thead>
<tr>
<th>Tehis</th>
<th>Koodiridu</th>
<th>Baite</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTML</td>
<td>15</td>
<td>505</td>
</tr>
<tr>
<td>Javascript (üldine)</td>
<td>43</td>
<td>1471</td>
</tr>
<tr>
<td>Javascript (JSON+DOM)</td>
<td>37</td>
<td>1750</td>
</tr>
<tr>
<td>Javascript (XML+XSLT)</td>
<td>62</td>
<td>2164</td>
</tr>
<tr>
<td>XSLT</td>
<td>38</td>
<td>1143</td>
</tr>
<tr>
<td>JSON</td>
<td>15</td>
<td>316</td>
</tr>
<tr>
<td>XML</td>
<td>13</td>
<td>418</td>
</tr>
</tbody>
</table>

JSON pakituna: 243b
XML pakituna: 298b
### XML vs JSON

**XML**
- Kodeering failis
- Sõnaderohkem
- Standard veebiteenustes
- XSLT on taaskasutatav
- XSLT on selgem kui DOM javascriptiga

**JSON**
- Potentsiaalne turvarisk kui realiseeritud hooltult („Eval is evil!“)
- Kodeering väljaspool faili
- Lühem

IE keeldub JSON-t vastu võtmast kui serverist saadetud kodeering on tundmatu.

Soovitan pikemate dokumentide puhul XML-i, lihtsamate puhul sobib ka JSON.
AJAX viiteid

- http://ajax.org
- http://ajaxpatterns.org
AJAX raamistikke

- prototype - [http://prototypejs.org/core](http://prototypejs.org/core)
- XSLTForms - [http://www.agencexml.com/xsltforms](http://www.agencexml.com/xsltforms)

- Valienda jõudluspõrked:
  - [http://mootools.net/slickspeed/](http://mootools.net/slickspeed/)
Javascript teekide jõudlus

Retrieve DOM element by ID

<table>
<thead>
<tr>
<th>Library</th>
<th>Code</th>
<th>ops / sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vanilla JS</td>
<td>document.getElementById('test-table');</td>
<td>12,137,211</td>
</tr>
<tr>
<td>Dojo</td>
<td>dojo.byId('test-table');</td>
<td>5,443,343</td>
</tr>
<tr>
<td>Prototype JS</td>
<td>$('test-table')</td>
<td>2,940,734</td>
</tr>
<tr>
<td>Ext JS</td>
<td>delete Ext.elCache['test-table']; Ext.get('test-table');</td>
<td>997,562</td>
</tr>
<tr>
<td>jQuery</td>
<td>$jq('#test-table');</td>
<td>350,557</td>
</tr>
<tr>
<td>YUI</td>
<td>YAHOO.util.Dom.get('test-table');</td>
<td>326,534</td>
</tr>
<tr>
<td>MooTools</td>
<td>document.id('test-table');</td>
<td>78,802</td>
</tr>
</tbody>
</table>
### Retrieve DOM elements by tag name

<table>
<thead>
<tr>
<th>Library</th>
<th>Code</th>
<th>ops / sec</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Vanilla JS</strong></td>
<td><code>document.getElementsByTagName(&quot;span&quot;);</code></td>
<td>8,280,893</td>
</tr>
<tr>
<td><strong>Prototype JS</strong></td>
<td><code>Prototype.Selector.select('span', document);</code></td>
<td>62,872</td>
</tr>
<tr>
<td><strong>YUI</strong></td>
<td><code>YAHOO.util.Dom.getElementsByTagName(function(){return true;},'span');</code></td>
<td>48,545</td>
</tr>
<tr>
<td><strong>Ext JS</strong></td>
<td><code>Ext.query('span');</code></td>
<td>46,915</td>
</tr>
<tr>
<td><strong>jQuery</strong></td>
<td><code>$j('span');</code></td>
<td>19,449</td>
</tr>
<tr>
<td><strong>Dojo</strong></td>
<td><code>dojo.query('span');</code></td>
<td>10,335</td>
</tr>
<tr>
<td><strong>MooTools</strong></td>
<td><code>Slick.search(document, 'span', new Elements);</code></td>
<td>5,457</td>
</tr>
</tbody>
</table>
## Javascript teekide jõudlus
### (Opera/Presto)

<table>
<thead>
<tr>
<th>Linn</th>
<th>Opera 1.1.2</th>
<th>Opera 1.4.1</th>
<th>Opera 1.5.1</th>
<th>Presto 1.1.1</th>
<th>Presto 1.4.1</th>
<th>Presto 1.5.1</th>
<th>Edge 1.1.1</th>
<th>Val 3.0.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baltic Sea</td>
<td>65 ms</td>
<td>55 ms</td>
<td>45 ms</td>
<td>50 ms</td>
<td>50 ms</td>
<td>50 ms</td>
<td>50 ms</td>
<td>50 ms</td>
</tr>
<tr>
<td>Ulyan</td>
<td>1.4 sec</td>
<td>1.2 sec</td>
<td>1.0 sec</td>
<td>900 ms</td>
<td>900 ms</td>
<td>900 ms</td>
<td>900 ms</td>
<td>900 ms</td>
</tr>
<tr>
<td>Tallinn</td>
<td>900 ms</td>
<td>800 ms</td>
<td>700 ms</td>
<td>650 ms</td>
<td>650 ms</td>
<td>650 ms</td>
<td>650 ms</td>
<td>650 ms</td>
</tr>
<tr>
<td>Minsk</td>
<td>1.0 sec</td>
<td>1.0 sec</td>
<td>1.0 sec</td>
<td>900 ms</td>
<td>900 ms</td>
<td>900 ms</td>
<td>900 ms</td>
<td>900 ms</td>
</tr>
<tr>
<td>Warsaw</td>
<td>1.0 sec</td>
<td>1.0 sec</td>
<td>1.0 sec</td>
<td>900 ms</td>
<td>900 ms</td>
<td>900 ms</td>
<td>900 ms</td>
<td>900 ms</td>
</tr>
<tr>
<td>Budapest</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
</tr>
<tr>
<td>Berlin</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
</tr>
<tr>
<td>Paris</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
</tr>
<tr>
<td>Rome</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
</tr>
<tr>
<td>Madrid</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
</tr>
<tr>
<td>Lisbon</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
</tr>
<tr>
<td>Barcelona</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
</tr>
<tr>
<td>Valencia</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
</tr>
<tr>
<td>Madrid</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
</tr>
<tr>
<td>Barcelona</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
</tr>
<tr>
<td>Valencia</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
<td>1.5 sec</td>
</tr>
</tbody>
</table>

---

18.03.2018

MTAT.03.230 Veebirakenduste loomine
### Javascript teekide jõudlus (IE11)

<table>
<thead>
<tr>
<th>selectors</th>
<th>jQuery 1.3.2</th>
<th>jQuery 1.4.1</th>
<th>jQuery 1.4.2</th>
<th>Prototype 1.6.1</th>
<th>MooTools 1.2.4</th>
<th>Zepto 1.4.1</th>
<th>YUI 3.0.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>maker</td>
<td>123 ms</td>
<td>270 ms</td>
<td>29 ms</td>
<td>77 ms</td>
<td>770 ms</td>
<td>57 ms</td>
<td>1277 ms</td>
</tr>
<tr>
<td>browser</td>
<td>16 ms</td>
<td>122 ms</td>
<td>10 ms</td>
<td>18 ms</td>
<td>127 ms</td>
<td>18 ms</td>
<td>52 ms</td>
</tr>
<tr>
<td>blink</td>
<td>22 ms</td>
<td>644 ms</td>
<td>16 ms</td>
<td>35 ms</td>
<td>644 ms</td>
<td>9 ms</td>
<td>73 ms</td>
</tr>
<tr>
<td>after</td>
<td>3 ms</td>
<td>272 ms</td>
<td>0 ms</td>
<td>1 ms</td>
<td>272 ms</td>
<td>1 ms</td>
<td>3 ms</td>
</tr>
<tr>
<td>hostname</td>
<td>22 ms</td>
<td>644 ms</td>
<td>16 ms</td>
<td>76 ms</td>
<td>644 ms</td>
<td>3 ms</td>
<td>52 ms</td>
</tr>
<tr>
<td>table</td>
<td>33 ms</td>
<td>248 ms</td>
<td>13 ms</td>
<td>4 ms</td>
<td>248 ms</td>
<td>6 ms</td>
<td>48 ms</td>
</tr>
<tr>
<td>anchor</td>
<td>34 ms</td>
<td>790 ms</td>
<td>30 ms</td>
<td>28 ms</td>
<td>750 ms</td>
<td>1 ms</td>
<td>750 ms</td>
</tr>
<tr>
<td>很正常</td>
<td>228 ms</td>
<td>590 ms</td>
<td>83 ms</td>
<td>28 ms</td>
<td>590 ms</td>
<td>6 ms</td>
<td>160 ms</td>
</tr>
<tr>
<td>add_onto</td>
<td>9 ms</td>
<td>20 ms</td>
<td>8 ms</td>
<td>5 ms</td>
<td>20 ms</td>
<td>4 ms</td>
<td>10 ms</td>
</tr>
<tr>
<td>original</td>
<td>12 ms</td>
<td>531 ms</td>
<td>24 ms</td>
<td>4 ms</td>
<td>531 ms</td>
<td>6 ms</td>
<td>27 ms</td>
</tr>
<tr>
<td>removeless</td>
<td>4 ms</td>
<td>531 ms</td>
<td>3 ms</td>
<td>5 ms</td>
<td>531 ms</td>
<td>2 ms</td>
<td>21 ms</td>
</tr>
<tr>
<td>settimeout</td>
<td>744 ms</td>
<td>590 ms</td>
<td>86 ms</td>
<td>67 ms</td>
<td>590 ms</td>
<td>8 ms</td>
<td>39 ms</td>
</tr>
<tr>
<td>setTimeout</td>
<td>22 ms</td>
<td>750 ms</td>
<td>34 ms</td>
<td>68 ms</td>
<td>750 ms</td>
<td>3 ms</td>
<td>54 ms</td>
</tr>
<tr>
<td>setTimeout</td>
<td>31 ms</td>
<td>750 ms</td>
<td>32 ms</td>
<td>68 ms</td>
<td>750 ms</td>
<td>3 ms</td>
<td>54 ms</td>
</tr>
<tr>
<td>settimeout</td>
<td>48 ms</td>
<td>750 ms</td>
<td>48 ms</td>
<td>56 ms</td>
<td>750 ms</td>
<td>3 ms</td>
<td>54 ms</td>
</tr>
<tr>
<td>response</td>
<td>78 ms O Found</td>
<td>57 ms O Found</td>
<td>28 ms 0 Found</td>
<td>94 ms 0 Found</td>
<td>57 ms O Found</td>
<td>32 ms 0 Found</td>
<td>63 ms O Found</td>
</tr>
</tbody>
</table>

18.03.2018

MTAT.03.230 Veebirakenduste loomine
XSLTForms baseerub XForms standartil, mis on tänapäeva populaarseim veebivormide standard (mitmed Eesti avalikud XTee teenused kasutavad XForms).
jQuery

• Populaarne tänu ASP.NET-le (2008 asendati ASP.NET Ajax teek jQuery-ga)

18.03.2018

MTAT.03.230 Veebirakenduste loomine 31

Vaadake ka
Veebilehitsejatega ühilduvuse saamiseks on parem kasutada standardset DOM ja JavaScript!
Nt. IE < 6.0, Fx < current-1, Chrome < current-1, Safari < 5.1 ei ole jQuery poolt toetatavad. St. mitte toetatud veebilehitsejate turuosa 2011 aastal oli üle 25% maailmas ja üle 28% Euroopas (http://gs.statcounter.com/#browser_version-eu-monthly-201101-201201-bar)!
Hetkel (veebruar 2014) toetavate veebilehitsejate turuosa on:
jQuery 1: 91%
jQuery 2: 63%

Heidenbug – viga, mis ei esine programmi analüüsides või siludes
Koostööbug – viga, mis on tingitud eri lõimede töö ajastusest
jQuery

- Loendid DOM vs jQuery
- jQuery practices
- Javascript vs jQuery
NB! AJAX päringud võivad vastust mitte saada! St. veebileht võiks teatud aja tagant kontrollida, kas päring on vastuse saanud ja vastavalt kasutajat informeerida probleemist (ka XMLHttpRequest.abort() ei tohiks ära unustada). AJAX päringu ebaõnnestumine ei tähenda, et võrguühendus puudub või server ei tööta – probleem võib olla ka veebilehitsejas, tulemüüris või antiviirustarkvaras!
Dubnaamilise veebi disainipõhimõtted

- ActiveX, Java, Flash, Javascript ei pruugi olla veebibruseris lubatud või toetatud
- Võrguühendus võib kahata
- Kõik ei ole vaja teha Javascriptiga (või HTML5-ga):
  - [http://www.slideshare.net/nzakas/enough-with-the-javascript-already](http://www.slideshare.net/nzakas/enough-with-the-javascript-already)

NB! AJAX päringud võivad vastust mitte saada! St. veebileht võiks teatud aja tagant kontrollida, kas päring on vastuse saanud ja vastavalt kasutajat informeerida probleemist (ka XMLHttpRequest.abort() ei tohiks ära unustada). AJAX päringu ebaõnnestumine ei tähenda, et võrguühendus puudub või server ei tööta – probleem võib olla ka veebilehitsejas, tulemüüris või anti-viirustarkvaras!
PROJEKTIST

18.03.2018 LTAT.05.004 Veebirakenduste loomine
<script>
(function () {
    var scr = document.createElement('script');
    scr.type = 'text/javascript'; scr.async = true;
    scr.src = '/Scripts/spring.js';
    var s = document.getElementsByTagName('script')[0];
    s.parentNode.insertBefore(scr, s);
})();
</script>
</body>
</html>
<script src="/bundles/bootstrap?v=3F23B0iizV2NnnamQFrx-NbYJNTFeB]2GM05SilbtQU1"></script>
<script src="/bundles/dashboard?v=jRceIpCt88j-40NgXOD2165k_3Q5AVPGiWSs_aVkl"></script>
</body>
</html>
window.addEventListener("load", onload_Document);
if (typeof window.opera !== "undefined")
    window.opera.addEventListener("load", onload_Document);

function onload_Document(args) {
    try {
        lisaViideSkriptite("http://connect.facebook.net/et_EE/all.js#xfbml=1");
    } catch(e) { alert(e); }
}

function lisaScriptEl(elScript) {
    var elScript = document.createElement("script");
elScript.setAttribute("type", "text/javascript");
elScript.setAttribute("src", skriptiURL);
elScript.appendChild(document.createTextNode(" "));

    headel = document.getElementsByTagName("head").item(0);
elScript = headel.getElementsByTagName("script");
if (elsScript == null || elsScript.item(elsScript.length - 1).nextSibling == null) {
    // lisame skripti viimase päisekirjena

<script type="text/javascript"
src="laiendused.js"></script>
</head>
CDN

- Content Delivery Network
  - Kõrgjõudlusserverid Maailma eri paigus serveerimas sama sisu

- Sama server pakub ressurssi mitmele lehele
  - Oht blokeeritud saada (petulehed, järgimiskaitse)
CDN

- **Blokeerimise vältimine**
  - Kui CDN-st faili ei saa, lisada viide kohalikus serveris olevale koopiale
  - Vihjeid:
    - Uue skripti või stiili laadimiseks saab lisada vastava elemendi HTML päisesse
    - Skripte laetakse ja käivitatakse päises nende esinemise järjekorras (v.a hiljem lisatud viited)
    - Funktsionaalsuse kontrollimiseks on Javascriptis vähemalt 4 eri viisi

18.03.2018

LTAT.05.004 Veebirakenduste loomine
Uudisteportaalide võrdlus

PUNKTIÜLESANDE KOKKUVÖTE

18.03.2018

MTAT.03.230 Veebirakenduste loomine
Ekraani tüüpide puhul on kaks kriteeriumit: kaugel-lähedal ja detailne-mittedetailne.

Kaugel-lähedal kriteerium ütleb, kas kasutaja on ekraanile lähedal või kaugel. Mida kaugemal on inimene ekraanist, seda suurema on veebilehitseja baaskirjastili suurus (font size). Teleritel on see vaikimisi suuruseks seatud, meediakeskus-arvuti korral tuleb seda ise veebilehitseja seadetest muuta.

Detailne-mittedetailne kriteerium ütleb, kui palju piksleid on ekraanil ekraani kauguse suhtes. Detailne ekraan on näiteks HD teler (1920x1080 px) või UXGA ekraan (1600x1200 px) või veelgi punktiderohkem ekraan. Ka mobiiltelefoni 840x480 px on körge detailsus, sest selle ekraani kasutame veelgi lähemal kui arvutiekraani. Madal detailsus on näiteks SD teler (786x576 px) või vanemad projektorid või arvuti ekraanid (1024x768 px). Üldine reegel on, et 50 cm kaugusel oleva ekraani punktide vahe ei tohi ületada 0,2 mm (seos kaugusega on lineaarne st. 3m puhul võib vahe olla juba 1,2mm) - kõik seadmed, mis sellele kriteeriumile ei vasta, loetakse madala lahutusvõimsuse ehk detailsusega seadmeteks.
Ekraani tüüpide puhul on kaks kriteeriumit: kaugel-lähedal ja detailne-mittedetailne.

Kaugel-lähedal kriteerium ütleb, kas kasutaja on ekraanile lähedal või kaugel. Mida kaugemal on inimene ekraanist, seda suurema on veebilehitseja baaskirjastili suurus (font size). Teleritel on see vaikimisi suuruseks seatud, meediakeskus-arvuti korral tuleb seda ise veebilehitseja seadetest muuta.

Detailne-mittedetailne kriteerium ütleb, kui palju piksleid on ekraanil ekraani kauguse suhtes. Detailne ekraan on näiteks HD teler (1920x1080 px) või UXGA ekraan (1600x1200 px) või veelgi punktiderohkem ekraan. Madal detailsus on näiteks SD teler (786×576 px) või vanemad projektorid või arvuti ekraanid (1024×768 px). Üldine reegel on, et 50 cm kaugusel oleva ekraani punktide vahe ei tohi ületada 0,2 mm (seos kaugusega on lineaarne st. 3m puhul vöib vahe olla juba 1,2mm) - kõik seadmed, mis sellele kriteeriumile ei vasta, loetakse madala lahutusvõimsuse ehk detailsusega seadmeteks.
Ekraani tüüpide puhul on kaks kriteeriumit: kaugel-lähedal ja detailne-mittedetailne.

Kaugel-lähedal kriteerium ütleb, kas kasutaja on ekraanile lähedal või kaugel. Mida kaugemal on inimene ekraanist, seda suurema on veebilehitseja baaskirjastilis suurus (font size). Teleritel on see vaikimisi suuruseks seatud, meediakeskus-arvuti korral tuleb seda ise veebilehitseja seadetest muuta.

Detailne-mittedetailne kriteerium ütleb, kui palju piksleid on ekraanil ekraani kauguse suhtes. Detailne ekraan on näiteks HD teler (1920x1080 px) või UXGA ekraan (1600x1200 px) või veelgi punktiderohkem ekraan. Ka mobiiltelefoni 840x480 px on kõrge detailssus, sest selle ekraanil kasutame veelgi lähemal kui arvutiekskaani. Madal detailssus on näiteks SD teler (786x576 px) või vanemad projektorid või arvuti ekraanid (1024x768 px). Üldine reegel on, et 50 cm kaugusel oleva ekraani punktide vahe ei tohi ületada 0,2 mm (seos kaugusega on lineaarne st. 3m puhul võib vahe olla juba 1,2mm) - kõik seadmed, mis sellele kriteeriumile ei vasta, loetakse madala lahutusvõimsuse ehk detailsusega seadmeteks.
Ekraani tüüpide puhul on kaks kriteeriumit: kaugel-lähedal ja detailne-mittedetailne.

Kaugel-lähedal kriteerium ütleb, kas kasutaja on ekraanile lähedal või kaugel. Mida kaugemal on inimene ekraanist, seda suurema on veebilehitseja baaskirjastilis suurus (font size). Teleritel on see vaikimisi suuruseks seatud, meediakeskus-arvuti korral tuleb seda ise veebilehitseja seadetest muuta.

Detailne-mittedetailne kriteerium ütleb, kui palju piksleid on ekraanil ekraani kauguse suhtes. Detailne ekraan on näiteks HD teler (1920x1080 px) või UXGA ekraan (1600x1200 px) või veelgi punktiderohkem ekraan. Ka mobiiltelefoni 840x480 px on kõrge detailsus, sest selle ekraani kasutame veelgi lähemal kui arvutiekraani. Madal detailsus on näiteks SD teler (786×576 px) või vanemad projektorid või arvuti ekraanid (1024×768 px). Üldine reegel on, et 50 cm kaugusel oleva ekraani punktide vahe ei tohi ületada 0,2 mm (seos kaugusega on lineaarne st. 3m puhul võib vahe olla juba 1,2 mm) - kõik seadmed, mis sellele kriteeriumile ei vasta, loetakse madala lahusvõimsuse ehk detailsusega seadmeteks.

Kvaliteet 2015

<table>
<thead>
<tr>
<th>Koht</th>
<th>HTML vead</th>
<th>HTML hoiat.</th>
<th>CSS vead</th>
<th>CSS hoiat.</th>
<th>Märgendkeel</th>
</tr>
</thead>
<tbody>
<tr>
<td>uudised.err.ee</td>
<td>HTML5 (21)</td>
<td>HTML5 (2)</td>
<td>223</td>
<td>215</td>
<td>HTML5</td>
</tr>
<tr>
<td>postimees.ee</td>
<td>HTML5 (98)</td>
<td>HTML5 (228)</td>
<td>73</td>
<td>197</td>
<td>HTML5</td>
</tr>
<tr>
<td>delfi.ee</td>
<td></td>
<td>241</td>
<td>39</td>
<td>291</td>
<td>94 HTML 4.01 Transitional</td>
</tr>
<tr>
<td>ohtuleht.ee</td>
<td>HTML5 (344)</td>
<td>HTML5 (54)</td>
<td>202</td>
<td>203</td>
<td>HTML5</td>
</tr>
<tr>
<td>epl.ee</td>
<td></td>
<td>85</td>
<td>32</td>
<td>184</td>
<td>39 HTML 4.01 Transitional</td>
</tr>
<tr>
<td>bbc.co.uk</td>
<td>HTML5 (0)</td>
<td>HTML5 (1)</td>
<td>8</td>
<td></td>
<td>28 HTML5</td>
</tr>
</tbody>
</table>

Maht (HTML) | Maht (el, initia Elemente (initi) | Maht (el, full p Elemente (full) | Taotumine | Maht (el, full p Elemente (full) |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>uudised.err.ee</td>
<td>212,75 kB</td>
<td>9,47 MB</td>
<td>262</td>
<td>9,47 MB</td>
</tr>
<tr>
<td>postimees.ee</td>
<td>43,24 kB</td>
<td>2,64 MB</td>
<td>148</td>
<td>2,64 MB</td>
</tr>
<tr>
<td>delfi.ee</td>
<td>36,56 kB</td>
<td>1,52 MB</td>
<td>219</td>
<td>1,93 MB</td>
</tr>
<tr>
<td>ohtuleht.ee</td>
<td>127,59 kB</td>
<td>7,39 MB</td>
<td>591</td>
<td>7,39 MB</td>
</tr>
<tr>
<td>epl.ee</td>
<td>14,22 kB</td>
<td>0,99 MB</td>
<td>130</td>
<td>1,16 MB</td>
</tr>
<tr>
<td>bbc.co.uk</td>
<td>118,08 kB</td>
<td>0,92 MB</td>
<td>98</td>
<td>0,92 MB</td>
</tr>
</tbody>
</table>

18.03.2018 MTAT.03.230 Veebirakenduste loomine 46
Ekraani tüüpide puhul on kaks kriteeriumit: kaugel-lähedal ja detailnedetailne.

Kaugel-lähedal kriteerium ütleb, kas kasutaja on ekraanile lähedal või kaugel. Mida kaugemal on inimene ekraanist, seda suurema on veebilehitseja baaskirjastili suurus (font size). Teleritel on see vaikimisi suuruseks seatud, meediakeskus-arvuti korral tuleb seda ise veebilehitseja seadetest muuta.

Detailne-mittedetailne kriteerium ütleb, kui palju piksleid on ekraanil ekraani kauguse suhtes. Detailne ekraan on näiteks HD teler (1920x1080 px) või UXGA ekraan (1600x1200 px) või veelgi punktiderohkem ekraan. Ka mobiiltelefoni 840x480 px on kõrge detailsus, sest selle ekraani kasutame veelgi lähemal kui arvutiekaani. Madal detailsus on näiteks SD teler (786×576 px) või vanemad projektorid või arvuti ekraanid (1024×768 px). Üldine reegel on, et 50 cm kaugusel oleva ekraani punktide vahe ei tohi ületada 0,2 mm (seos kaugusega on lineaarne st. 3m puhul võib vahe olla juba 1,2mm) - kõik seadmed, mis sellele kriteeriumile ei vasta, loetakse madala lahetusvõimsuse ehk detailsusega seadmeteks.
Ekraani tüüpide puhul on kaks kriteeriumit: kaugel-lähedal ja detailne-mittedetailne.

Kaugel-lähedal kriteerium ütleb, kas kasutaja on ekraanile lähedal või kaugel. Mida kaugemal on inimene ekraanist, seda suurema on veebilehitseja baaskirjastilis suurus (font size). Teleritel on see vaikimisi suuruseks seatud, meediakeskus-arvuti korral tuleb seda ise veebilehitseja seadetest muuta.

Detailne-mittedetailne kriteerium ütleb, kui palju piksleid on ekraanil ekraani kauguse suhtes. Detailne ekraan on näiteks HD teler (1920x1080 px) või UXGA ekraan (1600x1200 px) või veelgi punktiderohkem ekraan. Ka mobiiltelefoni 840x480 px on kõrge detailsus, sest selle ekraani kasutame veelgi lähemal kui arvutieks. Madal detailsus on näiteks SD teler (786×576 px) või vanemad projektorid või arvuti ekraanid (1024×768 px). Üldine reegel on, et 50 cm kaugusel oleva ekraani punktide vahe ei tohi ületada 0,2 mm (seos kaugusega on lineaarne st. 3m puhul võib vahe olla juba 1,2mm) - kõik seadmed, mis sellele kriteeriumile ei vasta, loetakse madala lahusvõimsuse ehk detailsusega seadmeteks.
4K TV ERR
Postimees - popularsus

How popular is postimees.ee?

Alexa Traffic Ranks
How is this site ranked relative to other sites?

Global Rank

5,996

↑ 236

Rank in Estonia

= 6

Bounce Rate

37.70%  ▷ 2.00%
Delfi - populaarsus

How popular is delfi.ee?

Alexa Traffic Ranks
How is this site ranked relative to other sites?

Global Rank
6,616 ▲ 399

Rank in Estonia
7

Bounce Rate
42.70% ▲ 2.00%
How popular is err.ee?

Alexa Traffic Ranks
How is this site ranked relative to other sites?

Global Rank
25,957

Rank in Estonia
25

Bounce Rate
49.60%
Populaarsuse põhjused

- Ööpäevaringselt tegutsev
- Uudiste ja teemade paljusus
- Kollasus
- Võimaluste rohkus
- Seotus muu meediaga (paberleht, raadioringhääling)
- Pildid, video
- Mahukas esileht
Veel värvidest:
  • Kollane – äri ja edu värv
  • Sinine – rahulik „kaine mõistuse“ värv
  • Punane – temperamentne, „kuum“ värv

• Kõrgem haridustase
  – Sinised (rahulikud) toonid
  – Vähem kõrvalisi võimalusi (meelelahutus, jms.)
  – Lihtne ja kiire
• Madalam haridustase
  – Palju võimalusi
  – Kirju, palju pilte
Single-Sign-On (SSO)

- Üks konto kehtib mitmes võrgus/teenuses
- Valik SSO pakkujaid:  
- Protokolle:  
  - SAML
  - OpenID
Federated Identity

- Mitme võrgu/teenuse kontod on omavahel seotud
- Levinud protokollid
  - OAuth - http://oauth.net
  - OpenID - http://openid.net/
  - SAML - http://saml.xml.org/
- Haldab OpenID Foundation
- Pakkujaid:
  - AOL
  - BBC
  - Google
  - IBM
  - MySpace
  - Orange
  - PayPal
  - VeriSign
  - LiveJournal
  - Steam
  - Yahoo!
  - http://openid.net/get-an-openid/
OpenID Authentication

Who are YOU? Send me a notarized referral letter.

Here is the certificate
name: Real Name
email: user@gmail
notary Google

Please write a referral stating that I’m user@gmail

Google - The Identity Provider
name: Real Name
email: user@gmail
notary Google

Here you go

http://www.youtube.com/watch?v=xcmY8Pk-qEk
- Twitter algatus
- Pakkujad:
  - Facebook
  - Foursquare
  - Yahoo!
  - Google
  - GitHub
  - Microsoft
  - LinkedIn
  - MySpace
  - Netflix
  - StatusNet
  - Twitter
  - Vimeo
  - VK
OAuth

Pseudo-Authentication using OAuth

Give me the valet key* to your house (account) so that I know you are the owner of the house.

Here is the valet key*

Please issue me a valet key* for the core APIs.

Here you go.

*valet key = limited scope OAuth Token

adapted from a drawing by ©_nat_en
EstEID

- http://id.ee/?id=31045
- http://id.ee/?id=10824
- http://id.ee/?id=30303
Taaskord SSO

- Paroolipõhine
- Token based (ID kaart)
- Biomeetriline
Autentimise tulevikuvisioone

http://www.youtube.com/watch?v=XC9GfkjK60
MTAT.03.230 Veebirakenduste loomine

18.03.2018
Järgmises loengus

- Disain
- XML Processing
Slaidide kasutamine

Juhul, kui te soovite neid slaide või osa nendest slaididest taaskasutada oma ettekannetes või esinemistes, palun nendes ettekannetes või esinemistes viidata sellele slaidikomplektile ja veebirakenduste loomise aine lehele.