



Razvoj mobilnih aplikacija i servisa

Arhitektura mobilnih aplikacija

**Katedra za računarstvo
Elektronski fakultet u Nišu**



Izbor arhitekture mobilne aplikacije

- ✿ **Mobilna strategija** – razmatranje parametara u izboru arhitekture i karakteristika mobilne aplikacije i servisa

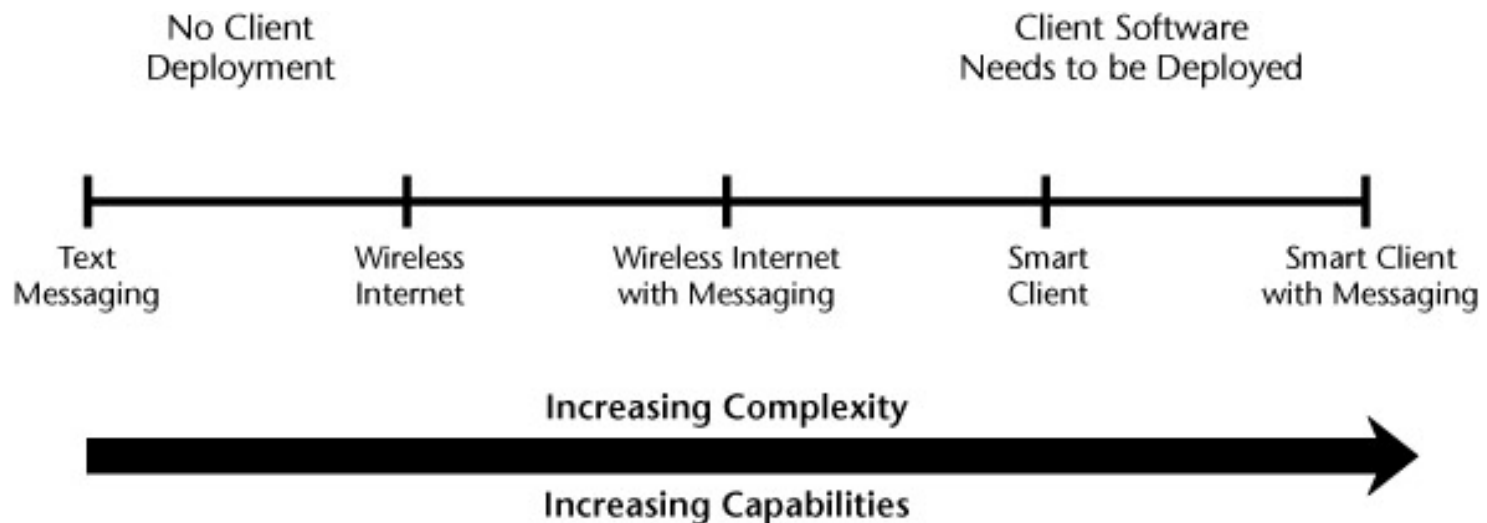
- ✦ Korisnici aplikacije
- ✦ Tip(-ovi) uređaja
- ✦ Konekcija i integracija sa IT sistemima kompanije
- ✦ Pristup podacima kompanije
- ✦ Obaveštavanje/notifikacija korisnika
- ✦ Sigurnost
- ✦ Trajanje baterije
- ✦ ...

- ✿ **Pravila mobilne strategije**

- ✦ Ne možete podržati sve ...
- ✦ Kreirati, a ne konvertovati postojeću aplikaciju
- ✦ Održavati je **jednostavnom i upotrebljivom!**

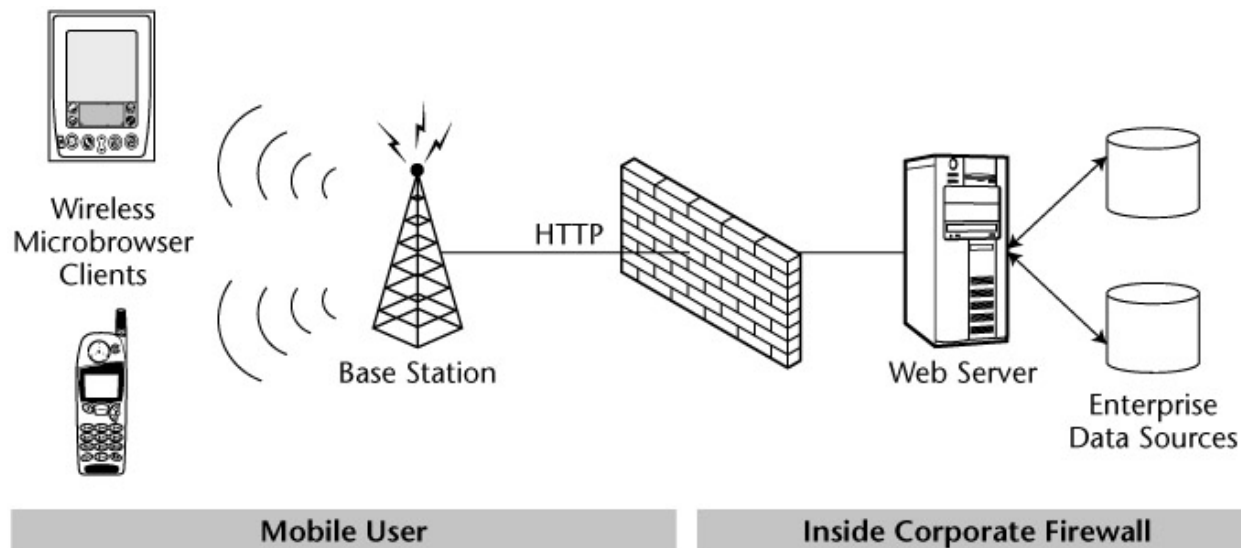
Arhitektura mobilnih aplikacija

- Bežične Internet aplikacije - mobilni Web sajtovi - aplikacije zasnovane na mobilnom web *browser-u*
- Mobilne Web aplikacije – HTML5, CSS3, JavaScript/AJAX
- Hibridne mobilne aplikacije - zasnovane na Web View-u
- Kros-platformske mobilne aplikacije
- Nativne mobilne aplikacije



Mobilni Web sajtovi – bežične Internet aplikacije

- Bežične Internet aplikacije imaju istu arhitekturu kao i klasične Web aplikacije
- Osnovne komponente bežične Internet arhitekture.
 - Web browser klijent
 - Web server
 - Izvori podataka i aplikaciona logika na Web serveru



Arhitektura mobilnih aplikacija

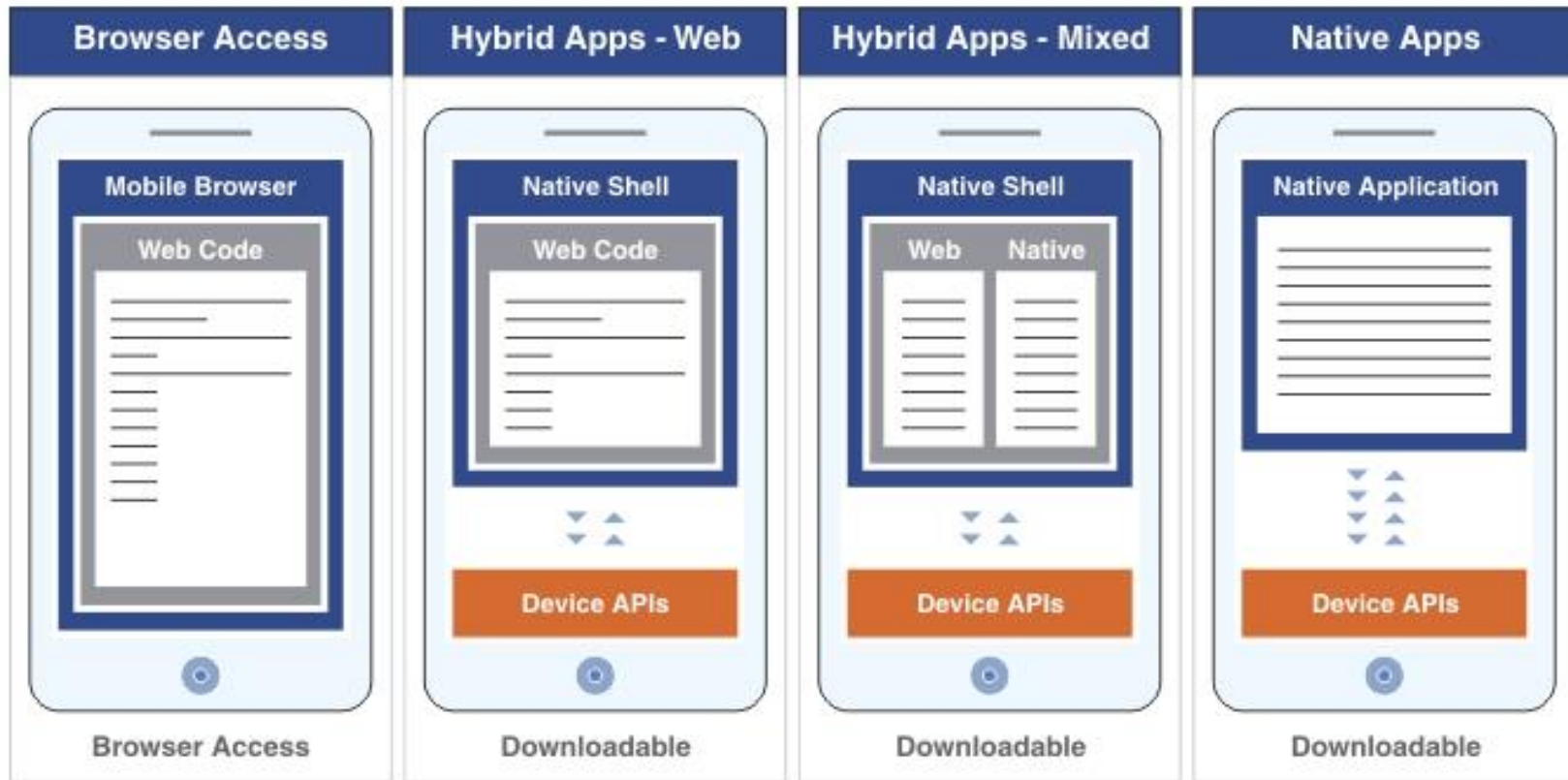
Razvoj mobilnih aplikacija i servisa



Mobilni Web sajtovi – bežične Internet aplikacije

- ✿ **Wireless Application Protocol (WAP)**
 - ✦ Open Mobile Alliance (ranije WAP Forum)
- ✿ **Korisnički aplikacioni servisi zasnovani na arhitekturi bežičnog Interneta**
 - ✦ **mMode** razvijen od strane AT&T Wireless (SAD) , zasnovan na WAP/WML
 - ✦ **i-mode** razvijen od strane NTT DoCoMo (Japan) zasnovan na C-HTML i sopstvenim protokolima (ALP, TLP)
- ✿ **HTTP/HTML - *Responsive Web Design***

Arhitektura mobilnih aplikacija





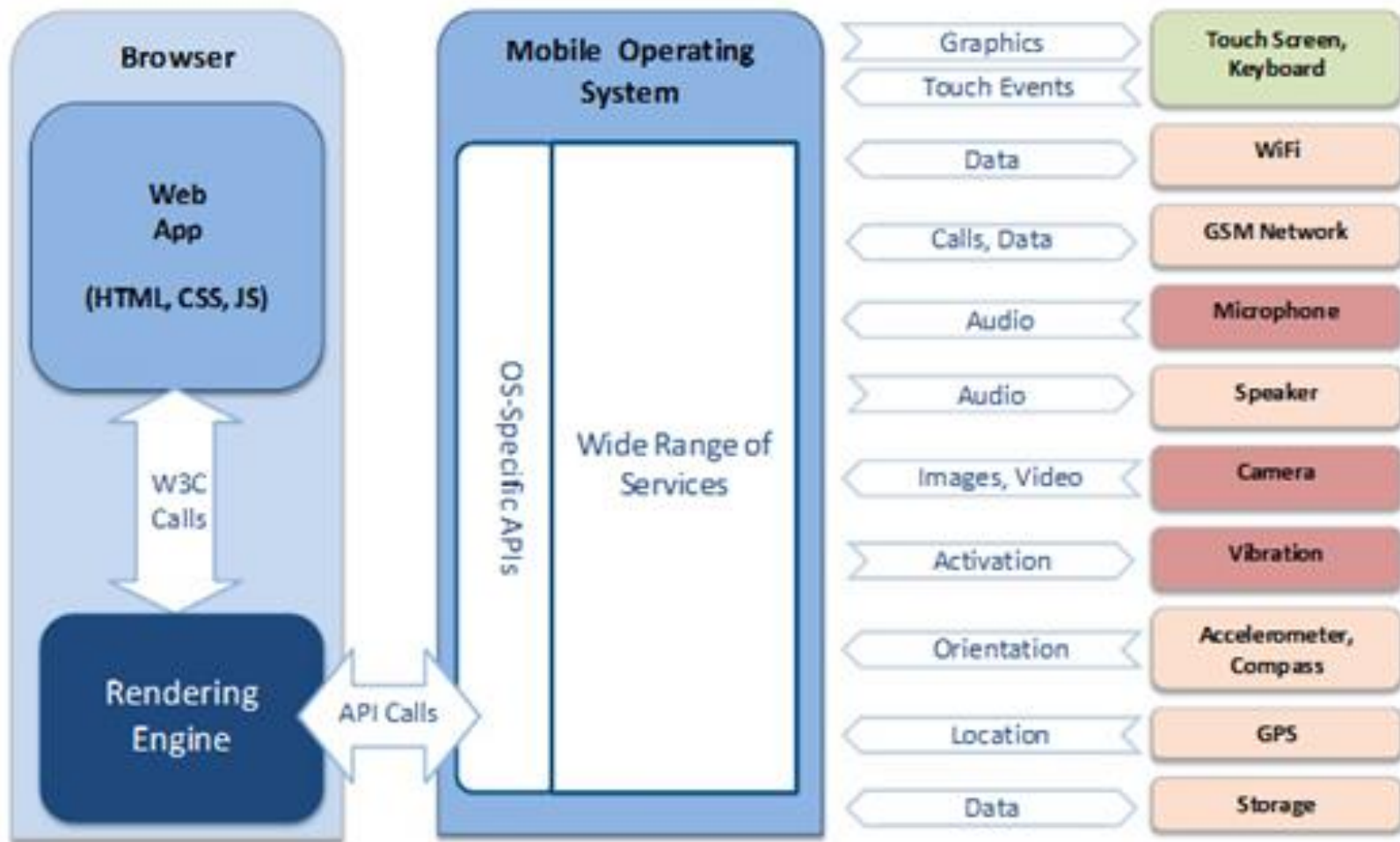
MOBILNE WEB APLIKACIJE



Mobilne Web aplikacije

- ✿ Zasnovane na - HTML5, CSS3 i JavaScript
- ✿ HTML5 & CSS3
 - ✦ Crtanje po canvasu (2D Drawing API):
 - ✦ Touch događaji
 - ✦ Geolocation API (GPS / Celularna mreža ili WiFi)
 - ✦ Aplikacioni keš
 - ✦ Lokalno smeštanje podataka
 - Web Store
 - Web SQL Database
 - Indexed Database (INDEXEDDB)
 - ✦ Multimedia
 - Podrška za video i audio tagove
- ✿ JavaScript/AJAX
- ✿ JavaScript okviri i biblioteke
 - ✦ Angular, Vue.js, jQuery Mobile, React, Sencha Touch, Ionic,...

Mobilne Web aplikacije



Arhitektura mobilnih aplikacija

Razvoj mobilnih aplikacija i servisa



What Web Can Do Today?

Seamless Experience

Offline Mode	YES ✓
Background Sync	YES ✓
Inter-App Sharing	YES ✓
Payments	YES ✓
Credentials	YES ✓

Native Behaviors

Local Notifications	YES ✓
Push Messages	YES ✓
User Idle Detection	YES ✓
Permissions	YES ✓
Task Scheduling	YES ✓

Camera & Microphone

Audio & Video Capture	YES ✓
Advanced Camera Controls	YES ✓
Recording Media	YES ✓
Real-Time Communication	YES ✓
Shape Detection	NO ✗

App Lifecycle

Store Distribution	not testable
Home Screen Installation	YES ✓
Run On Startup	not testable
Foreground Detection	YES ✓
Freeze/Resume Detection	YES ✓

Surroundings

Bluetooth	YES ✓
NFC	NO ✗
USB	YES ✓
Serial Port	YES ✓
Ambient Light	NO ✗

Operating System

Offline Storage	YES ✓
File Access	YES ✓
Contacts	NO ✗
SMS/MMS	NO ✗
Storage Quotas	YES ✓

Device Features

Network Type & Speed	YES ✓
Online State	YES ✓
Vibration	YES ✓
Battery Status	YES ✓
Device Memory	YES ✓

Location & Position

Geolocation	YES ✓
Geofencing	NO ✗
Device Position	YES ✓
Device Motion	YES ✓
Proximity Sensors	NO ✗

Screen & Output

Virtual & Augmented Reality	YES ✓
Fullscreen	YES ✓
Screen Orientation & Lock	YES ✓
Wake Lock	YES ✓
Presentation Features	YES ✓

Input

Touch Gestures	YES ✓
Speech Recognition	YES ✓
Clipboard (Copy & Paste)	YES ✓
Pointing Device Adaptation	YES ✓
Eye Dropper	YES ✓

<https://whatwebcando.today/>

Arhitektura mobilnih aplikacija

Razvoj mobilnih aplikacija i servisa

Mobilne Web aplikacije

Prednosti

- ✚ Nema potrebe za instaliranjem aplikacije
- ✚ Lako održavanje i ažuriranje bez potrebe za novim instaliranjem
- ✚ Podrška za kros-platform na svakom uređaju koji ima browser

Nedostaci

- ✚ Nema mogućnosti pristupa nativnom API-u i karakteristikama uređaja/platforme
- ✚ Ograničeni UI koji nema nativni *look & feel*
- ✚ Slabije performanse zavisne od pristupa preko bežične mreže
- ✚ Ne može se distribuirati preko Play/App store-a

Web standardi

- ✿ HTML, XHTML, WML, XHTML Basic, XHTML-MP, itd.
 - ✦ *Standards and drafts* - <http://www.w3.org/TR/>
- ✿ CSS: Cascading Style Sheets
 - ✦ CSS predstavlja jezik za definisanje stila prikaza HTML dokumenta
 - ✦ <http://www.w3.org/Style/CSS/>
- ✿ JavaScript
 - ✦ JavaScript (JS) je *open source* programski jezik implementiran u okviru Web browser-a u cilju kreiranja kompleksnijeg korisničkog interfejsa i dinamičkih Web strana
 - ✦ JavaScript, JSON, AJAX,...
 - ✦ ECMAScript 6 (2015), ECMAScript 2017 (2018, 2019, 2020,...), TypeScript, CoffeeScript, Dart,...
 - ✦ <https://developer.mozilla.org/en/JavaScript>

HTML5

- ❁ HTML5 predstavlja najnoviji HTML standard
 - ❁ (HTML 5.2 - W3C Recommendation, December 2018)
 - ❁ <https://www.w3.org/TR/html/>
 - ❁ <https://developers.google.com/web/> (<http://www.html5rocks.com/>)
 - ❁ http://www.w3schools.com/html/html5_intro.asp
- ❁ Pravila definisana u okviru HTML5 standarda:
 - ❁ Nove karakteristike treba da budu zasnovane na HTML, CSS, DOM i JavaScript-u
 - ❁ Redukuje se potreba za eksternim plug-inovima (npr. Flash)
 - ❁ Bolje rukovanje greškama
 - ❁ Više novih markup tagova kojima se zamenjuje upotreba script koda
 - ❁ HTML5 treba da bude nezavisan od uređaja/browser-a
 - ❁ Razvojni proces treba da bude otvoren za javnost (transparentan)

Istorijat Web tehnologija

Rough Timeline of Web Technologies

1991	HTML
1994	HTML 2
1996	CSS 1 + JavaScript
1997	HTML 4
1998	CSS 2
2000	XHTML 1
<i>2002</i>	Tableless Web Design
<i>2005</i>	AJAX
<i>2009</i>	HTML 5

Source: html5rocks.com

HTML5 – nove karakteristike

- ✿ 2D i 3D grafika
 - ✦ <canvas> element, SVG, WebGL (3D), 3D CSS Transforms, i SMIL
- ✿ <video> i <audio> elementi za prikaz i reprodukovanje multimedijalnih sadržaja
- ✿ Podrška za lokalno smeštanje podataka
 - ✦ Aplikacioni keš (*Application cache*)
 - ✦ Web storage: localStorage, sessionStorage
 - ✦ Indexed DB API (Indexed Database API) (ranije Web SQL)
- ✿ Geolocation API
- ✿ Novi semantički i elementi za specifične sadržaje:
 - ✦ <header>, <section>, <footer>, <aside>, <nav>, <main>, <article>, <figure>,...
- ✿ Novi ulazni tipovi i kontrole na formi:
 - ✦ calendar, date, time, email, tel, url, search, datalist, keygen...
- ✿ Device API
 - ✦ Camera API, Touch Events, Device Orientation,...
- ✿ Push API, Notifications API, Web Payments API,...
- ✿ WebRTC, Media capture and recording APIs



Osnovni HTML5

✿ http://www.w3schools.com/html/html5_intro.asp

```
<!DOCTYPE html>
```

```
<html>
```

```
  <meta charset="UTF-8">
```

```
  <head>
```

```
    <title> Title of the document </title>
```

```
  </head>
```

```
  <body>
```

```
    The content of the document.....
```

```
  </body>
```

```
</html>
```


HTML5 – Canvas i SVG

- HTML5 `<canvas>` element se koristi za iscrtavanje 2D i 3D grafike korišćenjem skript koda (obično JavaScript).
- Canvas uključuje više metoda za iscrtavanje linija, *pravougaonika, krugova, teksta*, i dodavanje *slika*.
- SVG (*Scalable Vector Graphics*) je XML jezik za prikaz 2D

```
<canvas id="myCanvas" width="200" height="100"  
  style="border:1px solid #000000;">  
</canvas>
```

```
<script>  
  var c = document.getElementById("myCanvas");  
  var ctx = c.getContext("2d");  
  ctx.fillStyle = "#FF0000";  
  ctx.fillRect(0,0,150,75);  
</script>
```



HTML5 integrisani SVG

```
<!DOCTYPE html>
```

```
<html>
```

```
  <body>
```

```
    <svg xmlns="http://www.w3.org/2000/svg" version="1.1"  
    height="190">
```

```
      <polygon points="100,10 40,180 190,60 10,60 160,180"  
      style="fill:lime;stroke:purple;stroke-width:5;fill-rule:evenodd;">
```

```
    </svg>
```

```
  </body>
```

```
</html>
```





HTML5 aplikacioni keš

- HTML5 uvodi aplikacioni keš, čime omogućava da se kešira kod/sadržaj u okviru Web aplikacije i da taj kod/sadržaj bude dostupan i bez Internet konekcije.
- Aplikacioni keš obezbeđuje tri prednosti za mobilne Web aplikacije:
 - Offline* pregledanje (*browsing*) – korisnici mogu da koriste aplikaciju i kada su *offline*
 - Brzina – keširani resursi se brže učitavaju
 - Redukovano opterećenje servera - browser će *download*-ovati samo ažurirane/promenjene resurse sa servera
- Cache Manifest* datoteka – tekst datoteka koja definiše koji sadržaj će biti keširan
 - 3 sekcije: CACHE MANIFEST, NETWORK, FALLBACK

```
<!DOCTYPE HTML>
<html manifest="demo.appcache">
...
</html>
```

```
CACHE MANIFEST
# 2012-02-21 v1.0.0
/theme.css
/logo.gif
/main.js
```

```
NETWORK:
login.asp
```

```
FALLBACK:
/html/ /offline.html
```

HTML5 – storage

Local Storage & Session Storage

JS Web Storage

```
// use localStorage for persistent storage
// use sessionStorage for per tab storage
saveButton.addEventListener('click', function () {
    window.localStorage.setItem('value', area.value);
    window.localStorage.setItem('timestamp', (new Date()).getTime());
}, false);
textarea.value = window.localStorage.getItem('value');
```

Web SQL & IndexedDB

JS Web SQL Database

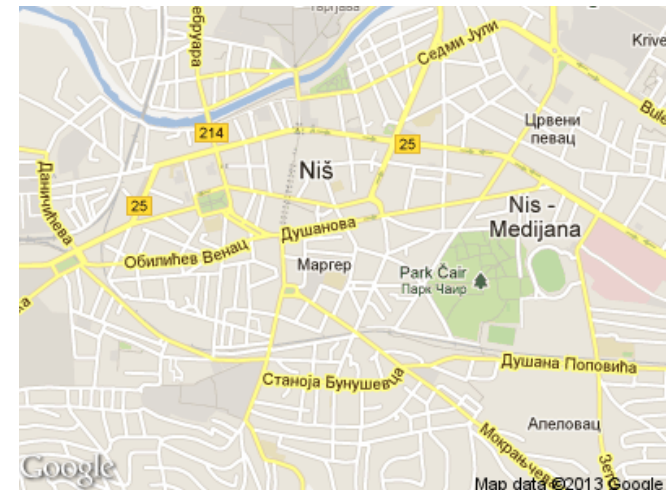
```
var db = window.openDatabase("DBName", "1.0", "description", 5*1024*1024); //5MB
db.transaction(function(tx) {
    tx.executeSql("SELECT * FROM test", [], successCallback, errorCallback);
});
```

HTML5 – Geolocation

JS Geolocation

```
if (navigator.geolocation) {  
    navigator.geolocation.getCurrentPosition(function(position) {  
        var latLng = new google.maps.LatLng(  
            position.coords.latitude, position.coords.longitude);  
        var marker = new google.maps.Marker({position: latLng, map: map});  
        map.setCenter(latLng);  
    }, errorHandler);  
}
```

```
<script>  
var x=document.getElementById("demo");  
function getLocation()  
{  
    if (navigator.geolocation)  
    {  
        navigator.geolocation.getCurrentPosition(showPosition, showError);  
    }  
    else{x.innerHTML="Geolocation is not supported by this browser.";}  
}  
  
function showPosition(position)  
{  
    var latlon=position.coords.latitude+","+position.coords.longitude;  
  
    var img_url="http://maps.googleapis.com/maps/api/staticmap?center="+  
    +latlon+"&zoom=14&size=400x300&sensor=false";  
    document.getElementById("mapholder").innerHTML="<img src='"+img_url+"'>";  
}
```



HTML5 – Geolocation

- *watchPosition()* – vraća tekuću poziciju korisnika i nastavlja da vraća ažurne pozicije kako se korisnik kreće (npr. u vozilu dok ima uključen GPS).
- *clearWatch()* – Zaustavlja *watchPosition()* metod.

```
<script>
var x=document.getElementById("demo");
function getLocation()
{
    if (navigator.geolocation)
    {
        navigator.geolocation.watchPosition(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>
```

HTML5 – Govorni ulaz & tipovi ulaza

HTML Speech Input

```
<input type="text" x-webkit-speech />
```

HTML Form field types on mobile

type="text"



Android Device

type="number"



Android Device

type="email"



iPhone Device

type="tel"



iPhone Device

HTML5 – audio & video

HTML

JS

Audio + Video

```
<audio id="audio" src="sound.mp3" controls></audio>  
document.getElementById("audio").muted = false;
```

```
<video id="video" src="movie.webm" autoplay controls></video>  
document.getElementById("video").play();
```


HTML5 – animacija

CSS Animations

```
@-webkit-keyframes pulse {  
  from {  
    opacity: 0.0;  
    font-size: 100%;  
  }  
  to {  
    opacity: 1.0;  
    font-size: 200%;  
  }  
}  
  
div {  
  -webkit-animation-name: pulse;  
  -webkit-animation-duration: 2s;  
  -webkit-animation-iteration-count: infinite;  
  -webkit-animation-timing-function: ease-in-out;  
  -webkit-animation-direction: alternate;  
}
```

HTML5 Web worker-i

- ❁ **Web worker** je JavaScript kod koji se izvršava u pozadini, nezavisno od ostalih skript metoda, bez narušavanja performansi Web stranice.
- ❁ Kada se izvršava skript funkcija u okviru HTML stranice, stranica postaje "blokirana" za interakciju dok se skript funkcija ne završi.
- ❁ Ukoliko izvršava **Web worker**, korisnik je u mogućnosti da nastavi sa interakcijom: klikom, selektovanjem elemenata stranice, itd., dok se Web worker izvršava u pozadini.



HTML5 Web worker - primer

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

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

<p><strong>Note:</strong> Internet Explorer 9 and earlier versions do not support Web
Workers.</p>

<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, your browser does not support Web
Workers...";
    }
}

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

</body>
</html>
```

demo_workers.js

```
var i = 0;

function timedCount() {
    i = i + 1;
    postMessage(i);
    setTimeout("timedCount()",500);
}

timedCount();
```

HTML5 Server-Sent Events

- HTML5 Server-Sent Events omogućava da Web aplikacija/stranica dobija ažurne podatke sa servera.
- Primer: Facebook/Twitter ažuriranja, ažuriranje cene akcija na berzi, tokovi vesti (*news feeds*), sportski rezultati, itd.

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

<h1>Getting server updates</h1>
<div id="result"></div>

<script>
if(typeof(EventSource) !== "undefined") {
    var source = new EventSource("demo_sse.php");
    source.onmessage = function(event) {
        document.getElementById("result").innerHTML += event.data + "<br>";
    };
} else {
    document.getElementById("result").innerHTML = "Sorry, your browser does
not support server-sent events...";
}
</script>

</body>
</html>
```

```
<?php
header('Content-Type: text/event-stream');
header('Cache-Control: no-cache');

$time = date('r');
echo "data: The server time is: {$time}\n\n";
flush();
?>
```

demo_sse.php

```
<%
Response.ContentType = "text/event-stream"
Response.Expires = -1
Response.Write("data: " & now())
Response.Flush()
%>
```

demo_sse.asp

HTML5 - WebSocket

- ✿ Otvaranje *WebSocket* konekcije postiže se jednostavnim pozivom *WebSocket* konstruktora

```
var connection = new WebSocket('ws://html5rocks.websocket.org/echo',  
['soap', 'xmpp']);
```

- ✿ Dok je konekcija otvorena moguće je slati podatke na server i primiti podatke sa servera.

```
connection.onopen = function () {  
    connection.send('Ping'); // Send the message 'Ping' to the server  
};  
  
// Log errors  
connection.onerror = function (error) {  
    console.log('WebSocket Error ' + error);  
};  
  
// Log messages from the server  
connection.onmessage = function (e) {  
    console.log('Server: ' + e.data);  
};
```

HTML5 – WebSocket (2)

- Slanje podataka na server obavlja se pozivom *send('your message')* metode *connection* objekta

```
connection.send('your message');

// Sending canvas ImageData as ArrayBuffer
var img = canvas_context.getImageData(0, 0, 400, 320);
var binary = new Uint8Array(img.data.length);
for (var i = 0; i < img.data.length; i++) {
    binary[i] = img.data[i];
}
connection.send(binary.buffer);

// Sending file as Blob
var file = document.querySelector('input[type="file"]').files[0];
connection.send(file);
```

- Takođe i server u svakom trenutku može slati poruke Web klijentu.
- Svaki put kad stigne poruka od servera, poziva se *onmessage* callback metoda, koja kao ulazni argument dobija *event* objekat, i poruci pristupa preko *data* atributa ovog objekta.

Mobilni Web – razvojni alati

- ✚ HTML5/CSS3/ JavaScript - AJAX
- ✚ W3C Mobile Web Best Practices (Application Best Practices)
 - ✚ <http://www.w3.org/TR/mobile-bp/>
 - ✚ <http://www.w3.org/TR/mwabp/>
- ✚ Wireframe/prototip alati (Wireframer, NinjaMock, ...)
- ✚ iUI - User Interface Framework for Mobile Web apps
 - ✚ <http://www.iui-js.org/>
- ✚ Mobilni Web aplikacioni okviri
 - ✚ Ionic - <http://ionicframework.com/>
 - ✚ jQuery Mobile - <http://jquerymobile.com/>
 - ✚ Sencha Touch - <http://www.sencha.com/products/touch/>
 - ✚ Dojo Mobile - <http://dojotoolkit.org/features/mobile>
 - ✚ ...
- ✚ Mobilni hibridni aplikacioni okviri
 - ✚ PhoneGap - Apache Cordova <http://phonegap.com/>
<http://cordova.apache.org/>



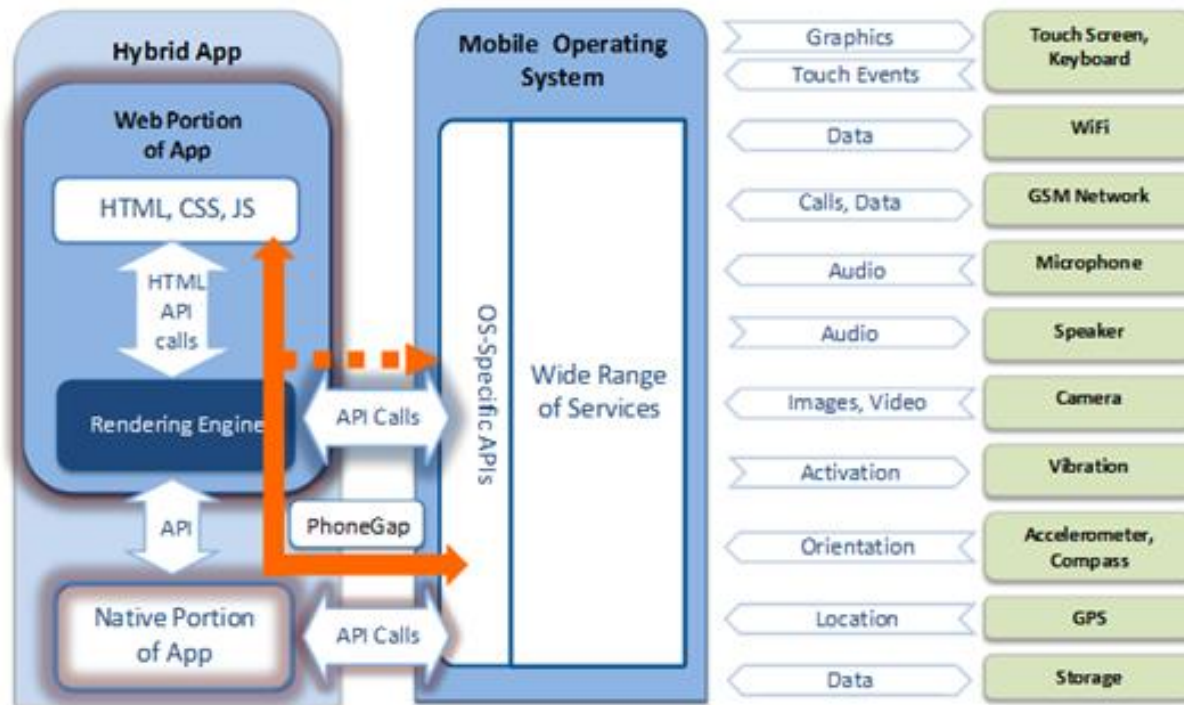
KROS-PLATFORMSKE MOBILNE APLIKACIJE

Kros-platformske mobilne aplikacije

- ✿ **Hibridne mobilne aplikacije** zasnovane na Web tehnologijama
 - ✦ Cordova (PhoneGap) – mobilne aplikacije razvijene korišćenjem JavaScript, HTML i CSS u okviru WebView-a
- ✿ **Kros-platformske** aplikacije zasnovane na interpretiranju JavaScript-a
 - ✦ React Native
 - ✦ NativeScript
 - ✦ Titanium (Appcelerator)
- ✿ **Kros-kompajlirane** mobilne aplikacije
 - ✦ Xamarin
 - ✦ RubyMotion

Hibridne mobilne aplikacije

- ✿ Razvijene korišćenjem HTML, CSS i JS i izvršavaju se u okviru WebView-a koji zauzima ceo ekran
- ✿ Preko **nativnog wrapper-a** pristupaju API-u platforme i karakteristikama uređaja/platforme

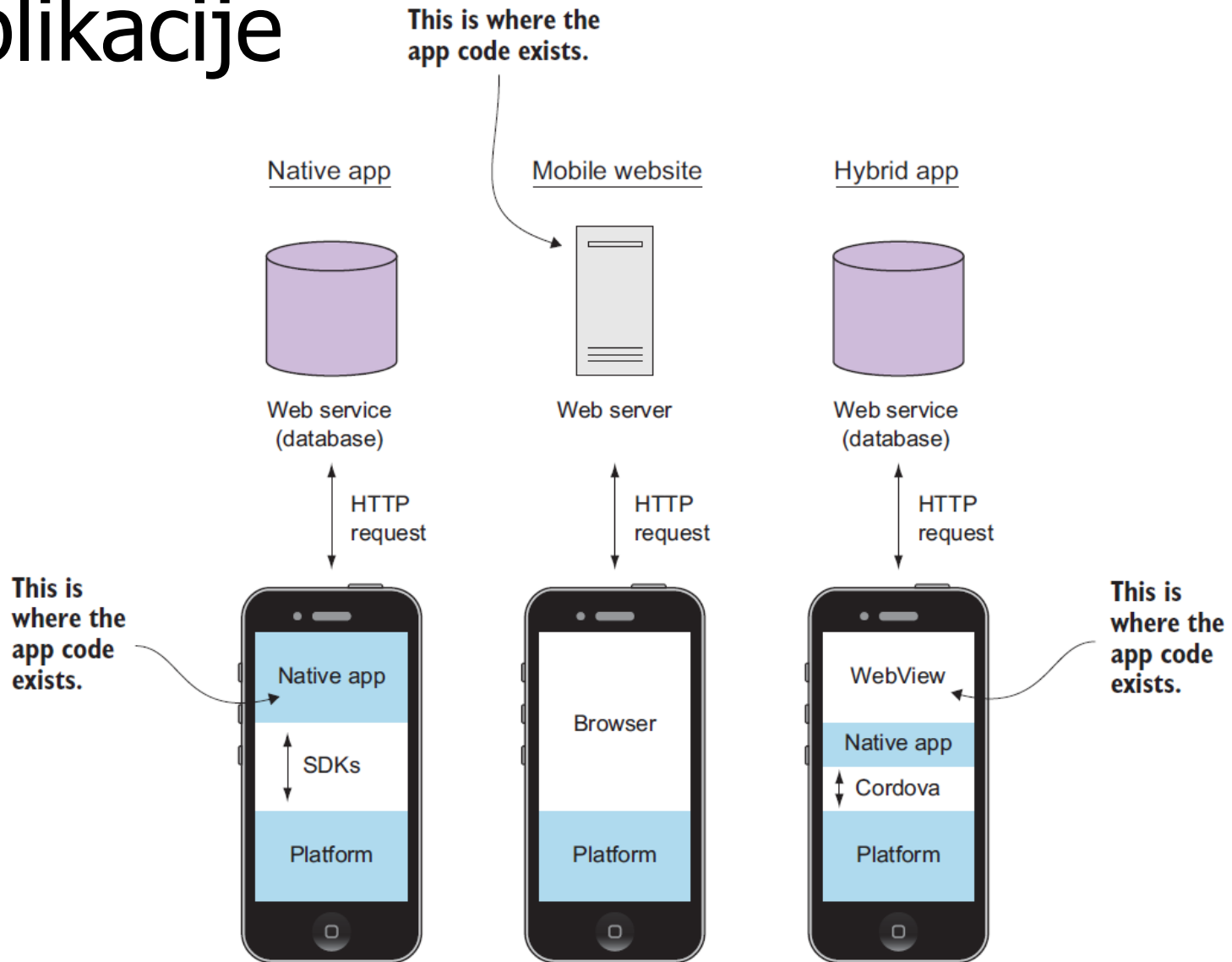


Arhitektura mobilnih aplikacija

Razvoj mobilnih aplikacija i servisa



Nativne, Web i hibridne mobilne aplikacije



Arhitektura mobilnih aplikacija

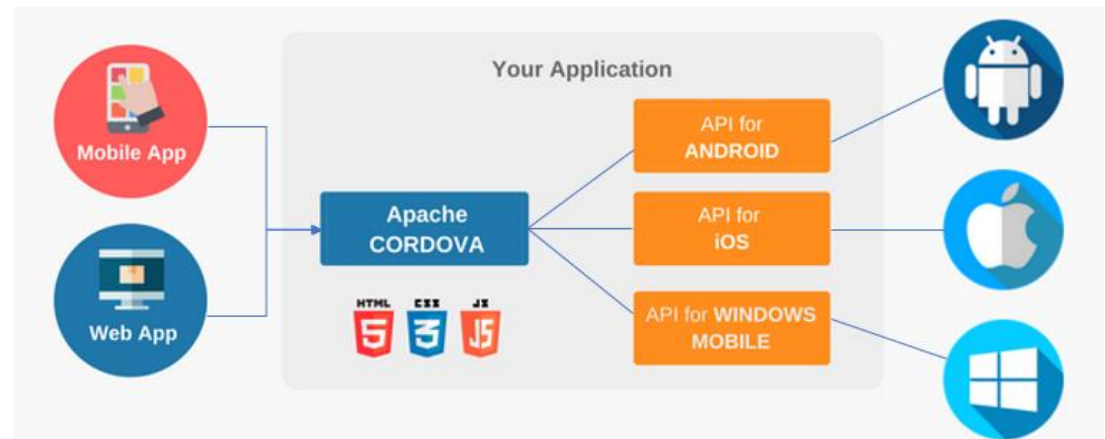
Razvoj mobilnih aplikacija i servisa

Hibridne mobilne aplikacije

- ❁ Obezbeđuju kros-platformski razvoj i izvršavanje
- ❁ Treba da omoguće native karakteristike i korisničko iskustvo uz korišćenje, uglavnom, Web tehnologija
 - ❑ Moguće je koristiti kros-kompajliranje ili interpretiranje/evaluaciju koda napisanog u određenom programskom jeziku, npr. C# ili JavaScript zajedno sa nativnim aplikacionim komponentama
- ❁ Hibridna mobilna aplikacija je mobilna Web aplikacija koja se izvršava u okviru Web view-a (*Web browser* kontrole) obuhvaćenog nativnom aplikacijom (aplikacionom ljuskom - *shell-om*)
- ❁ Predstavlja kompromis između visokih troškova razvoja posebne native aplikacije za svaku platformu i nemogućnosti pristupa karakteristikama platforme i mobilnog uređaja od strane mobilne Web aplikacije
- ❁ Mogućnost postavljanja aplikacije na App store i Google Play

Hibridne mobilne aplikacije

- ❁ WebView aplikacije – izvršavaju se u internom Web browser-u koji je obuhvaćen nativnom aplikacijom
- ❁ Hibridne mobilne platforme
 - ❁ Apache Cordova (Adobe PhoneGap) - <https://cordova.apache.org/>
 - ❁ Ionic (Angular/React/Vue & Apache Cordova) - <https://ionicframework.com/>
 - ❁ Capacitor - <https://capacitorjs.com/>
 - ❁ Meteor.js (<https://www.meteor.com/>)
 - ❁ Trigger.io



Arhitektura mobilnih aplikacija

Razvoj mobilnih aplikacija i servisa

Kros-platformske mobilne aplikacije

- ❁ Interpretirane ili kros-kompajlirane hibridne aplikacije koje se interpretiraju/kompajliraju u bytecode ili nativni kod za svaku od podržanih platformi
 - ❁ React Native (<https://reactnative.dev/>)
 - ❁ Flutter (<https://flutter.dev/>)
 - ❁ NativeScript (<https://nativescript.org/>)
 - ❁ Appcelerator Titanium (<https://www.appcelerator.com/>)
 - ❁ Embarcadero FireMonkey
(<https://www.embarcadero.com/products/rad-studio/fm-application-platform>)
 - ❁ Xamarin (<https://dotnet.microsoft.com/apps/xamarin>)
 - ❁ Kotlin Multiplatform (<https://www.jetbrains.com/kotlin-multiplatform/>)



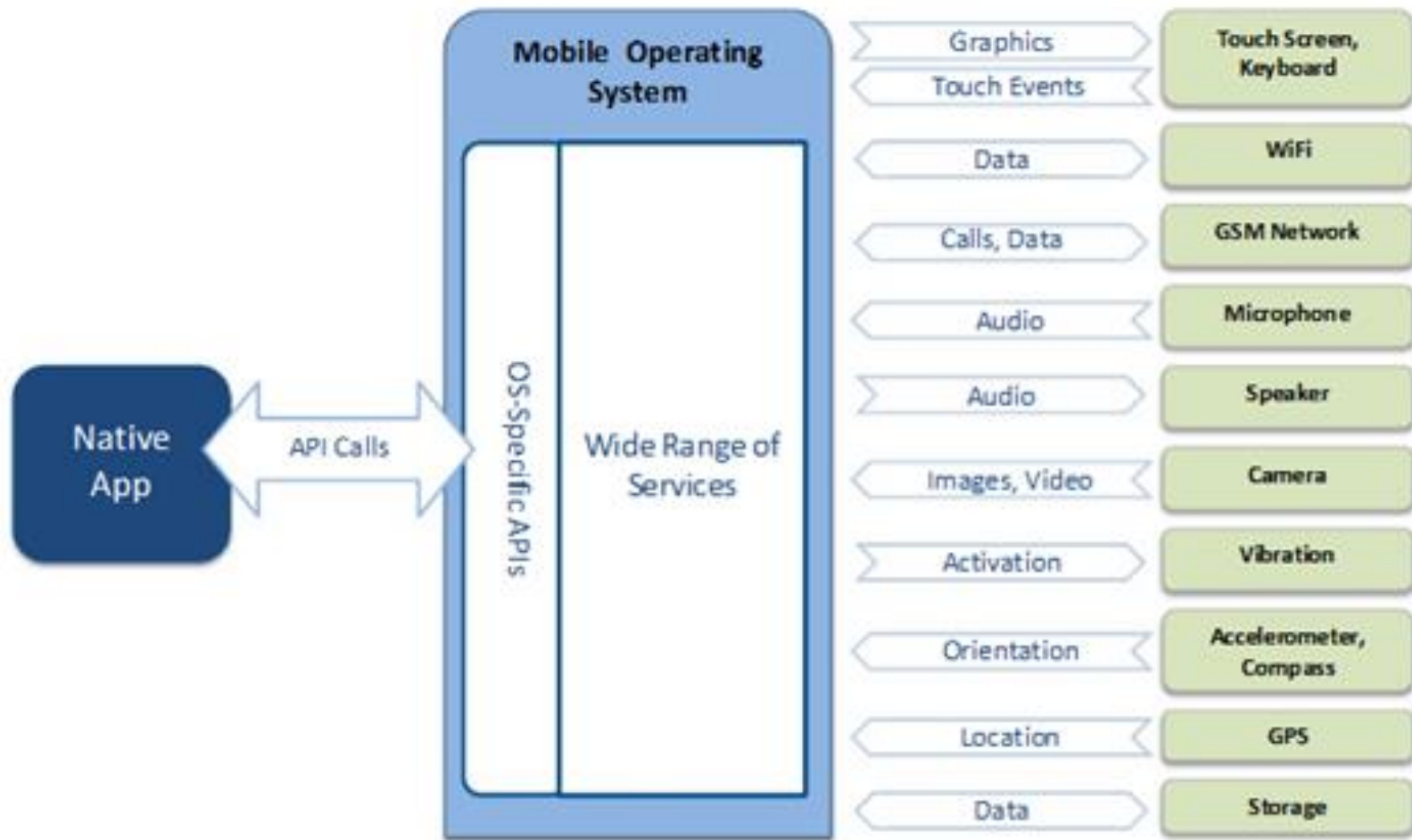
Kros-platformske mobilne aplikacije

- ❁ Kros-platformske mobilne aplikacije su odlična opcija ako:
 - ❁ Želite da vam se mobilna aplikacija izvršava na više mobilnih platformi
 - ❁ U aplikaciji pristupate mogućnostima i komponentama uređaja, poput GPS, akcelerometra ili kamere
 - ❁ Pristupate nativnom API-u
 - ❁ Želite da aplikacije bude upotrebljiva i bez bežične mrežne konekcije (*offline*)
 - ❁ Nemate potrebu za naprednim grafičkim performansama koje možete dobiti samo u nativnoj aplikaciji
 - ❁ Želite da aplikaciju objavite na *App store*

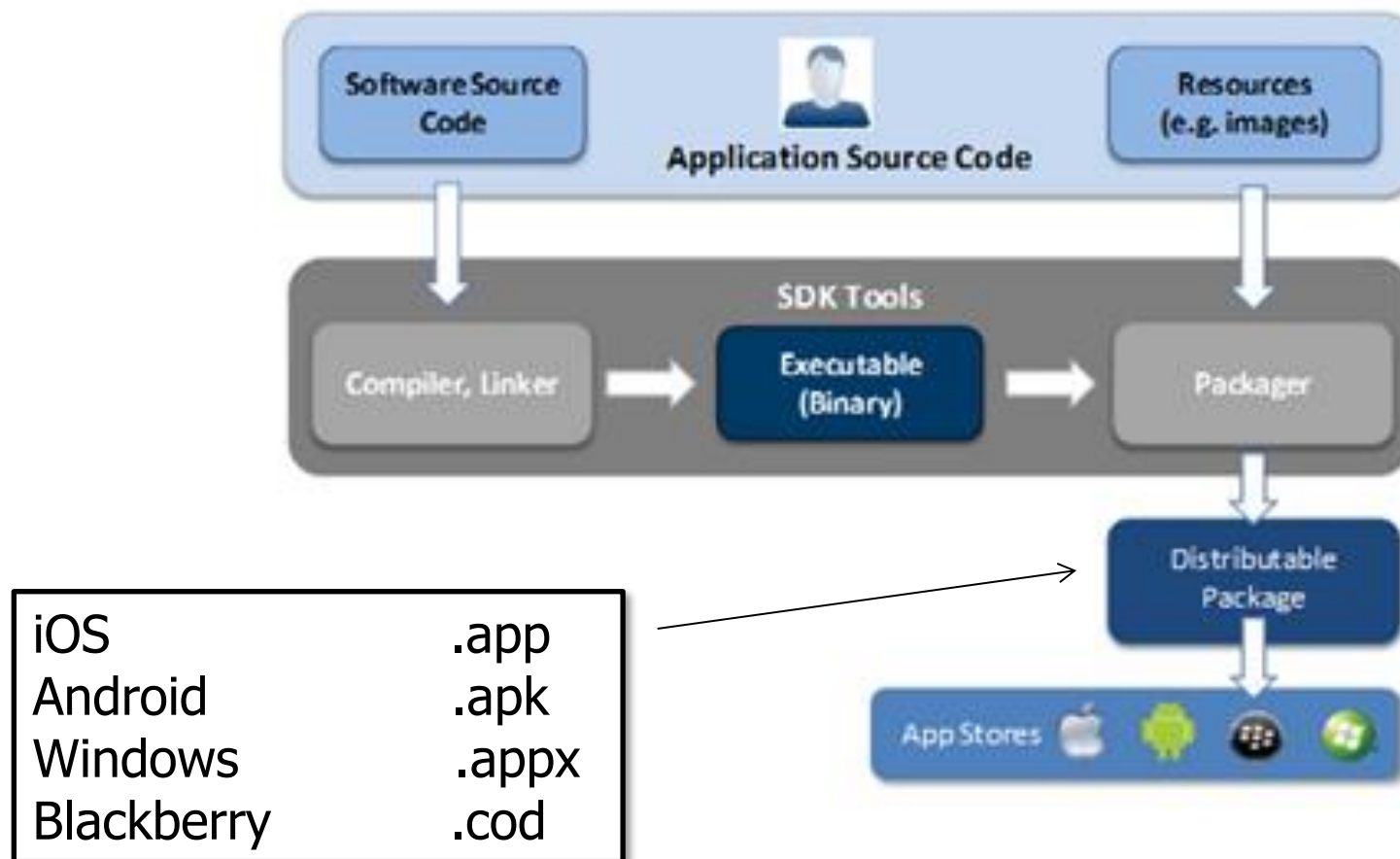


NATIVNE MOBILNE APLIKACIJE

Nativne mobilne aplikacije



Nativne mobilne aplikacije (2)



Nativne mobilne aplikacije

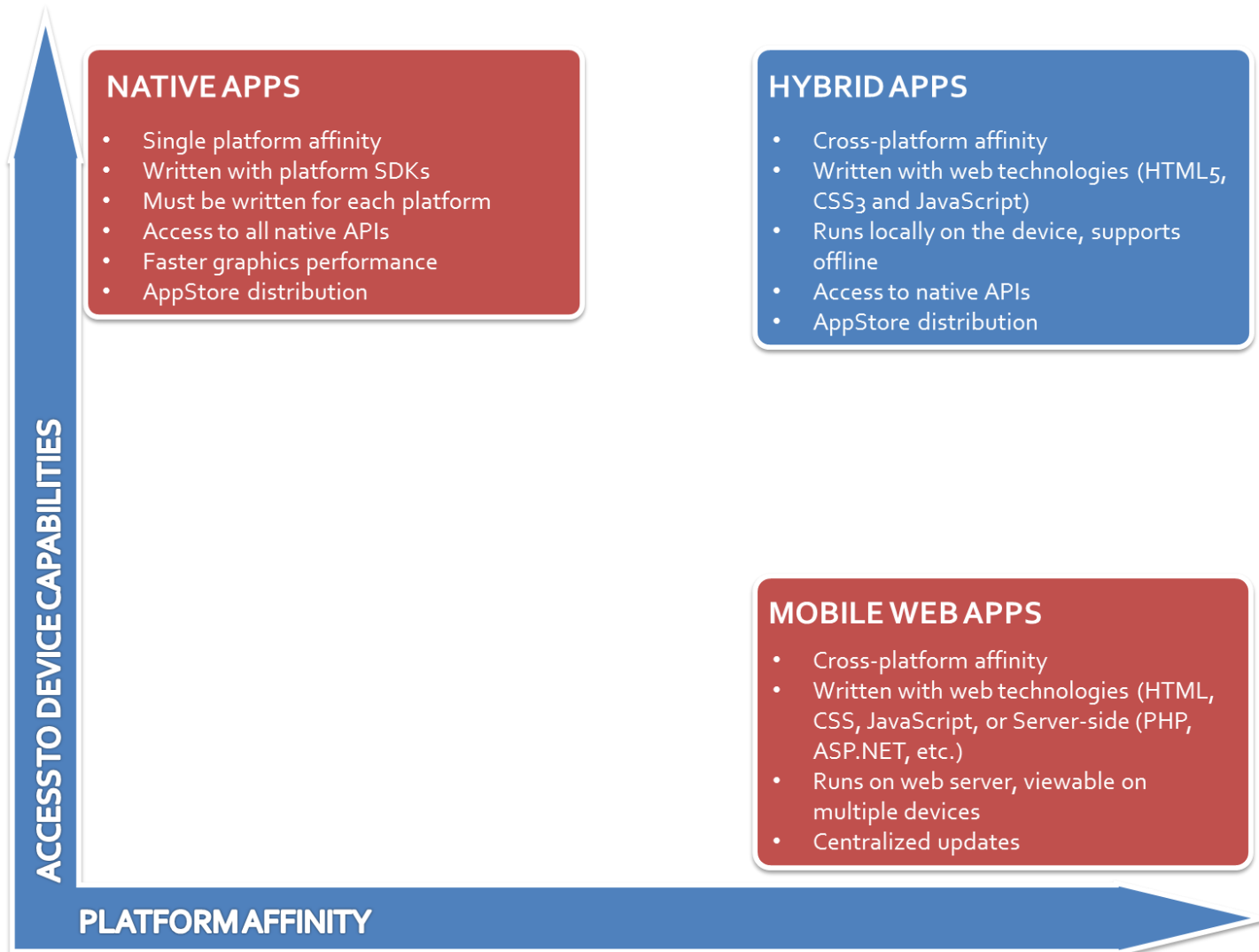
Prednosti

- ➊ Direktan pristup mobilnoj platformi preko nativnog API-a
- ➋ Visoke performanse
- ➌ Prilagođena platformi i njenoj evoluciji

Nedostaci

- ➊ Neophodno dobro poznavanje programskog (-ih) jezika i nativnog API-a
- ➋ Razvoj isključivo za jednu platformu
- ➌ Visoki troškovi i napor u razvoju aplikacije

Izbor tipa mobilne aplikacije



Literatura

- ✚ Maximiliano Firtman, *Programming the Mobile Web*, 2nd Edition, O' Reilly Media, 2013.
- ✚ *Mobile Developer's Guide To The Galaxy*, 18th Edition, 2019
- ✚ Breaking the Mobile Web
 - ✚ <http://www.mobilexweb.com/>
- ✚ Raymond K. Camden, *Apache Cordova in Action*, Manning Publications; 1 edition, 2015

Pitanja i komentari

