



2. Web technológiák

Dr. Bilicki Vilmos

Szegedi Tudományegyetem
Informatikai Tanszékcsoport
Szoftverfejlesztés Tanszék





Az előadás áttekintése

► Trendek:

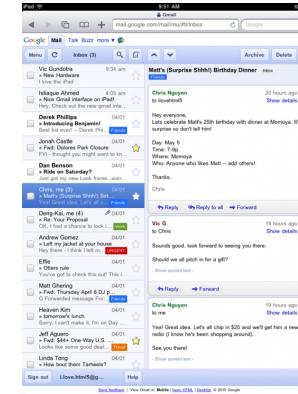
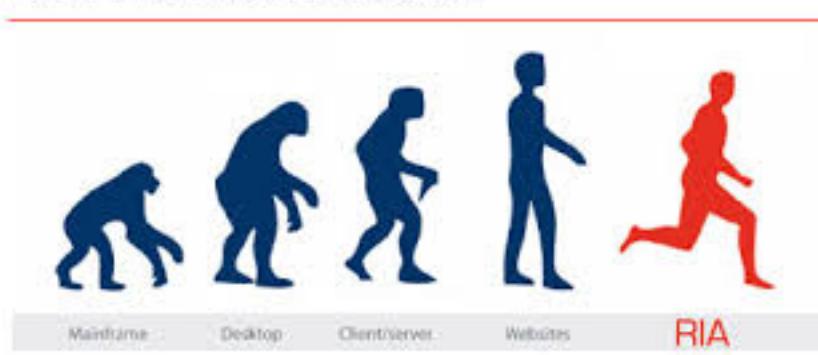
- Gazdag Internet Alkalmazás (Rich Internet Application)
- Reszponzív Web alkalmazás (Responsive Application)
- Hibrid Mobil Alkalmazás (Hybrid Mobile Application)
- Progresszív Mobil Alkalmazás (Progressive Web Application)

► Technológiák

- HTML 5
- CSS 3
- BOOSTRAP

Rich Internet Application

User Interface Evolution

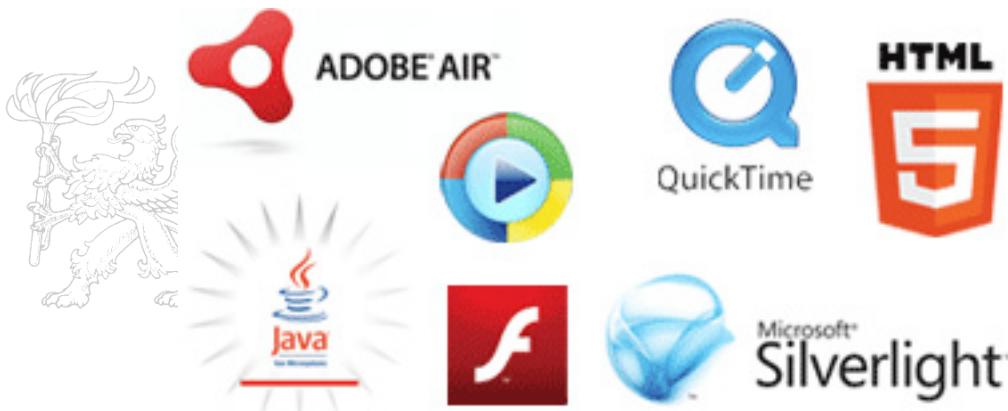
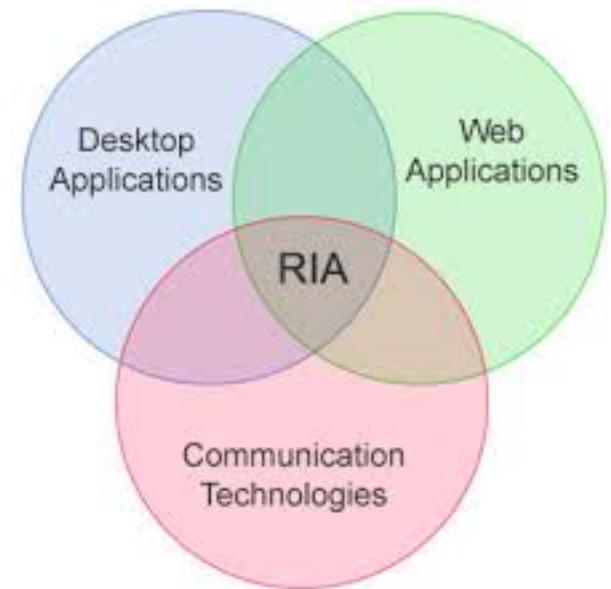


- ▶ Vékony kliens helyett vastag web kliens
- ▶ Web alkalmazás asztali alkalmazás érzéssel/képeséggel (TCO csökkentés)
 - Egyszerű olcsó telepítés karbantartás
 - Internet skála
- ▶ 2011-ig beépülő modulokkal



Rich Internet Application

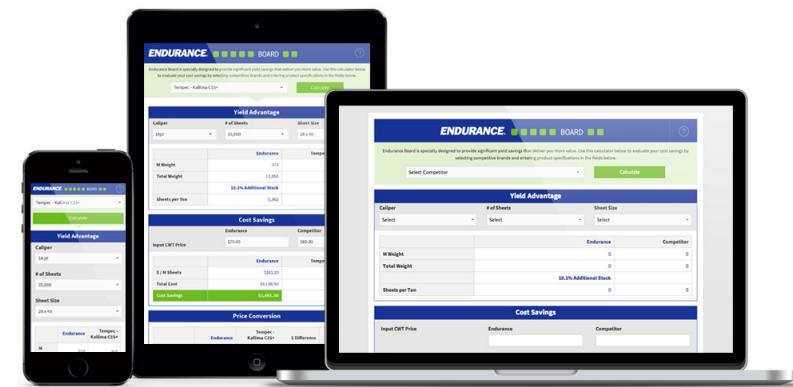
- ▶ Jellemzően böngésző beépülő modul alapú megoldás
- ▶ Asztali alkalmazás mint cél





Responsive Web Application

- ▶ Web alapú de mobilon futó alkalmazás mint cél
- ▶ A cél: mobil böngészőben is használható alkalmazás
- ▶ Böngésző alapú megoldás (JavaScript+HTML+CSS)
- ▶ 2001: audi.com (dinamikusan mértezhető weboldal)
- ▶ 2010-ben jelent meg mint jelentős igény





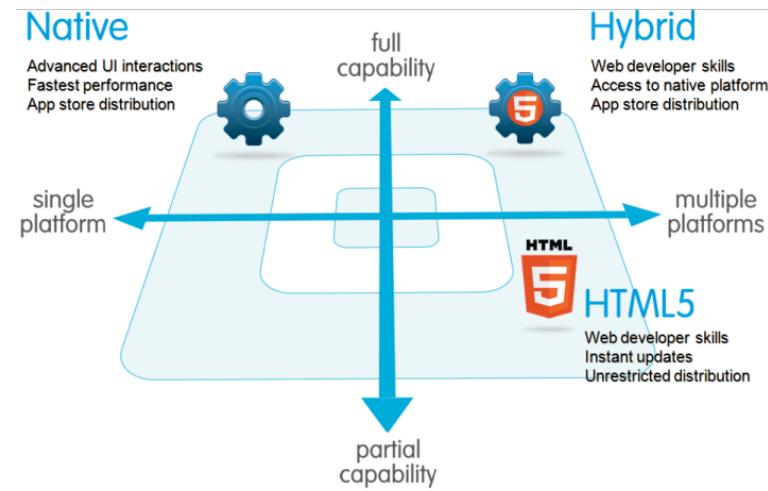
Responsive Web Application

- ▶ A weboldal folyékony rács segítségével alkalmazkodik a felülethez (fluid grid)
 - Relativ pozícionálás
 - Lépték osztályok (sx, md, lg,)
- ▶ Flexibilis képek (mérthez alkalmazkodnak)
- ▶ Média lekérdezések
 - Más stílust használ más-más környezetben
- ▶ Technológia:
 - HTML5
 - AJAX
 - CSS (SASS, ...)



Hybrid mobile application

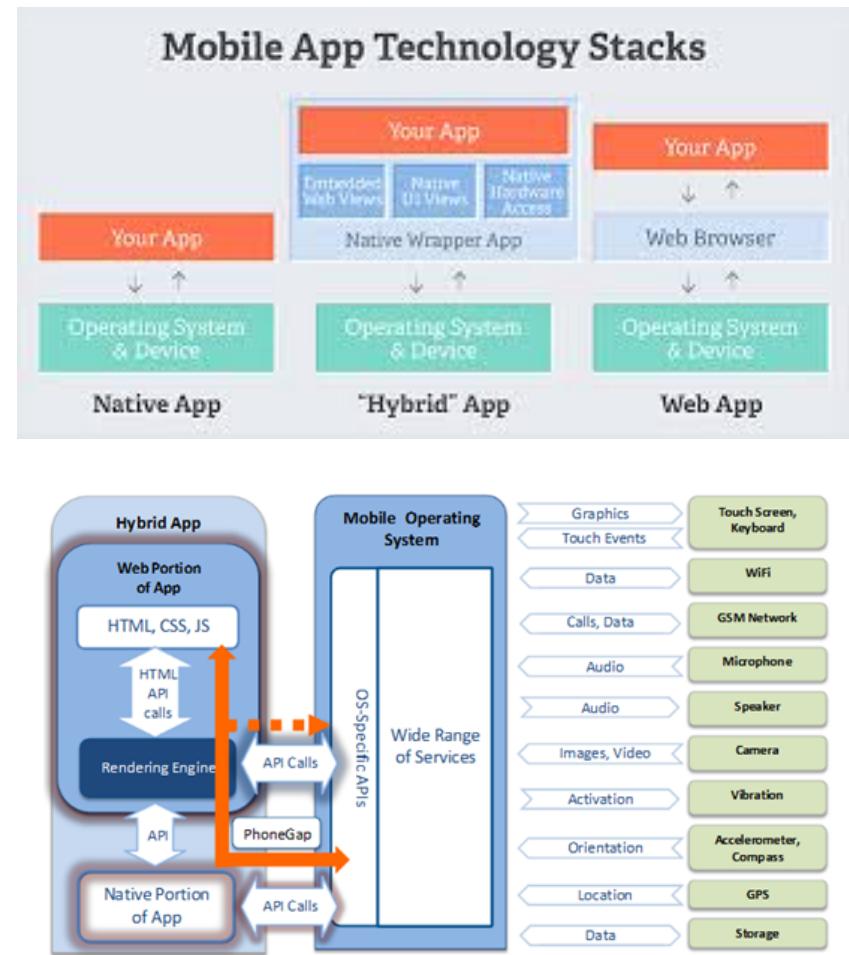
- ▶ Mobil alkalmazás képességek elérése mint cél
- ▶ Platform független
- ▶ Web szoftver verem
 - HTML5
 - AJAX
 - CSS
- ▶ Futtató környezet
 - Cordova





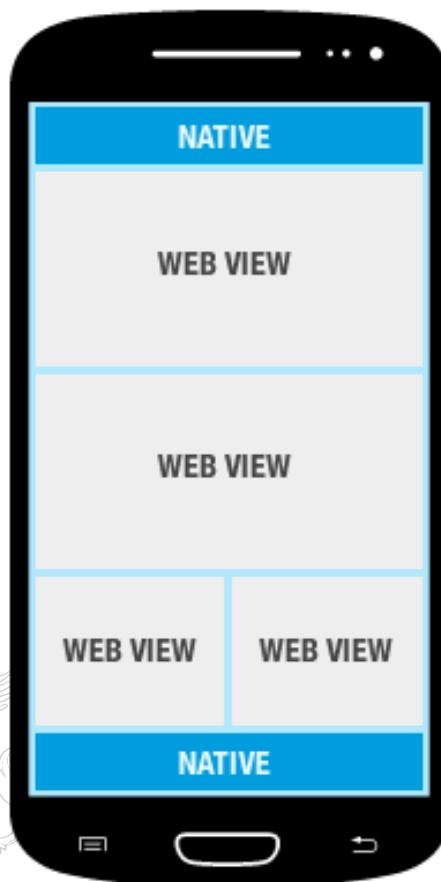
Hybrid mobile application

- ▶ Egyedi elemek
 - GUI
 - Natív mobil
- ▶ Teljes fordító lánc (build chain)
- ▶ WebView mint futtató környezet
- ▶ Finom átmenet a mobil és a natív alkalmazás között
- ▶ Periféria támogatás
 - Okos óra, ...





Hibrid mobil alkalmazások



NATIVE CONTAINERS



iOS - xCode/Objective-C
Android - Eclipse/Java
Windows - Visual Studio/C#

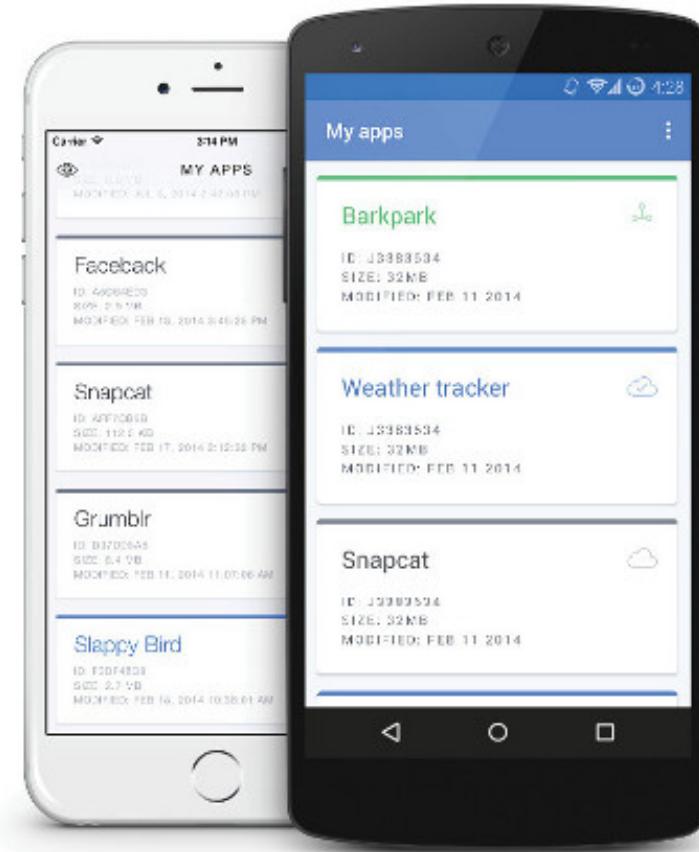
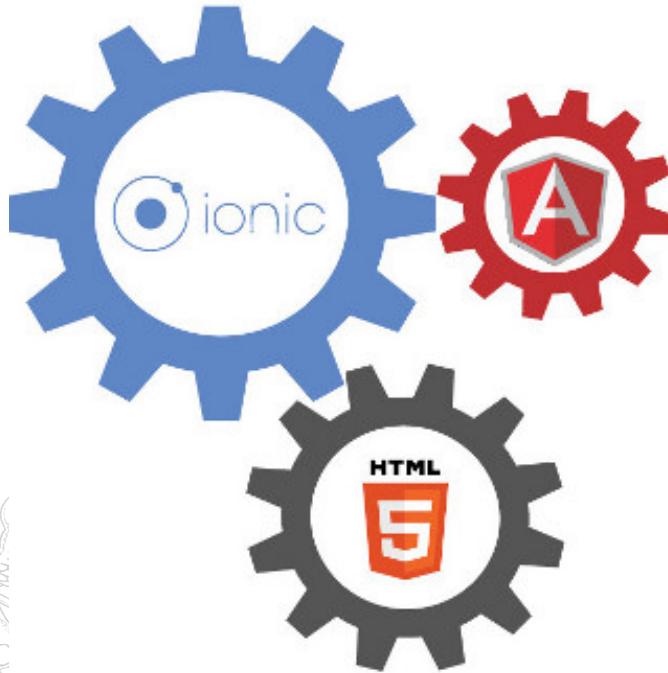
MOBILE WEB

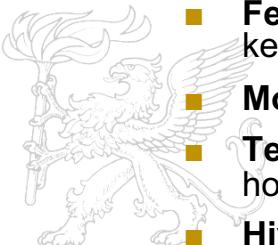


HTML
CSS
JavaScript



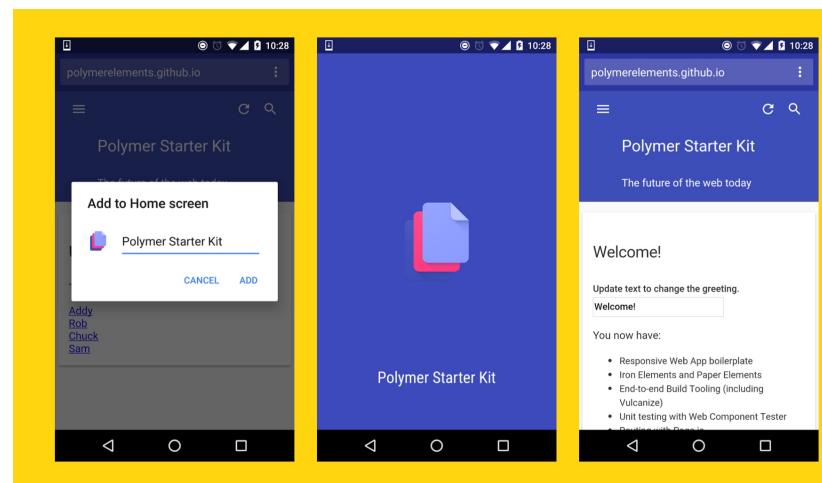
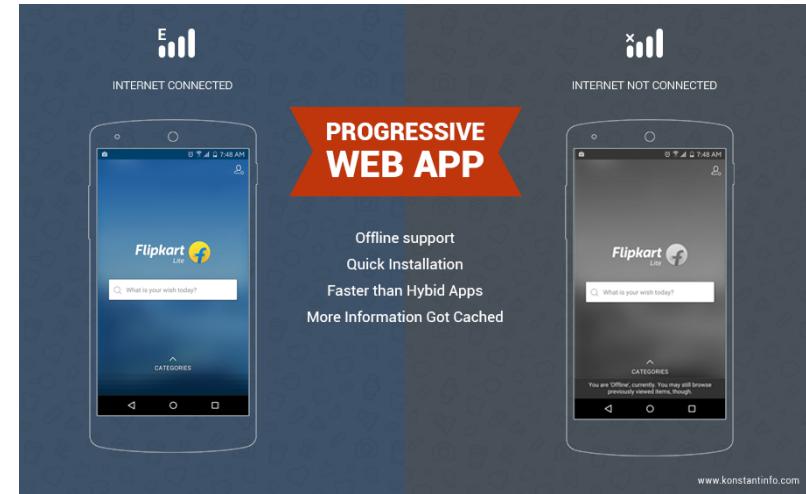
Hibrid mobil alkalmazások

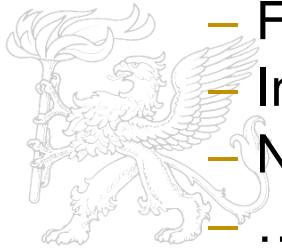




Progressive Web Application

- ▶ Web képességekkel is bíró mobil alkalmazások
- ▶ Képességek:
 - **Progresszív** – Böngésző függetlenül működnek
 - **Reszponzív** - Felbontás függő működésmód
 - **Kapcsolat független** – A háttér folyamatok segítségével kezeli a lassú vagy adott időszakban hiányzó hálózati kapcsolatokat
 - **App szerű** – Ugyanúgy néz ki és működik mint a natív alkalmazás
 - **Friss** – Mindig naprakész (háttér frissítés szálak)
 - **Biztonságos** – HTTPS felett kommunikál
 - **Felderíthető** – W3C leírók segítségével a kereső motrok megismerhetik
 - **Motiváló** – Elérjük a felhasználót, pl.: push
 - **Telepíthető** – A felhasználó kihelyezheti a home képernyőre
 - **Hivatkozható** – URL segítségével megosztható, nem kell komplex telepítés

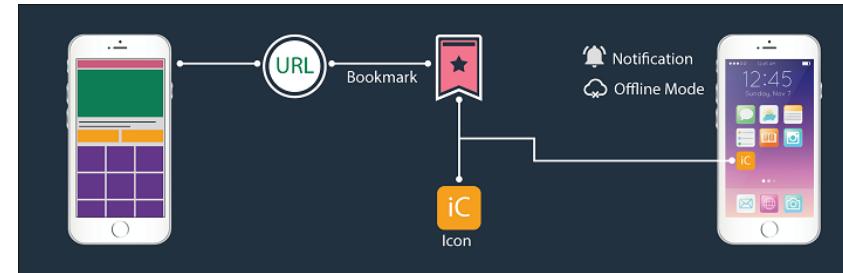
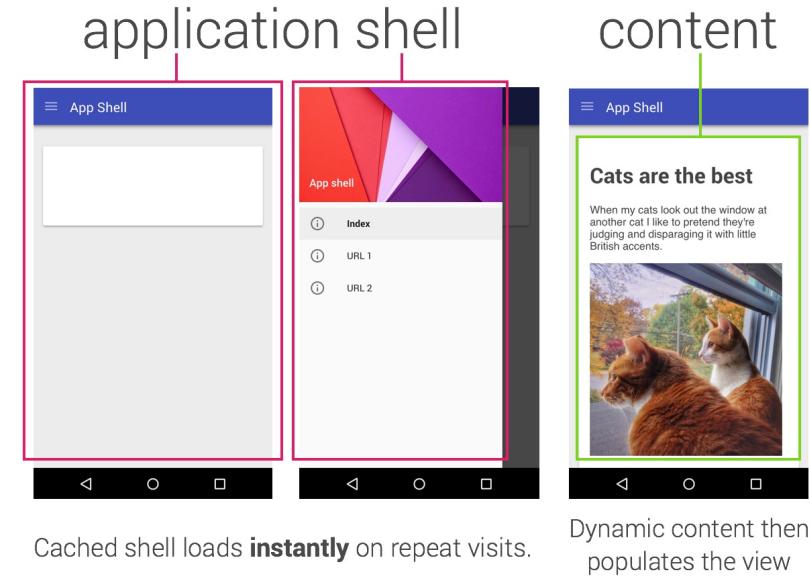




Progressive Web Application

Technológia:

- HTML5
- AJAX
- CSS
- Web leíró (W3C manifest)
- Háttér folyamatok (worker)
- App Shell
 - Cache API
 - Fetch API
 - IndexexDB API
 - Notification API
 - ...



Web technológia verem

- ▶ HTML5 ≈ HTML 5 + CSS 3 + JavaScript
- ▶ AJAX
- ▶ HTML5
- ▶ CSS3
- ▶ BOOTSRAP



HTML5 ≈ HTML 5 + CSS 3 + JavaScript

- ▶ HTML5 egy eszköz gyűjtemény:
 - Jelölő nyelv (HTML 5)
 - Megjelenítés (CSS 3)
 - Interakció (DOM, Ajax, APIs)
- ▶ A web evolúciójának következménye



Rövid historikus áttekintés

91-92	93-94	95-96	97-98	99-00	01-02	03-04	05-06	07-08	09-10	11-12	13-14
HTML 1	HTML 2		HTML 4	XHTML 1					HTML 5		
		CSS 1	CSS 2		T-less D	Web 2.0			CSS3		
		JS	ECMA, DOM	DOM 2			Ajax		DOM, APIs		

HTML 5

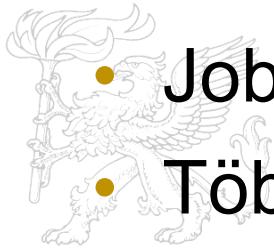
2004	WHATWG started
2008	W3C Working Draft
2012 (2010)	W3C Candidate Rec
2022	W3C Rec

CSS

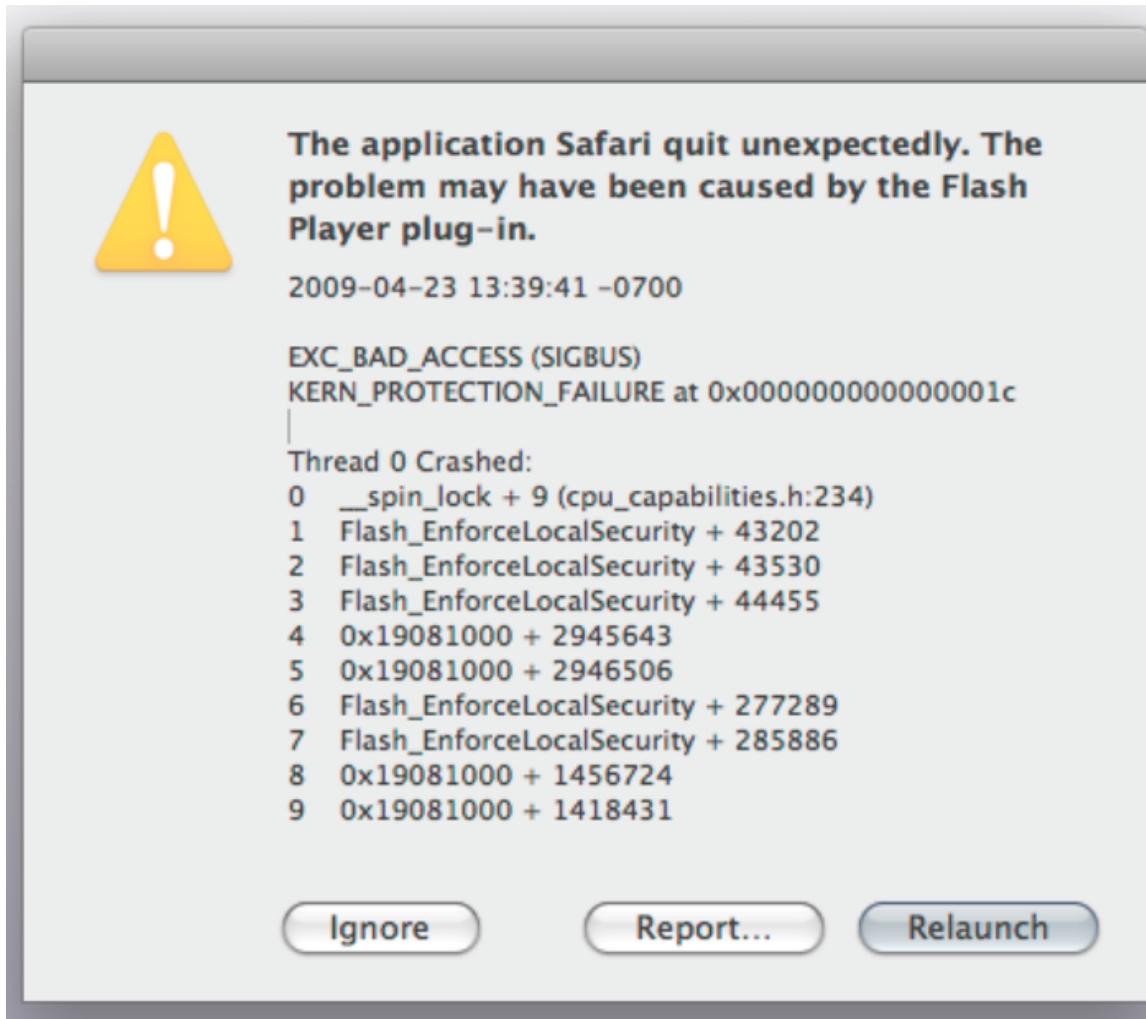
1996 – CSS 1	W3C Rec
1998 – CSS 2	W3C Rec
1999 – CSS 3	Proposed
2005 – CSS 2.1	W3C Candidate Rec
2001 – CSS 3	W3C Working Draft

■ HTML5 Célok

- A natív alkalmazások képességének megközelítése
- Bplatform független (Windows, Linux, iPhone, Android, ...)
- A bővítése HTML, CSS és JavaScript alapokon történjen
- A szükséges böngésző beépülő modulok számának csökkentése
 - Jobb hibakezelés
 - Több jelölő elem a szkript megoldások kiváltására



Ilyen gondok megelőzése





HTML5 területek

- ▶ HTML
- ▶ Úrlapok
- ▶ Kiegészítő szabványok
 - Offline alkalmazások
 - Helyi tárterület
 - CSS

Képeségek

- ▶ **Valósidejű űrlap validáció**
 - A kliens odalon validálhatjuk az adatot (struktúra alapján)
- ▶ **Automatikus kiegészítés**
 - Az adott beviteli mező a gépelés alapján automatikusan kiegészítheti a szöveget
- ▶ **Részinformáció kérés**
 - Az oldal letöltése nélkül adott részek frissíthetőek
- ▶ **Gazdag felhasználói elemkészlet**
 - Controls such as tree controls, menus, and progress bars may be provided that do not require page refreshes.
- ▶ **Szerver által kezdeményezett adat frissítés**
 - Web push segítségével a szerver jelezheti, ha valami frissült





HTML bővítmények

- ▶ Dokumentum struktúra: div, section, article, nav, aside, header, footer
- ▶ Audio, Video és Beágyazás
- ▶ Rajzlap (Canvas): utak, gradiansek, kép manipuláció, események
- ▶ A kereső motorokat kiszolgáló mikro adatok

■ HTML5 minta

```
<!doctype html>

<html lang="en">

  <head>
    <title>Web page title</title>
    <meta charset="utf-8" >
  </head>

  <body>
    Web page content
  </body>

</html>
```



Új HTML 5 elemek



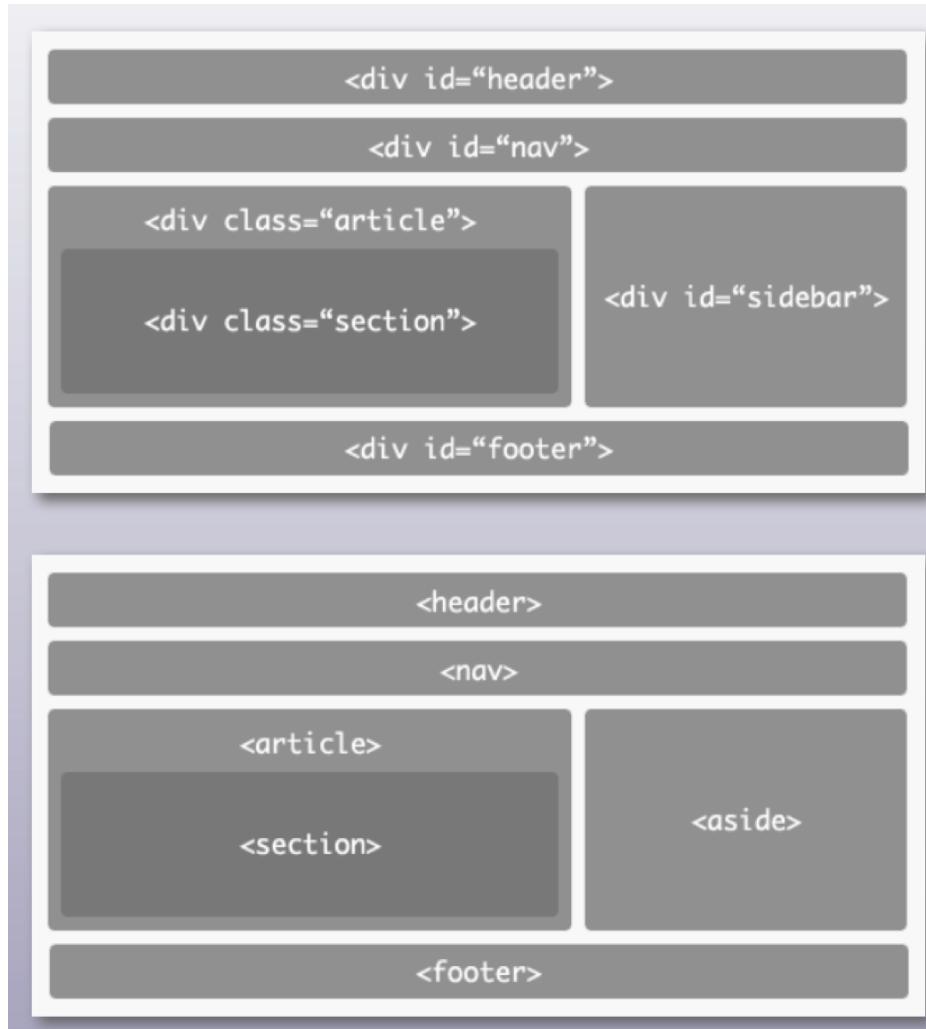
New Element	Description
<article>	Represents a self-contained structure in document. For example, a blog post or a forum
<aside>	Represents a piece of information that belongs to a content
<audio>	Container used to add sound
<canvas>	Used to represent graphic elements
<command>	Defines a command
<detailist>	Container used to create a list
<details>	Reveals information of an element
<embed>	Displays embedded content
<figcaption>	Used to add a caption to a <figure> tag
<figure>	Container used to group media files
<footer>	This container wraps what formerly was inside <div id="footer">
<header>	This container wraps what formerly was inside <div id="header">
<hgroup>	Groups <h1> to <h6> elements
<keygen>	Containers for forms that identifies a key pair generator field
<mark>	Used to highlight parts of a text



<meter>	Represents fractional values in a graphically
<nav>	Groups the navigations section of the page
<output>	Container that represents the progress of a duty
<progress>	Used to represent the progress of a duty
<rp>	Tells the browser what to render if ruby annotations are not supported
<rt>	Represents pronunciation of ruby annotations
<ruby>	Defines ruby annotations
<section>	Represents a segment of a document
<source>	Defines media resources in <audio> or <video> elements
<summary>	Container to indicate the header for a <details> element
<time>	Defines date and time
<video>	Used to add video
<wbr>	Tells the browser when to add a line break



Oldal struktúrát meghatározó elemek



HTML bővítmények

► Kép jelölő elem

```
  
<p>Image of Mars. </p>
```

```
<figure>  
    
  <figcaption>  
    <p>This is an image of something interesting. </p>  
  </figcaption>  
</figure>
```





HTML bővítmények

► Szerkeszthető tartalom

```
<!DOCTYPE html>

<html lang="en">
<head>
    <meta charset="utf-8">
    <title>untitled</title>
</head>
<body>
    <h2> To-Do List </h2>
    <ul contenteditable="true">
        <li> Break mechanical cab driver. </li>
        <li> Drive to abandoned factory
        <li> Watch video of self </li>
    </ul>
</body>
</html>
```

To-Do List

- Break mechanical cab driver.
- Drive to abandoned factory
- Watch video of self

HTML bővítmények

► Email beviteli mező

```
<!DOCTYPE html>

<html lang="en">
<head>
  <meta charset="utf-8">
  <title>untitled</title>
</head>
<body>
  <form action="" method="get">
    <label for="email">Email:</label>
    <input id="email" name="email" type="email" />

    <button type="submit"> Submit Form </button>
  </form>
</body>
</html>
```

Email: Submit Form

nettutsplus.com is not a legal email address

HTML bővítmények

► Alapértelmezett szöveg

```
<input name="email" type="email" placeholder="doug@givethesepooleair.com" />
```

Chrome

Email:

Firefox

Email:

Opera

Email:

Safari

Email:

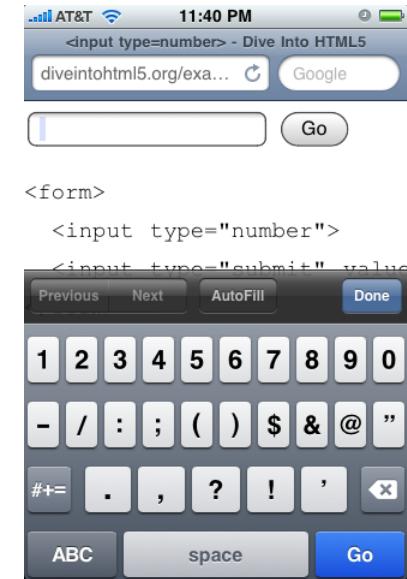
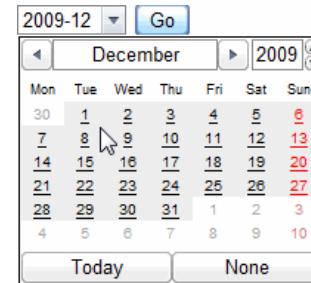




Úrlap elemek

- ▶ Specifikus beviteli mező: email, URL, number, search
- ▶ Csúszka
- ▶ Dátum választó
- ▶ Validáció

```
<form action="" method="post">
  <label for="username">Create a Username: </label>
  <input type="text"
    name="username"
    id="username"
    placeholder="4 <-> 10"
    pattern="[A-Za-z]{4,10}"
    autofocus
    required>
  <button type="submit">Go </button>
</form>
```





Szoftver eszköztár

- ▶ Offline alkalmazás
- ▶ Tárterület
- ▶ Kommunikáció
 - Web Workers
 - Web Sockets
- ▶ Asztali képességek
 - Drag and Drop
- ▶ Geolokáció



Offline alkalmazás

- ▶ Gyorsítótár kezelés (gyakran 5 MByte)
 - Offline böngészés
 - Gyorsítás
 - Szervert terhelés csökkentés
- ▶ Adatok (fájlok) kezelése
 - Mindég gyorsítótárazott (CACHE MANIFEST)
 - Csak hálózati hiba esetén (FALLBACK)
 - Nem gyorsítótárazható (NETWORK)

CACHE MANIFEST

#v1.01

#Explicitly cached files

CACHE:

index.html
Stylesheet.css
Images/logo.png

NETWORK:

Search.cfm
Login.cfm
/dynamicpages

FALLBACK:

/dynamicpage.cfm /static.html

```
<html  
  manifest="http://m.health.unm.ed  
u/someapp.manifest">  
...  
</html>
```



Helyi tárterület

- ▶ A sütiken túl – helyi tartós tár (window.localStorage)
 - JavaSzkriptből kezelhető (név/érték)
 - Perzisztens
 - 5MB terület “forrásonként”
 - Biztonságos (csak a böngészővel kommunikál)
- ▶ Viszony tár (window.sessionStorage)
 - A böngésző futása alatt él (név/érték)
 - minden oldal és lapfűl új viszony
- ▶ Böngésző alapú SQLite vagy IndexedDB



Példák

► Web tár

```
window.localStorage['value'] = 'Save this!';
```

► Viszony tár

```
sessionStorage.useLater('fullname', 'Garth Colasurdo');  
alert("Hello " + sessionStorage.fullname);
```

► Adatbázis tár

```
var database = openDatabase("Database Name", "Database Version");  
database.executeSql("SELECT * FROM test", function(result1) {
```



Helyi tár

HTML5 presentation

http://slides.html5rocks.com/#slide8

```
var db = window.openDatabase("Database Name", "Database Version");
db.transaction(function(tx) {
  tx.executeSql("SELECT * FROM test", [], successCallback, errorCallback);
});
```

Prepare presentation [Delete]

Practice presentation [Delete]

Make edits [Delete]

Give presentation [Delete]

[new to do item](#) [create table](#) [drop table](#)

See the generated database: Developer > Developer Tools > Storage

The screenshot shows the Chrome Developer Tools Storage panel. On the left, a tree view lists 'DATABASES' (DBTest), 'LOCAL STORAGE' (slides.html5rocks.com), 'SESSION STORAGE', and 'COOKIES'. The DBTest database is expanded, showing a table named 'Table1Test' with the following data:

	id	text
1	5510	Prepare presentation
2	2245	Practice presentation
3	4601	Make edits
4	8815	Give presentation



Kommunikáció

► Web workers

■ Háttérben futó JavaScript kód

```
<!DOCTYPE html>
<html>
<body>



Count numbers: <output id="result"></output></p>
<button onclick="startWorker()">Start Worker</button>
<button onclick="stopWorker()">Stop Worker</button>
<br><br>



<script>
var w;

function startWorker() {
    if(typeof(Worker) !== "undefined") {
        if(typeof(w) == "undefined") {
            w = new Worker("demo_workers.js");
        }
        w.onmessage = function(event) {
            document.getElementById("result").innerHTML = event.data;
        };
    } else {
        document.getElementById("result").innerHTML = "Sorry! No Web Worker support.";
    }
}

function stopWorker() {
    w.terminate();
    w = undefined;
}
</script>

</body>
</html>
```



Kommunikáció

► Web sockets

- Kétirányú kommunikáció HTTP felett
- Socket.send()
- Socket.receive()

```
<!DOCTYPE HTML>
<html>
  <head>

    <script type="text/javascript">
      function WebSocketTest()
      {
        if ("WebSocket" in window)
        {
          alert("WebSocket is supported by your Browser!");

          // Let us open a web socket
          var ws = new WebSocket("ws://localhost:9998/echo");

          ws.onopen = function()
          {
            // Web Socket is connected, send data using send()
            ws.send("Message to send");
            alert("Message is sent...");
          };

          ws.onmessage = function (evt)
          {
            var received_msg = evt.data;
            alert("Message is received...");
          };

          ws.onclose = function()
          {
            // websocket is closed.
            alert("Connection is closed...");
          };
        }
        else
        {
          // The browser doesn't support WebSocket
          alert("WebSocket NOT supported by your Browser!");
        }
      }
    </script>

  </head>
  <body>

    <div id="sse">
      <a href="javascript:WebSocketTest()">Run WebSocket</a>
    </div>

  </body>
</html>
```



D&D

► minden elemre alkalmazható


```
<!DOCTYPE HTML>
<html>
<head>
<script>
function allowDrop(ev) {
    ev.preventDefault();
}

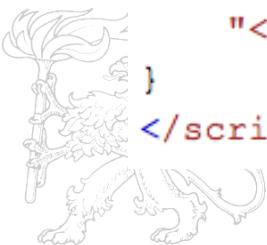
function drag(ev) {
    ev.dataTransfer.setData("text", ev.target.id);
}

function drop(ev) {
    ev.preventDefault();
    var data = ev.dataTransfer.getData("text");
    ev.target.appendChild(document.getElementById(data));
}
</script>
</head>
<body>

<div id="div1" ondrop="drop(event)" ondragover="allowDrop(event)"></div>



</body>
</html>
```



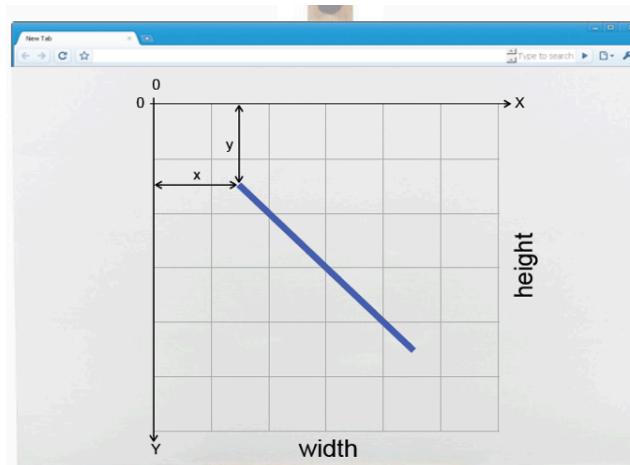
Geolokáció

- ▶ Felhasználó pozíciója lekérdezhető

```
<script>
var x = document.getElementById("demo");
function getLocation() {
    if (navigator.geolocation) {
        navigator.geolocation.getCurrentPosition(showPosition);
    } else {
        x.innerHTML = "Geolocation is not supported by this browser.";
    }
}
function showPosition(position) {
    x.innerHTML = "Latitude: " + position.coords.latitude +
    "<br>Longitude: " + position.coords.longitude;
}
</script>
```

Rajzolás

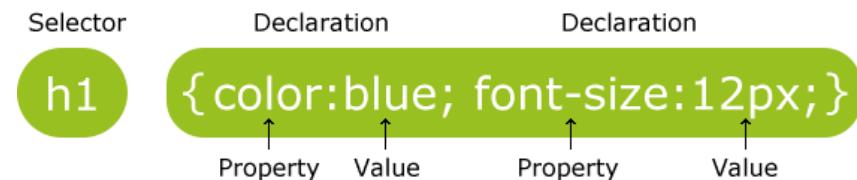
```
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #000000;">  
</canvas>  
  
var c = document.getElementById("myCanvas");  
var ctx = c.getContext("2d");  
ctx.moveTo(0,0);  
ctx.lineTo(200,100);  
ctx.stroke();
```



CSS

- ▶ Egy stílus leíró nyelv
- ▶ A HTML elemek kinézetét, stílusát adja meg (képernyőn, papíron, ...)
- ▶ Stílus alapon lehet egy helyről befolyásolni a kinézetet (pl.: IOS vs. Android)

```
body {  
    background-color: lightblue;  
}  
  
h1 {  
    color: white;  
    text-align: center;  
}  
  
p {  
    font-family: verdana;  
    font-size: 20px;  
}
```





CSS Selector

- ▶ A módosítandó elemet keresi meg
 - Elem kiválasztó (element selector)

```
p {  
    text-align: center;  
    color: red;  
}  
  
h1, h2, p {  
    text-align: center;  
    color: red;  
}
```

- Azonosító kiválasztó (id selector)

```
#paral {  
    text-align: center;  
    color: red;  
}
```

- Osztály kiválasztó (class selector)

```
.center {  
    text-align: center;  
    color: red;  
}
```

```
<p class="center large">This paragraph refers to two classes.</p>
```



CSS alkalmazása

► Külső

```
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
```

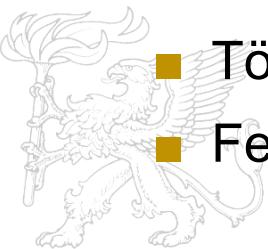
► Belső

```
<head>
<style>
body {
    background-color: linen;
}

h1 {
    color: maroon;
    margin-left: 40px;
}
</style>
</head>
```

► Beágyazott

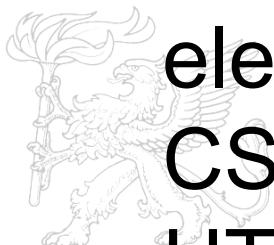
```
<h1 style="color:blue;margin-left:30px;">This is a heading.</h1>
```



CSS3

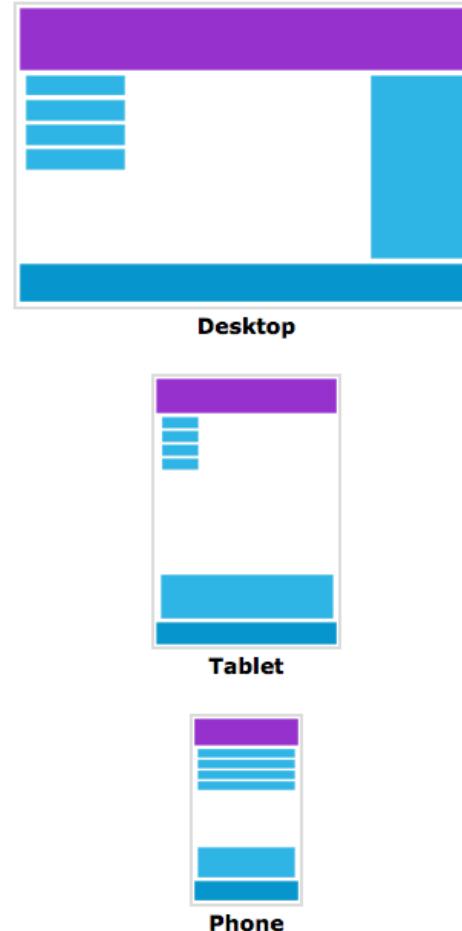
► Modulokra bontott

- Választó (Selector)
- Doboz modell (Box model)
- Háttér és határok (Backgrounds and borders)
- Képek és helyettesítő tartalmak (Images and Replaced Content)
- Szöveg hatások (Text effects)
- 2D/3D transzformáció (2D/3D transformations)
- Animáció (Animations)
- Több tömbös kialakítás (Multiple Column Layout)
- Felhasználói Felületek (user Interfaces)



Reszponzív CSS

- ▶ minden eszközön használhatónak kell lennie a weboldalnak
- ▶ Ehhez elegendő a CSS és a HTML használata

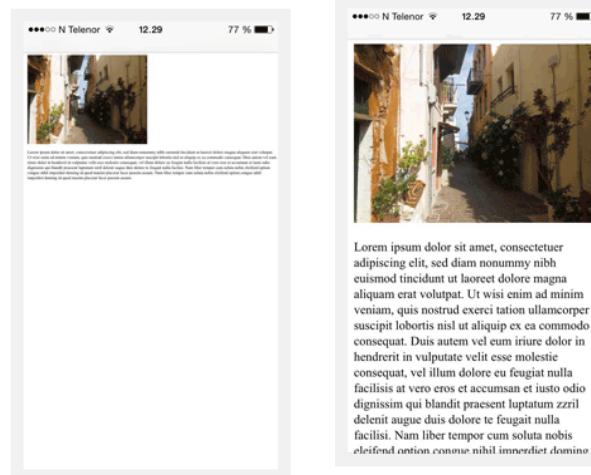




Nézetaljak (Viewport)

- ▶ A weboldal felhasználó által látható része
- ▶ Eszközfüggő
- ▶ HTML viewport meta elem

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```





Reszponzív alkalmazás

- ▶ A felhasználóknak a horizontális görgetés megszokott, a vertikális nem
- ▶ Ökölszabályok
 - Ne használjunk nagy rögzített szélességű elemeket
 - A tartalom megjelenítését ne egy adott szélességhez optimalizáljuk
 - Használjuk a CSS média lekérdezését, hogy e szerint alakítsuk a stílust a képernyő mérethez
 - Használjunk relatív méretezést, pozicionálást

Reszponzív web alapok

- ▶ Rács nézet
- ▶ Relatív arányok



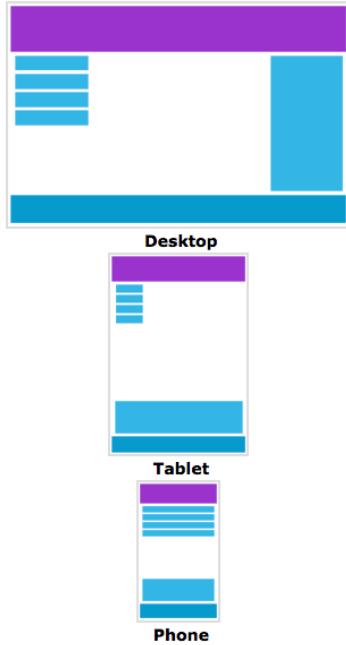
```
html {  
    font-family: "Lucida Sans", sans-serif;  
}  
.header {  
    background-color: #9933cc;  
    color: #ffffff;  
    padding: 15px;  
}  
.menu {  
    width: 25%;  
    float: left;  
}  
.main {  
    width: 75%;  
    float: left;  
}  
  
* {  
    box-sizing: border-box;  
}  
  
<div class="row">  
    <div class="col-3">...</div>  
    <div class="col-9">...</div>  
</div>
```

```
.header {  
    background-color: #9933cc;  
    color: #ffffff;  
    padding: 15px;  
}  
.menu {  
    width: 25%;  
    float: left;  
}  
.main {  
    width: 75%;  
    float: left;  
}  
  
* {  
    box-sizing: border-box;  
}  
  
<div class="row">  
    <div class="col-3">...</div>  
    <div class="col-9">...</div>  
</div>
```

```
html {  
    font-family: "Lucida Sans", sans-serif;  
}  
.header {  
    background-color: #9933cc;  
    color: #ffffff;  
    padding: 15px;  
}  
.menu {  
    width: 25%;  
    float: left;  
}  
.main {  
    width: 75%;  
    float: left;  
}  
  
* {  
    box-sizing: border-box;  
}  
  
<div class="row">  
    <div class="col-3">...</div>  
    <div class="col-9">...</div>  
</div>
```

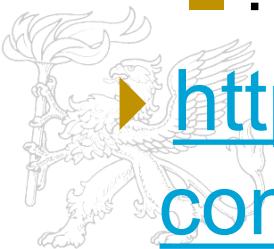


Médium lekérdezés



```
<div class="row">
<div class="col-3 col-m-3">...</div>
<div class="col-6 col-m-9">...</div>
<div class="col-3 col-m-12">...</div>
</div>
```

```
/* For mobile phones: */
[class*="col-"] {
    width: 100%;
}
@media only screen and (min-width: 600px) {
    /* For tablets: */
    .col-m-1 {width: 8.33%;}
    .col-m-2 {width: 16.66%;}
    .col-m-3 {width: 25%;}
    .col-m-4 {width: 33.33%;}
    .col-m-5 {width: 41.66%;}
    .col-m-6 {width: 50%;}
    .col-m-7 {width: 58.33%;}
    .col-m-8 {width: 66.66%;}
    .col-m-9 {width: 75%;}
    .col-m-10 {width: 83.33%;}
    .col-m-11 {width: 91.66%;}
    .col-m-12 {width: 100%;}
}
@media only screen and (min-width: 768px) {
    /* For desktop: */
    .col-1 {width: 8.33%;}
    .col-2 {width: 16.66%;}
    .col-3 {width: 25%;}
    .col-4 {width: 33.33%;}
    .col-5 {width: 41.66%;}
    .col-6 {width: 50%;}
    .col-7 {width: 58.33%;}
    .col-8 {width: 66.66%;}
    .col-9 {width: 75%;}
    .col-10 {width: 83.33%;}
    .col-11 {width: 91.66%;}
    .col-12 {width: 100%;}
}
```



Reszponzív keretrendszerek

- ▶ Szabványos stílusok, minták, koncepciók
- ▶ Sok CSS keretrendszer létezik:
 - Bootstrap -
 - Foundation
 - Skeleton
 - ...
- ▶ <http://www.vermillion.com/responsive-comparison/>



Bootstrap

- ▶ Nyílt forrású felhasználói felület keretrendszer (Twitter 2011- , github)
- ▶ HTML, CSS minták a különböző elemekre
- ▶ Könnyű reszponzív felületet készíteni
- ▶ Mobilt elsőként (Mobile - First) elvet helyezi előtérbe
- ▶ A legtöbb böngészőben megfelelően működik

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Bootstrap Example</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
  <script src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
</head>
<body>

  <div class="container">
    <h1>My First Bootstrap Page</h1>
    <p>This is some text.</p>
  </div>

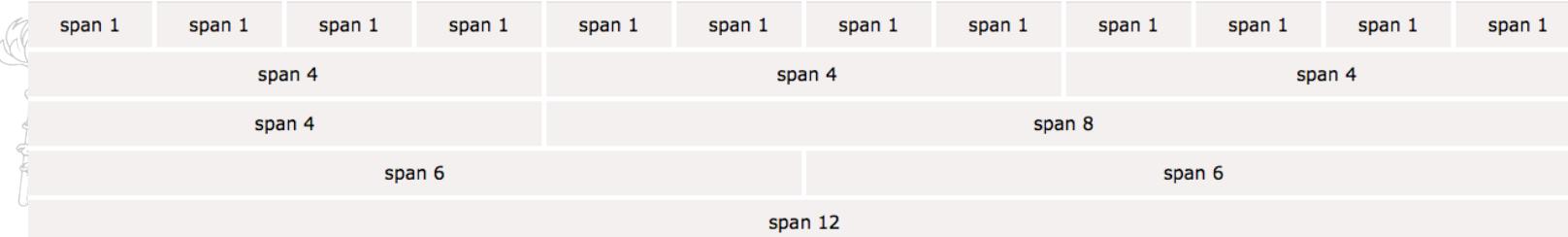
</body>
</html>
```

Bootstrap - rácsok

- ▶ 12 sozolpos nézet
- ▶ Oszlop kategóriák
 - xs - telefon
 - sm - tablet
 - md - asztali
 - lg – nagy asztali
- ▶ A lehetséges max határozza meg a stílust

```
<div class="row">
  <div class="col-sm-4">.col-sm-4</div>
  <div class="col-sm-4">.col-sm-4</div>
  <div class="col-sm-4">.col-sm-4</div>
</div>

<div class="row">
  <div class="col-sm-4">.col-sm-4</div>
  <div class="col-sm-8">.col-sm-8</div>
</div>
```





Rács szabályok

- ▶ A soroknak konténerben kell lennie
 - .container (fix)
 - .container-fliud (teljes szélesség)
- ▶ Sorokkal készíthetünk horizontális oszlop csoportokat
- ▶ A tartalmat az oszoplokba helyezzük, a soroknak csak oszolpokat szabad tartalmaznia
- ▶ Kész sor, oszlop osztályokat használhatunk (.row, .col-sm-4,...)
- ▶ Az oszolpok között padding-gal definiálhatunk térközt
- ▶ Összefogó oszlopok esteén a 12-ből lefedett oszlopok számát kell megadnunk



Horizontálisan rétegelt

```
<div class="row">
    <div class="col-sm-4 col-sm-push-8">.col-sm-4 .col-sm-push-8</div>
    <div class="col-sm-8 col-sm-pull-4">.col-sm-8 .col-sm-pull-4</div>
</div>

<div class="row">
    <div class="col-xs-9 col-md-7">.col-xs-9 .col-md-7</div>
    <div class="col-xs-3 col-md-5">.col-xs-3 .col-md-5</div>
</div>

<div class="row">
    <div class="col-xs-6 col-md-10">.col-xs-6 .col-md-10</div>
    <div class="col-xs-6 col-md-2">.col-xs-6 .col-md-2</div>
</div>

<div class="row">
    <div class="col-xs-6">.col-xs-6</div>
    <div class="col-xs-6">.col-xs-6</div>
</div>

<div class="row">
    <div class="col-xs-7 col-sm-6 col-lg-8">.col-xs-7 .col-sm-6 .col-lg-8</div>
    <div class="col-xs-5 col-sm-6 col-lg-4">.col-xs-5 .col-sm-6 .col-lg-4</div>
</div>

<div class="row">
    <div class="col-xs-6 col-sm-8 col-lg-10">.col-xs-6 .col-sm-8 .col-lg-10</div>
    <div class="col-xs-6 col-sm-4 col-lg-2">.col-xs-6 .col-sm-4 .col-lg-2</div>
</div>
```



Bootstrap képességek

► Gyakorlatilag minden HTML elemhez

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Doooley	july@example.com

Class	Description
.active	Applies the hover color to the table row or table cell
.success	Indicates a successful or positive action
.info	Indicates a neutral informative change or action
.warning	Indicates a warning that might need attention
.danger	Indicates a dangerous or potentially negative action

```
<ul class="pagination">
  <li><a href="#">1</a></li>
  <li><a href="#">2</a></li>
  <li><a href="#">3</a></li>
  <li><a href="#">4</a></li>
  <li><a href="#">5</a></li>
</ul>
```

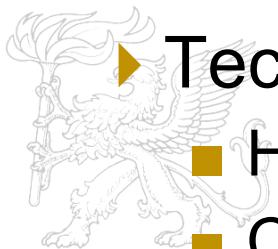


```
<ul class="list-group">
  <li class="list-group-item">New <span class="badge">12</span></li>
  <li class="list-group-item">Deleted <span class="badge">5</span></li>
  <li class="list-group-item">Warnings <span class="badge">3</span></li>
</ul>
```

New	12
Deleted	5
Warnings	3

```
<ul class="nav nav-tabs">
  <li class="active"><a href="#">Home</a></li>
  <li class="dropdown">
    <a class="dropdown-toggle" data-toggle="dropdown" href="#">Menu 1</a>
    <span class="caret"></span>
    <ul class="dropdown-menu">
      <li><a href="#">Submenu 1-1</a></li>
      <li><a href="#">Submenu 1-2</a></li>
      <li><a href="#">Submenu 1-3</a></li>
    </ul>
  </li>
  <li><a href="#">Menu 2</a></li>
  <li><a href="#">Menu 3</a></li>
</ul>
```

Home Menu 1 ▾ Menu 2 Menu 3



Összefoglaló

► Trendek:

- Gazdag Internet Alkalmazás (Rich Internet Application)
- Reszponzív Web alkalmazás (Responsive Application)
- Hibrid Mobil Alkalmazás (Hybrid Mobile Application)
- Progresszív Mobil Alkalmazás (Progressive Web Application)

► Technológiák

- HTML 5
- CSS 3
- BOOSTRAP