



Apps mit HTML5, JavaScript und CSS3

Apps -Die Werkstatt

Webserver (lokal oder extern)

Lokal testen:

→ XAMPP (Lokaler Webserver, der auf einem lokalen Rechner das Web simuliert)

Download unter <http://www.apachefriends.org/de/>



Extern testen:

→ Webspaces bei einem Provider nutzen und die Testdaten per **FTP-Programm** auf den Server übertragen.

→ FTP-Programme (z.B. FileZilla, Win-FTP) gibt es im Web kostenlos zum Download.



Apps -Die Werkstatt

Zugang für Tablets und Smartphones auf **lokalen XAMPP** innerhalb eines **WLAN**

1. **httpd-xampp.conf** im Verzeichnis **APACHE/CONF/EXTRA** suchen
2. In **httpd-xampp.conf** den Eintrag „**Allow from :::1 127.0.0.0/8**“ in „**Allow from all**“ ändern
3. Starten von **cmd.exe** in der Befehlszeile von Windows
4. Über den Befehl **ipconfig** die ip-Adresse Ihres lokalen Webserver herausfinden

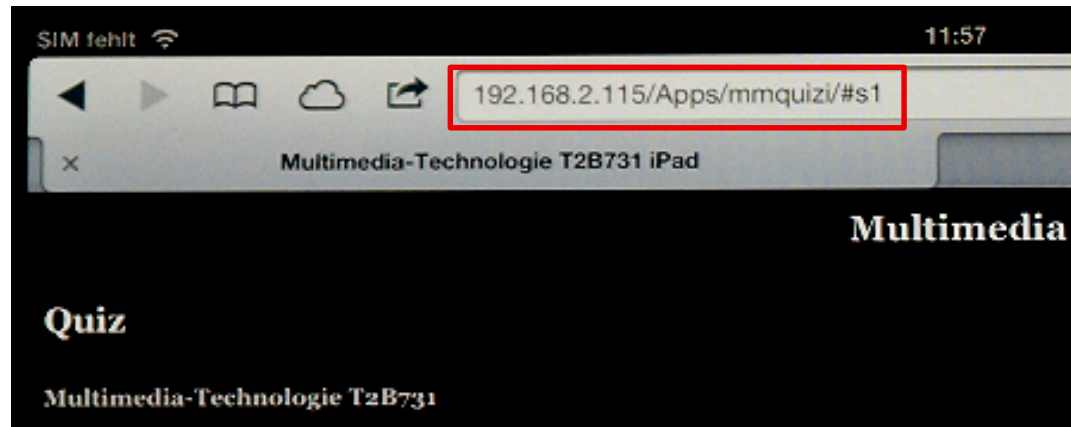
```
Windows-IP-Konfiguration

Ethernet-Adapter LAN-Verbindung:

    Verbindungsspezifisches DNS-Suffix: Speedport_W_723U_
    Verbindungsspezifische IPv6-Adresse : f588:15847:9c44::
    IPv4-Adresse . . . . . : 192.168.2.115
    Subnetzmaske . . . . . : 255.255.255.0
    Standardgateway . . . . . : 192.168.2.1
```

Apps -Die Werkstatt

5. Aufruf der (Web-)App vom lokalen Webserver z.B.: `192.168.2.115/Apps/mmquizzi`



Apps -Die Werkstatt

Im Web testen: → Webserver bei einem Provider mieten(Externer Webserver)

WebApp mit URL aufrufen z.B.:

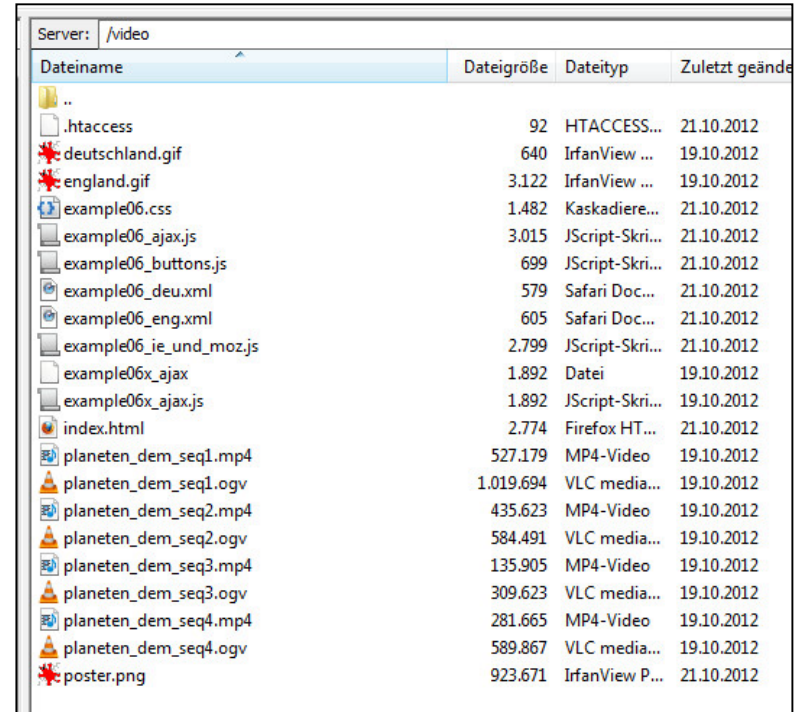
www.technischeredaktion.com/video

Verzeichnis auf dem Webserver:

Beim Einsatz von Videos muss sich eine **.htaccess**-Datei mit den folgenden Einträgen im Verzeichnis befinden:

```
#MP4 video mimetype
```

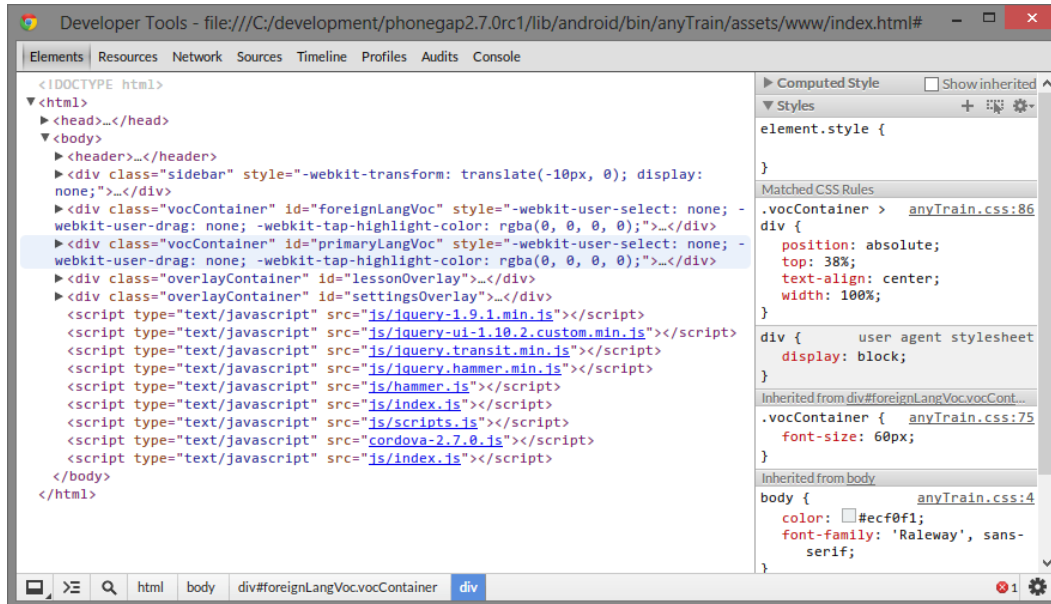
```
AddType video/mp4 .mp4
```



Dateiname	Dateigröße	Dateityp	Zuletzt geändert
..			
.htaccess	92	HTACCESS...	21.10.2012
deutschland.gif	640	IrfanView ...	19.10.2012
england.gif	3.122	IrfanView ...	19.10.2012
example06.css	1.482	Kaskadiere...	21.10.2012
example06_ajax.js	3.015	JScript-Skri...	21.10.2012
example06_buttons.js	699	JScript-Skri...	21.10.2012
example06_deu.xml	579	Safari Doc...	21.10.2012
example06_eng.xml	605	Safari Doc...	21.10.2012
example06_ie_und_moz.js	2.799	JScript-Skri...	21.10.2012
example06x_ajax	1.892	Datei	19.10.2012
example06x_ajax.js	1.892	JScript-Skri...	19.10.2012
index.html	2.774	Firefox HT...	21.10.2012
planeten_dem_seq1.mp4	527.179	MP4-Video	19.10.2012
planeten_dem_seq1.ogv	1.019.694	VLC media...	19.10.2012
planeten_dem_seq2.mp4	435.623	MP4-Video	19.10.2012
planeten_dem_seq2.ogv	584.491	VLC media...	19.10.2012
planeten_dem_seq3.mp4	135.905	MP4-Video	19.10.2012
planeten_dem_seq3.ogv	309.623	VLC media...	19.10.2012
planeten_dem_seq4.mp4	281.665	MP4-Video	19.10.2012
planeten_dem_seq4.ogv	589.867	VLC media...	19.10.2012
poster.png	923.671	IrfanView P...	21.10.2012

Apps -Die Werkstatt

→ Chrome Developer Tools oder Firebug vereinfachen die App-Entwicklung ungemein



Mobile Dokumentationserstellung

Die „Mobile Dokumentationserstellung“ kann in zwei grundsätzliche Vorgehensweisen unterteilt werden. Die erste Vorgehensweise ist die Erstellung von Apps, mit denen die Dokumentation zur Verfügung gestellt wird. Die zweite Vorgehensweise ist die Erstellung und Bereitstellung von Dokumentation durch Anwendungen wie Content-Management-Systeme oder Editoren. In diesem Fall werden die Informationen als Websites, im ePub-Format oder als PDF-Dateien bereit gestellt. Die für mobile Geräte optimale Bereitstellung von Informationen durch Content-Management-Systeme wird durch den Einsatz von CSS3 und Templates, die das typische Design für mobile Geräte zur Verfügung stellen, erleichtert.

Die Erstellung von Mobiler Dokumentation in Apps hängt davon ab, welche **Klasse von Apps** erstellt werden soll. Hierbei wird zwischen den drei Varianten **Native App**, **WebApp** und **Hybride App** unterschieden.

Je nach App-Klasse gibt es einen unterschiedlichen Erstellungsprozess und unterschiedliche Gerätenutzungsmöglichkeiten. Die App-Klassen wirken sich stark auf die Erstellungskosten aus. Bei nativen Apps sind die Programmierkosten höher als bei WebApps, außerdem sind eventuell Varianten für die unterschiedlichen Betriebssysteme (iOS, Android, Windows etc...) zu erstellen und zu pflegen.

Bei WebApps und Hybriden Apps hingegen sind die technischen Möglichkeiten bei der Gerätenutzung geringer als bei Native Apps. Sie sind jedoch im Erstellungsprozess und im Pflegeprozess kostengünstiger.

Native Apps

Native Apps werden mit speziellen Software Development Kits (SDK), abhängig vom Gerätebetriebssystem (z.B. Android oder iOS), erstellt.

Es ist notwendig, die Anwendungen in der geeigneten Programmiersprache zu erstellen, um die technischen Möglichkeiten der Geräte bestmöglich auszunutzen.

- Im Falle des Betriebssystems **Android** ist die Programmiersprache **Java**,
- im Falle des Betriebssystems **iOS** von Apple ist die Programmiersprache **Objective-C** und **Swift**
- im Falle des Betriebssystems **Windows 8** ist es **C#**.

Der Vorteil dieser Technologien ist, dass diese Anwendungen **deutlich performanter** sind als WebApps. So ist es möglich, flüssig laufende 3D-Programme für mobile Geräte zur Verfügung zu stellen oder **gleichzeitig mehrere Apps** zu nutzen, **ohne die Batterie schnell zu entleeren** oder die Hardware zu überfordern. Ein zusätzlicher Vorteil ist die **Vermarktungsmöglichkeit der Native Apps über die App Stores**.

Web-Apps

WebApps können mit HTML5, CSS3 und JavaScript hergestellt werden.

Eine WebApp ist eigentlich nur eine Website, die sich fast wie eine native Anwendung verhält und bedienen lässt. Um die Interaktivität einer WebApp fast wie die einer Native App erscheinen zu lassen, ist es erforderlich, mit JavaScript und/oder JavaScript-Frameworks wie z.B. jQuery Mobile oder Sencha Touch zu arbeiten.

Durch diese Technologien ist es möglich, eine Website wie ein Programm reagieren zu lassen. Die Seite wird je nach Bedarf und Eingaben geändert oder passt sich an, ohne neu geladen zu werden. Die Seite wird zur Laufzeit dynamisch geändert und wirkt dadurch wie ein Programm.

Ein Vorteil von WebApps ist, dass sie nicht auf den Endgeräten installiert werden müssen. Sie werden über den Browser der Endgeräte aufgerufen und ausgeführt. Sind die WebApps gut gemacht, dann bleibt der Browser dabei unsichtbar und das Design lässt die WebApp fast wie eine Native App erscheinen.

Ein weiterer Vorteil ist, dass Updates nicht nachinstalliert werden müssen, denn der Browser lädt ja immer die aktuellsten Daten vom Server. Der größte Vorteil jedoch ist, dass WebApps geräteunabhängig und betriebssystemunabhängig sind und es somit nicht mehr notwendig ist, mehrere Varianten der Anwendung bereitzuhalten, was im Bereich der Technischen Dokumentation sehr wichtig ist.

Hybride-Apps

Mit der Technologie der Hybride Apps sollen die Vorteile von Native Apps und WebApps genutzt und die Nachteile beider Technologien umgangen werden. Der Zugriff auf die Hardware eines Smartphones soll wie bei der Native App möglich sein und die Anwendung soll wie eine WebApp nicht von einer Plattform abhängen.

Erstellt wird die Hybride App, indem eine WebApp mit Web-Technologien entwickelt wird und dann – je nach der gewünschter Zielplattform – in eine Native App transformiert wird. Da die entstandene Anwendung als nativ bezeichnet werden kann, ist es möglich, sie in einem App-Store zu vertreiben und für die Ausführung auf dem Smartphone zu installieren.

Es ist nicht nur möglich, mit Web-Technologien zu arbeiten. Es gibt sogar Lösungen, die das Entwickeln mit C, C++, Java oder einer .NET-Programmiersprache erlauben. Ein Werkzeug, das aus WebApps Hybride Apps erstellen kann, ist PhoneGap, das erst kürzlich von Adobe gekauft wurde, jedoch nach wie vor kostenfrei zur Verfügung steht. Weitere Werkzeuge dieser Art sind Titanium oder Sencha Touch.

Desktop vs. Mobile - Gestaltungsunterschiede

Auflösungen:

Alle modernen iPhones und iPods mit Retina-Display: 960 x 640 Pixel

iPads der dritten Generation 2048 x 1536 Pixel

Geräte haben zwei verschiedene Ausrichtungen „Landscape“ und „Portrait“.

	Low Density	Medium Density	High Density	Extra High Density
Small Screen	240 × 320	–	480 × 640	–
Normal Screen	240 × 400 240 × 432	320 × 480	480 × 800 480 × 854 600 × 1024	640 × 960
Large Screen	480 × 800 480 × 854	480 × 800 480 × 854 800 × 1024	–	–
Extra Large Screen	1024 × 600	1280 × 800 1024 × 768 1280 × 768	1536 × 1152 1920 × 1152 1920 × 1200	2048 × 1536 2560 × 1536 2560 × 1600

Exkurs Bildschirme

Bei Tablets oder Smartphones ist immer ein farbiger, kratzfester LCD- oder AMOLED-Touchscreen.

Touchscreen-Aufbau

Berührungsempfindliche Oberfläche: Die sitzt direkt hinter dem Schutzglas und besteht aus mehreren leitfähigen Schichten. Wie das funktioniert, erfahren Sie im Artikel: „[Touchscreens: Anfassen erwünscht](#)“.

Bildschirm: Der besteht aus Tausenden Bildpunkten, die – je nach Technik – zwei oder drei „Subpixel“ in den Grundfarben Rot, Blau, Grün enthalten.

Welche Bildschirmtechniken gibt es?

LCD-Displays sind herkömmliche Flachbildschirme (TFT-LCDs), wie sie auch in PC-Monitoren, Notebooks oder Fernsehern zu finden sind. Damit arbeiten alle Tablets und viele Smartphones.

AMOLEDs stecken bisher vor allem in [Geräten](#) mit kleinen Displays, etwa Smartphones oder mobilen Musik- und Videoabspielern. LG und Samsung bringen AMOLED-TV-Geräte mit 55-Zoll-Bildschirm auf den Markt, auch AMOLED-Tablets und Notebooks sind denkbar.

<http://www.computerbild.de/artikel/cb-Ratgeber-Kurse-PC-Hardware-Wissen-Displays-LCD-AMOLED-Touchscreen-Retina-7363166.html>

CSS für unterschiedliche Geräte

Responsive Design

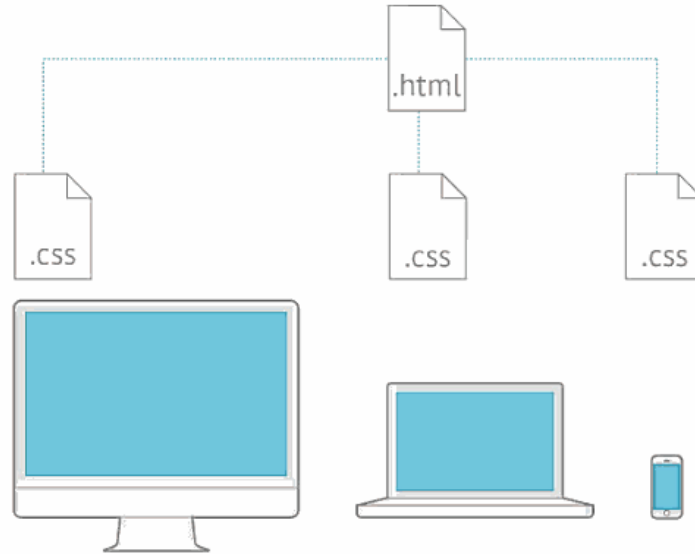


Abbildung 10: Alternative CSS Files für verschiedene Ausgabegeräte. Der Inhalt einer HTML Datei wird durch unterschiedliche Formatierung in unterschiedlichen CSS Files an das Display des konkreten Ausgabegeräts angepasst.

Think CROSS - Change MEDIA: Eine Standortbestimmung im Jahr 2012 von Christine Michitsch

CSS3 – Media Queries

Media Queries ermöglichen es, Definitionen für bestimmte Bildschirmgrößen festzulegen. Damit ist es möglich mit der selben Datei die Webseite für ein Smartphone und ein Tablett anzupassen.

Media Queries können direkt in der CSS-Datei eingetragen werden z.B.:

```
@media screen and (min-device-width: 768px) {  
    #wrapper {  
        width: 960px;  
        height: 600px;  
    }  
}
```

Diese Wrapper-Größe wird ab einer bestimmten Bildschirmgröße (iPad) verwendet.

CSS3 – Media Queries

Bildschirmbereiche zwischen 300px und 800px angeben.

```
@media screen and (min-width: 300px) and (max-width: 800px) {  
/* Hier sollte die Definition stehen, die nur für einen bestimmten Bereich gelten  
soll. */  
}
```

Angabe ob ein Gerät im Hochformat oder im Querformat gehalten wird.

```
@media screen and (min-device-width : 768px) and (orientation : landscape) {}
```

```
@media screen and (min-device-width : 768px) and (orientation : portrait) {}
```

[Tipp! http://www.intensivstation.ch/css3/media-queries/](http://www.intensivstation.ch/css3/media-queries/)

CSS3 – Media Queries als Link im HTML

```
<link rel="stylesheet" type="text/css" href="styles_1.css"  
      media="screen and (min-width: 350px)" />
```

```
<link rel="stylesheet" type="text/css" href="styles_2.css"  
      media="screen and (min-width: 750px)" />
```


App-Icons für die verschiedenen Auflösungen

iPhone 2, 3: 57 x 57 px

iPhone 4/4S: 114 x 114 px

iPad: 72 x 72 px

iPad (Retina) 144 x 144 px

```
<head>
```

```
<link href="styles/styles_smart.css" rel="stylesheet" type="text/css" />
```

```
<link href="styles/dek_screen.css" rel="stylesheet" type="text/css" />
```

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0",  
maximum-scale=1.0, user-scalable=0">
```

```
<meta name="apple-mobile-web-app-capable" content="yes" />
```

```
<link rel="apple-touch-icon" href="bilder/icon-iphone72.png" sizes="72x72"/>
```

```
<link rel="apple-touch-icon" href="bilder/icon-iphone114.png" sizes="114x114"/>
```

```
<link rel="apple-touch-icon" href="bilder/icon-iphone144.png" sizes="144x144"/>
```

```
<link rel="apple-touch-icon" href="bilder/icon-iphone.png" />
```

```
<title>App-Icon-Übung</title>
```

```
<meta name="apple-mobile-web-app-status-bar-style" content="black-translucent" />
```

```
</head>
```

Viewport für „mobile“ Websites anpassen

```
<meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=no">
```

initial-scale:

Der Wert bestimmt den Zoomgrad. 1.0 bedeutet, dass die Inhalte 1:1 dargestellt werden. Auf einem Screen mit 320px Breite füllt eine 320px-breite Grafik die ganze Breite aus. 2.0 erzeugt eine zweifache Vergrößerung.

user-scalable:

Das Attribut definiert, ob man auf der Seite zoomen kann (yes) oder nicht (no).

minimum-scale und maximum-scale:

Einschränkung des Zoomgrads. Wird die maximale Skalierung auf 2.0 gesetzt, kann der Inhalt maximal 2x-fach vergrößert werden.



Adobe Cordova auf Windows installieren

Um Android-Apps zu entwickeln sind einige Technologien auf dem Rechner zu installieren:

- **Node.js**
- **Git**
- **Cordova**
- **Java**
- **Ant**
- **Android SDK Tools**



Adobe Cordova auf Windows installieren

Node.js installieren

Das Installationsprogramm erhalten Sie hier: <http://nodejs.org>

Das Installationsprogramm ausführen. Node.js sollte automatisch in die PATH-Umgebungsvariable von Windows eingetragen werden.

Um die Installation zu testen öffnen Sie mit dem Befehl cmd.exe.

Gehen Sie in das Startmenü. In Windows 7 finden Sie direkt unter dem Menüpunkt "Alle Programme" eine Suchmaske. Über dieser suchen Sie nach der "cmd.exe".

Geben Sie **node --version** ein. Jetzt sollte die Versionsnummer angezeigt werden.

Adobe Cordova auf Windows installieren --> PATH-Umgebungsvariable einrichten

The image shows two overlapping Windows windows. The background window is 'Systemeigenschaften' (System Properties) with the 'System' tab selected. The foreground window is 'Umgebungsvariablen' (Environment Variables) for user 'scma0005'. In the 'Umgebungsvariablen' window, the 'PATH' variable is highlighted in red, showing the path 'c:\Program Files\nodejs\npm;C:\Program Files\Java\jdk1.8.0_11...'. The 'JAVA_HOME' variable is also highlighted in red, showing the path 'C:\Program Files\Java\jdk1.8.0_111'. The 'Systemvariablen' section shows 'ANT_HOME', 'ComSpec', 'GTK_BASEPATH', 'NUMBER_OF_PROCESSORS', 'OS', 'OU', and 'Path'.

Systemsteuerung > System und Sicherheit > System

Systemeigenschaften

Computername Hardware Erweitert Computerschutz Remote

Sie müssen als Administrator angemeldet sein, um diese Änderungen durchführen zu können.

Leistung

Visuelle Effekte, Prozessorzeitplanung, Speichernutzung und virtueller Speicher

Benutzerprofile

Desktopeinstellungen bezüglich der Anmeldung

Starten und Wiederherstellen

Systemstart, Systemfehler und Debuginformationen

Umgebungsvariablen...

OK Abbrechen Übernehmen

Computerbeschreibung:

Domäne: ads.hs-karlsruhe.de

Windows-Aktivierung

Umgebungsvariablen

Benutzervariablen für scma0005

Variable	Wert
OneDrive	C:\Users\scma0005\OneDrive
PATH	c:\Program Files\nodejs\npm;C:\Program Files\Java\jdk1.8.0_11...
TEMP	%USERPROFILE%\AppData\Local\Temp
TMP	%USERPROFILE%\AppData\Local\Temp
Workdir	C:\Users\scma0005\Documents

Neu... Bearbeiten... Löschen

Systemvariablen

Variable	Wert
ANT_HOME	C:\ant
ComSpec	C:\windows\system32\cmd.exe
GTK_BASEPATH	C:\Program Files (x86)\GtkSharp\2.12\
JAVA_HOME	C:\Program Files\Java\jdk1.8.0_111
NUMBER_OF_PROCESSORS	4
OS	Windows_NT
OU	IZ\ITS
Path	C:\ProgramData\Oracle\Java\javapath;C:\windows\system32;C:\...

Neu... Bearbeiten... Löschen



Adobe Cordova auf Windows installieren

- GIT -

Installieren Sie das **Versionierungssystem Git**. Diese Software wird von Cordova im Hintergrund genutzt.

<http://git-scm.com>



Adobe Cordova auf Windows installieren

- cordova -

Installieren Sie **Cordova**.

Cordova wird mit dem Node Package Manager (npm) installiert.

Geben Sie den folgenden Befehl im Kommandozeilenfenster (cmd.exe) ein:

```
npm install -g cordova
```

Testen Sie die Cordova-Installation mit dem Befehl:

```
cordova --version
```

Wenn Sie eine Versionsnummer sehen, war die Installation erfolgreich.



Adobe Cordova auf Windows installieren

- Java -

Java Installation

Der Android SDK benötigt den **Java Development Kit (JDK)** mindestens in der Version 1.6.

Prüfen ob Sie den JDK schon installiert haben mit `javac -version`

Wenn Sie noch keinen JDK installiert haben, gehen sie wie folg vor:

1. Download des Java SE JDK (SE = Standard Edition) von Oracle: www.oracle.com/technetwork/java/javase/downloads/.
"Windows x86" bei 32-bit Windows,
"Windows x64" bei 64-bit Windows.
1. Führen Sie die Installationsdatei aus. Merken Sie sich das Verzeichnis in das Sie den JDK installieren. Sie müssen das Verzeichnis in einem späteren Schritt in den PATH eintragen.
2. Passen Sie nun Ihren PATH an. Öffnen Sie den **Control Panel**, klicken Sie **System and Security**, klicken Sie **System**, Klicken Sie **Change settings**, welches das Systemeigenschaftenfenster öffnet. Wählen Sie den **Advanced** tab, dann klicken Sie auf die **Environment Variables** Schaltfläche.
3. In der Liste **User variables** wählen Sie **PATH** und klicken Sie die **Edit** Schaltfläche.

Adobe Cordova auf Windows installieren --> PATH-Umgebungsvariable einrichten

The image shows two overlapping windows from the Windows operating system. The background window is 'Systemeigenschaften' (System Properties) with the 'System' tab selected. The foreground window is 'Umgebungsvariablen' (Environment Variables) for the user 'scma0005'.

In the 'Systemeigenschaften' window, the 'System' tab is active, and the 'Umgebungsvariablen...' button is highlighted with a red box. The breadcrumb path at the top is 'Systemsteuerung > System und Sicherheit > System', also highlighted with a red box.

The 'Umgebungsvariablen' window displays two tables of environment variables:

Variable	Wert
OneDrive	C:\Users\scma0005\OneDrive
PATH	c:\Program Files\nodejs\npm;C:\Program Files\Java\jdk1.8.0_11...
TEMP	%USERPROFILE%\AppData\Local\Temp
TMP	%USERPROFILE%\AppData\Local\Temp
Workdir	C:\Users\scma0005\Documents

The 'PATH' variable is highlighted with a red box. Below the table are buttons for 'Neu...', 'Bearbeiten...', and 'Löschen'.

Variable	Wert
ANT_HOME	C:\ant
ComSpec	C:\windows\system32\cmd.exe
GTK_BASEPATH	C:\Program Files (x86)\GtkSharp\2.12\
JAVA_HOME	C:\Program Files\Java\jdk1.8.0_111
NUMBER_OF_PROCESSORS	4
OS	Windows_NT
OU	IZ\ITS
Path	C:\ProgramData\Oracle\Java\javapath;C:\windows\system32;C:\...

The 'JAVA_HOME' variable is highlighted with a red box. Below the table are buttons for 'Neu...', 'Bearbeiten...', and 'Löschen'.



Adobe Cordova auf Windows installieren

- Java -

Einrichten der **Variable value**: Trennen Sie den neuen Eintrag durch ein Semicolon gefolgt vom Pfad zum bin-Verzeichnis des JDK.

`;C:\Program Files\Java\jdk1.8.0_11\bin` Danach mit **OK** bestätigen.

Die **JAVA_HOME** Variable einrichten, wenn sie nicht schon existiert.

Wenn Sie noch nicht existiert klicken Sie die **Neu**-Schltfläche.

Variable Name: JAVA_HOME

Variable Wert: C:\Program Files\Java\jdk1.8.0_11Click the **OK** button.

Danach mit **OK** bestätigen.

Schließen Sie nun alle Fenster und öffnen Sie das cmd-Fenster und testen Sie die Einrichtung mit:

`javac -version`

Wenn die Versionsnummer angezeigt wird, waren Sie erfolgreich!



Adobe Cordova auf Windows installieren

- ANT -

Installation von Ant

Apache Ant ist ein Build-System für Java, das von Cordova und dem Android SDK verwendet wird. Gehen Sie folgendermaßen vor, um Ant zu installieren:

Laden Sie Ant von hier herunter: ant.apache.org/bindownload.cgi. Holen Sie sich den Zip-Download auf der Seite. Klicken Sie auf die Zip-Datei-Verknüpfung für die neueste Version, z. Apache-ant-1.9.4-bin.zip, und speichern Sie die Datei auf Ihrem Computer.

Entpacken Sie die ZIP-Datei in das Verzeichnis auf Ihrem Computer, auf dem Ant installiert werden soll. Sie können jedes Verzeichnis für die Installation auswählen. In diesem Leitfaden verwenden wir dies als Beispiel:

C: \Users\miki\ant

Beachten Sie, dass die Dateien im Ant-Paket direkt in dieses Verzeichnis gehen sollten. Notieren Sie sich das Verzeichnis, da Sie es zum PATH hinzufügen müssen.

Um Ant zum PATH hinzuzufügen, öffnen Sie die Systemsteuerung, klicken Sie auf System und Sicherheit, klicken Sie auf System, klicken Sie auf Einstellungen ändern, klicken Sie auf die Registerkarte Erweitert und dann auf die Schaltfläche Umgebungsvariablen.

In der Liste Benutzervariablen wählen Sie PATH und klicken Sie auf die Schaltfläche Bearbeiten.



Adobe Cordova auf Windows installieren

- ANT -

Am Ende des Feldes Variablenwert fügen Sie ein Semikolon hinzu, gefolgt vom Pfad zum bin-Verzeichnis der Ant-Installation. Hier ist ein Beispiel:

```
;C:\Users\miki\ant\bin
```

Klicken Sie auf die Schaltfläche OK.

Als Nächstes fügen Sie die Variable ANT_HOME hinzu. Klicken Sie auf die Schaltfläche Neu. Im Feld Variablentyp:

ANT_HOME

Geben Sie im Feld Variablenwert den Pfad zum Verzeichnis ein, in dem Ant installiert ist, ohne das Semikolon und das Unterverzeichnis / bin:

```
C:\Users\miki\ant
```

Klicken Sie auf die Schaltfläche OK.

Klicken Sie erneut auf die Schaltfläche OK, um das Fenster Umgebungsvariablen zu schließen.

Testen Sie jetzt die Installation. Schließen Sie alle geöffneten Befehlsfenster, und öffnen Sie ein neues Befehlsfenster und geben Sie Folgendes ein:

```
ant -version
```

Wenn Sie eine Versionsnummer sehen, haben Sie Ant erfolgreich installiert!



Adobe Cordova auf Windows installieren

- Android SDK -

Installieren Sie die Android SDK Tools

Die SDK-Tools für Android werden von Cordova verwendet, um Android-Apps zu bauen.

SDK Tool installieren:

Gehen Sie auf die Seite developer.android.com/sdk scrollen Sie die Seite und klicken Sie auf "ALLE DOWNLOADEN UND GRÖSSEN ANZEIGEN". Unter "SDK-Tools", klicken Sie auf die Windows-Installationsprogramm exe-Datei und laden Sie es (diese Datei heißt z. B.: installer_r23.0.2-windows.exe).

Führen Sie das Installationsprogramm aus, nach dem Herunterladen aus. Es empfiehlt sich, die vom Installer verwendeten Standardeinstellungen zu verwenden. Notieren Sie sich das Verzeichnis, in dem das SDK installiert ist, da Sie es im nächsten Schritt zum PATH hinzufügen müssen.

Um die SDK-Tools dem PATH hinzuzufügen, öffnen Sie die Systemsteuerung, klicken Sie auf System und Sicherheit, klicken Sie auf System, klicken Sie auf Einstellungen ändern, klicken Sie auf die Registerkarte Erweitert und dann auf die Schaltfläche Umgebungsvariablen.

In der Liste Benutzervariablen wählen Sie PATH und klicken Sie auf die Schaltfläche Bearbeiten.

Am Ende des Feldes Variablenwert fügen Sie ein Semikolon hinzu, gefolgt von dem Pfad zu den Tools und den Plattform-Tools-Direktiven der Android SDK-Installation. Hier ist ein Beispiel für das Hinzufügen (beachten Sie, dass es zwei Pfade in einer Zeile gibt, die durch ein Semikolon getrennt sind):

```
;C:\Users\miki\AppData\Local\Android\android-  
sdk\tools;C:\Users\miki\AppData\Local\Android\android-sdk\platform-tools
```



Adobe Cordova auf Windows installieren

- Android SDK -

Testen Sie jetzt die Installation.

adb -version

Dies sollte die Version der Android Debug Bridge anzeigen.

Als letzten Schritt müssen Sie die spezifische Android-SDK-Version von Cordova einrichten. Dies geschieht durch Ausführen des Android SDK Managers durch Eingabe des Befehls:

Android

Dies startet ein Fenster, in dem Sie spezifische Android SDKs installieren können.

Beim ersten Start des Android SDK-Managers gibt es voreingestellte Auswahlmöglichkeiten. Es wird empfohlen, diese zu übernehmen.

Beachten Sie, dass das von Cordova benötigte Android-SDK sich in Zukunft ändern wird, da neue Versionen von Cordova und Android veröffentlicht werden. Öffnen Sie in diesem Fall den Android SDK-Manager erneut und installieren Sie die erforderliche API-Version(en).



Cordova auf Windows einrichten und ausführen

Cordova installieren

Die Cordova-Befehlszeile läuft auf Node.js und ist auf NPM verfügbar.

Öffnen Sie eine Eingabeaufforderung oder Terminal (cmd.exe), und geben Sie ein

```
npm install -g cordova
```



Cordova auf Windows einrichten und ausführen

Erstellen Sie ein Projekt

Erstellen Sie ein leeres Cordova-Projekt mit dem Befehlszeilentool. Navigieren Sie zu dem Verzeichnis, in dem Sie Ihr Projekt erstellen und tippen möchten

```
cordova create <path>
```

Um alle Optionen zu sehen, geben Sie

```
cordova help create
```

ein.

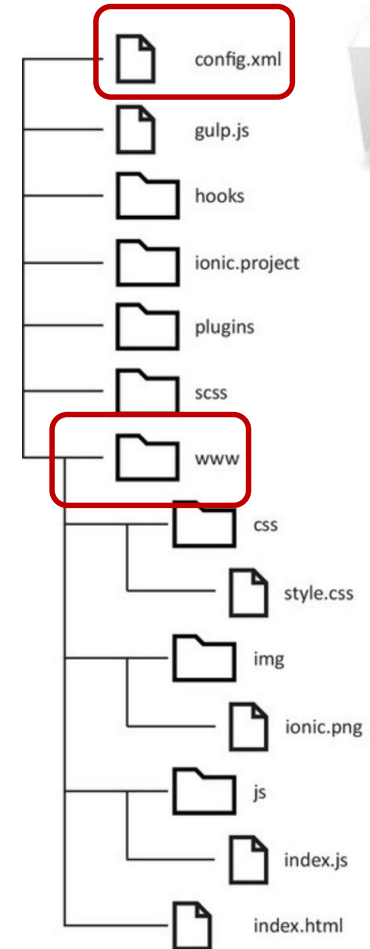


Cordova auf Windows einrichten und ausführen

Jetzt befindet sich unter Ihrem angelegten App-Verzeichnis folgende Struktur:

Die wichtigste Datei ist die **config.xml**.

Der wichtigste Ordner ist **www**. In diesen Ordner legen wir die Dateien der Web-App (*.jpg, *.html, *.css, *.js).





Cordova auf Windows einrichten und ausführen

Eine Plattform hinzufügen:

Navigieren Sie nach dem Erstellen eines Cordova-Projekts zum Projektverzeichnis. Aus dem Projektverzeichnis müssen Sie eine Plattform (z.B: android oder ios) hinzufügen, für die Sie Ihre App erstellen möchten.

```
cordova platform add <platform name>
```

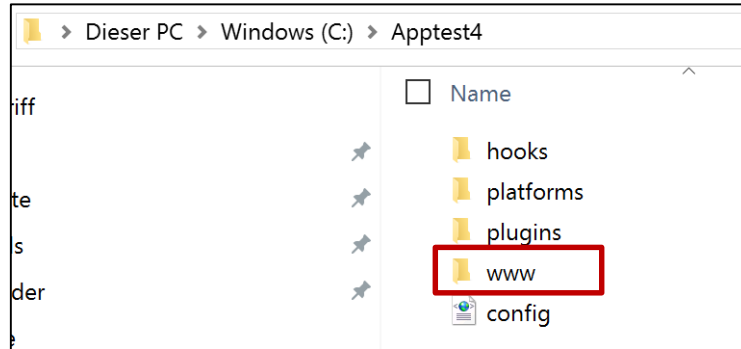
Um eine vollständige Liste der Plattformen zu erhalten, können Sie

```
cordova platform
```

aus.



Cordova auf Windows einrichten und ausführen

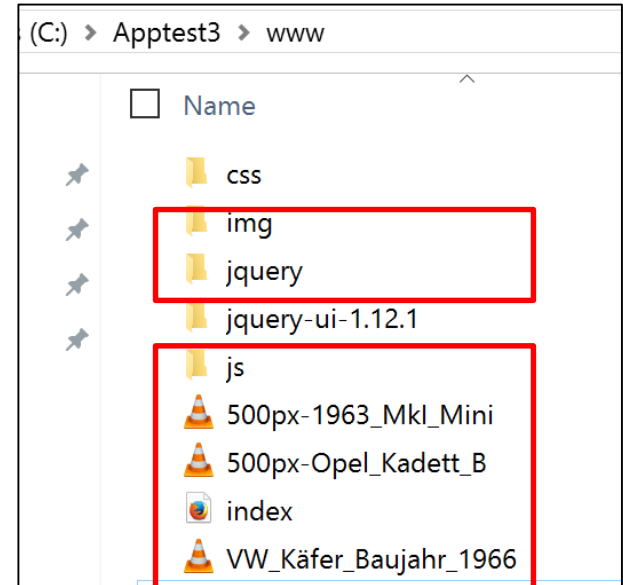


Im Verzeichnis Apptest4 sind nun alle Cordova-Daten angelegt und der Ordner `www`, in den alle von der Web-Seite benötigten Daten zu kopieren sind.



Cordova auf Windows einrichten und ausführen

Die rot umrandeten Daten stammen aus einer jQuery-Übung und sollen in eine Hybride App umgewandelt werden.





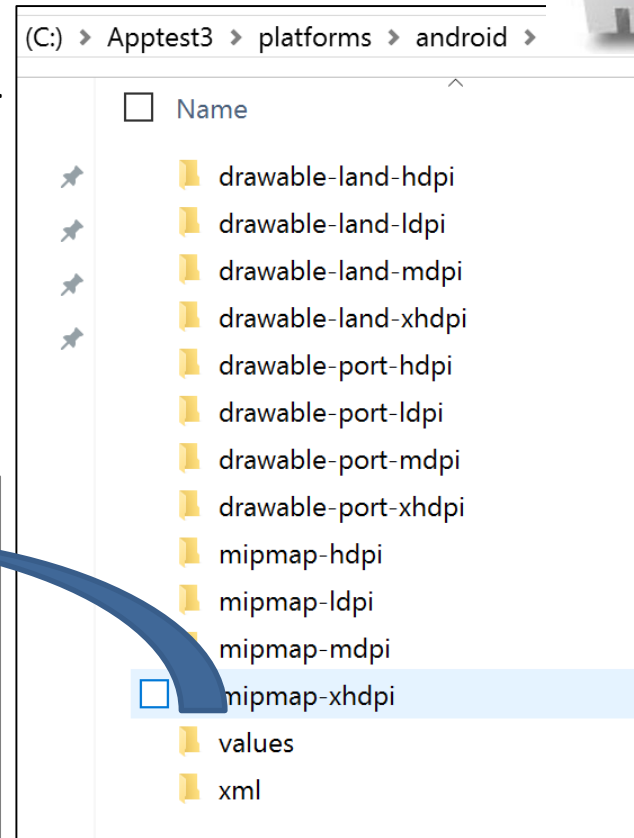
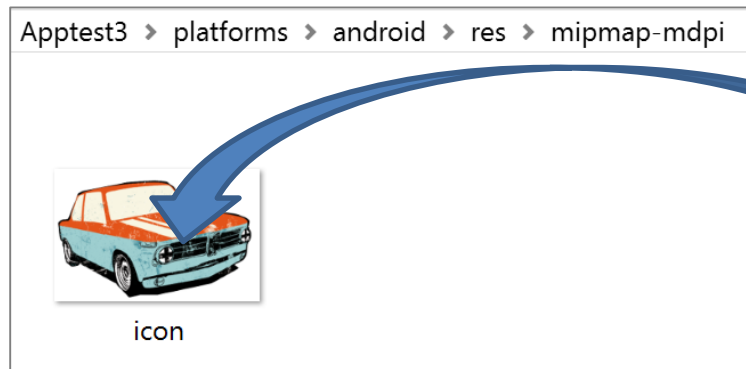
Cordova – config.xml einrichten

```
<?xml version="1.0" encoding="UTF-8" ?>
<widget xmlns      = "http://www.w3.org/ns/widgets"
        xmlns:gap   = "http://phonegap.com/ns/1.0"
        id          = "com.phonegap.example"
        versionCode = "10"
        version     = "1.0.0" >
<!-- versionCode is optional and Android only -->
  <name>webgl2</name>
  <!-- <platform name="ios" />    --> <!-- <platform name="winphone" />  -->
    <platform name="android">
      <icon src="icon.png" width="180" height="180" />
    </platform>
  <description>
    An example for webgl.
  </description>
  <author href="http://technischeredaktion.com/multimediamprojekte/"
        email="martin.schober@hs-karlsruhe.de">
    Martin Schober
  </author>
</widget>
```



Cordova auf Windows einrichten und ausführen

Erstellung der Icons für die App in unterschiedlichen Größen.





Cordova auf Windows einrichten und ausführen

Run your app

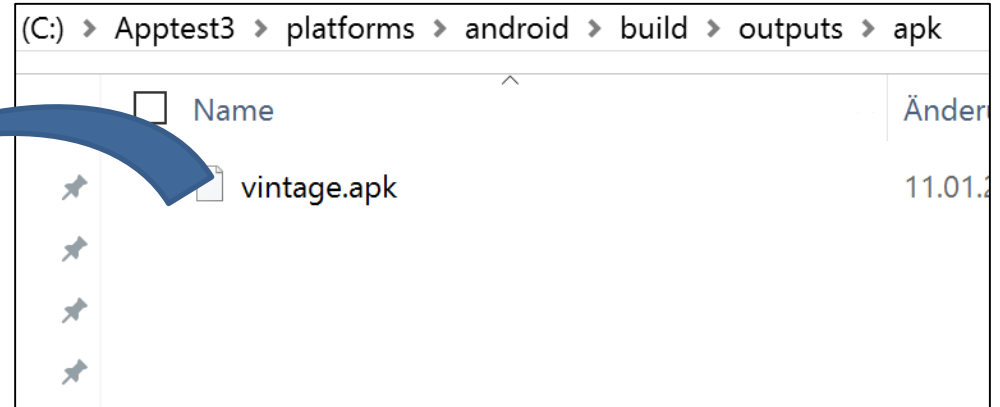
From the command line, run

```
cordova run <platform name>
```



Cordova auf Windows einrichten und ausführen

Die apk-Datei auf das Gerät mit Android-Betriebssystem übertragen und installieren.



Ionic – Open-Source-Framework für Hybride Apps als Alternative zu Apache Cordova



Ionic – Open-Source-Framework für Hybride Apps

Ionic ist ein MIT-lizenziertes Open-Source-Frontend-Framework zur Entwicklung von hybriden HTML5-Mobile-Apps.

Es beinhaltet die neuesten Entwicklungen der mobilen Plattformen und Standards.

HTML5, SASS-/CSS-Framework, AngularJS

Ionic integriert etablierte Technologien und stellt ein großes Ecosystem an weiteren Services bereit und verknüpft alles mittels „Glue“-Code und über ein mächtiges Kommandozeilentool (CLI).

Im Kern stellt Ionic ein Sass-/CSS-Framework (ähnlich Bootstrap/Foundation) zur Nachbildung von UI-Komponenten im nativen Look and Feel einer App dar.



Ionic – Open-Source-Framework für Hybride Apps

Voraussetzung für die Installation von Ionic ist Node.js.

Die Installation geschieht mit dem npm-Package-Manager, der Ionic und Cordova systemweit installiert.

Die Installation geschieht in der Windows-Konsole cmd.exe:

```
npm install -g cordova ionic
```

Die Überprüfung der erfolgreichen Installation:

```
ionic -version
```

Bei Erfolg wird die Versionsnummer ausgegeben.



Ionic – Open-Source-Framework für Hybride Apps

Ein neues Projekt erstellen:

```
ionic start TestApp [template]
```

Jetzt finden sich die Projektdateien in einem neuen Ordner mit dem Namen TestApp wieder. Der Parameter ***template*** ist optional und gibt an, welches der angebotenen Starter-Templates genutzt wird.

Mögliche Templates sind:

- | | |
|-------------------|--|
| tabs | - App mit Tab-Navigation (Standard) |
| sidemenu | - App mit einem Seitenmenü |
| maps | - Beispiel mit Google Maps |
| salesforce | - Kooperation zur Nutzung des Salesforce CRM |
| blank | - ein leeres Projekt |

Im [www-Verzeichnis](#) befinden sich jetzt die eigentlichen Anwendungsquellen, die für diese Einführung wichtig sind.



Ionic – Open-Source-Framework für Hybride Apps

Die Projektstruktur

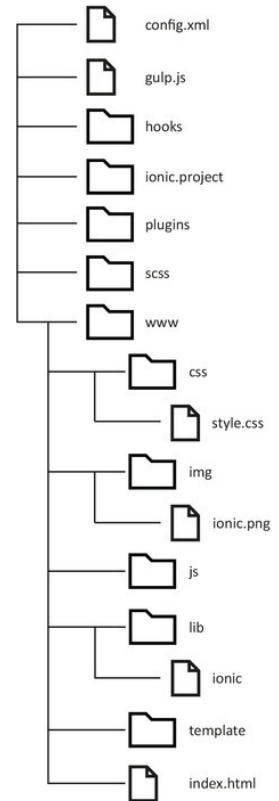
Das Framework aktualisieren: **ionic lib update**

Eine Plattform hinzufügen: **ionic platform add android**

Eine Plattform entfernen: **ionic platform remove android**

Die Anwendung im Browser ausführen: **ionic serve**

Simulation verschiedener Betriebssystem: **ionic emulate android**





Ionic – Open-Source-Framework für Hybride Apps

Bauen der Anwendung: `ionic build android`

Erstellen einer android-App mit Ionic Capacitor

```
ionic start multi blank
```

```
Angular yes
```

```
cd multi
```

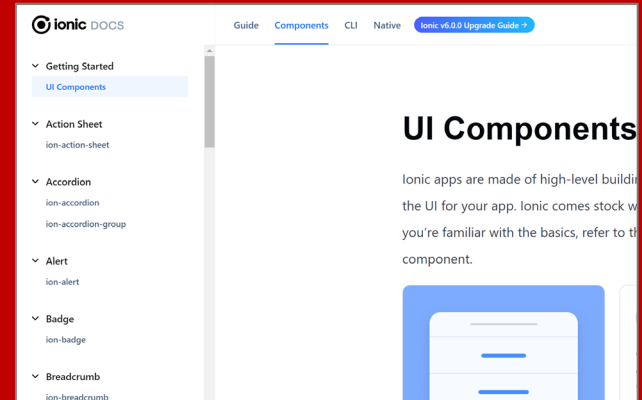
```
ionic serve (App wird im Browser dargestellt.)
```

```
ionic capacitor build android (App wird im Android Studio geöffnet.)
```

In Android Studio Build kann die apk-Datei erstellt werden.

<https://ionicframework.com/docs/components>

UI-components um eine App zu bauen.

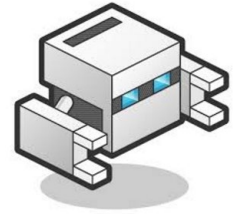


Erstellen einer android-App mit Ionic Capacitor

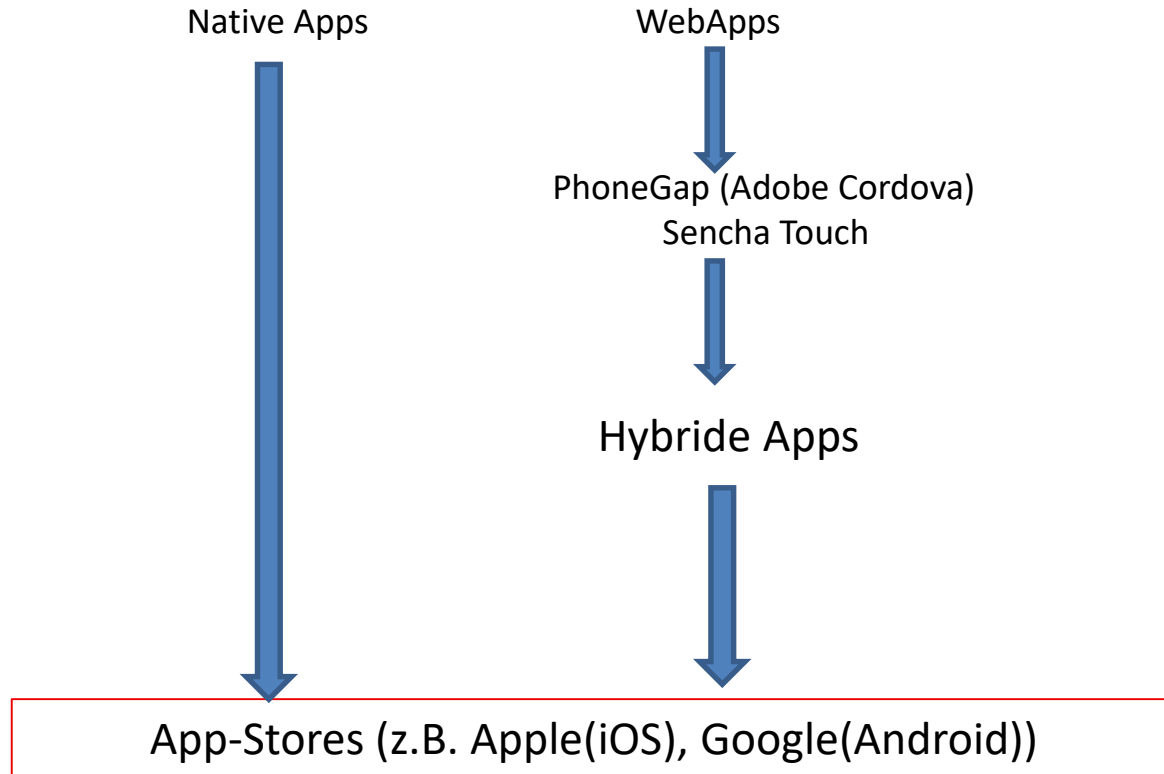
Im Ordner „Ihre-App\src\app\home finden Sie die Datei [home.page.html](#)

```
1 <ion-header [translucent]="true">
2   <ion-toolbar>
3     <ion-title>
4     Accordion
5   </ion-title>
6 </ion-toolbar>
7 </ion-header>
8
9 <ion-content [fullscreen]="true">
10  <ion-header collapse="condense">
11    <ion-toolbar>
12      <ion-title size="large">Blank</ion-title>
13    </ion-toolbar>
14  </ion-header>
15
16  <div id="container">
17    <ion-accordion-group>
18      <ion-accordion value="first">
19        <ion-item slot="header" color="light">
20          <ion-label>Ford Capri</ion-label>
21        </ion-item>
22        <div class="ion-padding" slot="content">
23          <ion-img src="../../assets/Capri81.PNG" alt="Capri"></ion-img>
24          <ion-text color="primary" style="text-align: left;">
25            <p>
26              Die dritte Generation des Ford Capri (MK 3) kam 1978 als Nachfolger des Ford Capri (MK 2)
27              auf den Markt. Der Ford Capri war ein dreitüriges Sportcoupé. Der Capri wurde mit folgenden
28              Motoren angeboten: 1,6 - 2,8 Liter-Benziner (70 - 188 PS). Der letzte Ford Capri MK 3 lief
29              1986 vom Band.
30            </p>
31          </ion-text>
32        </div>
33      </ion-accordion>
34      <ion-accordion value="second">
35        <ion-item slot="header" color="light">
36          <ion-label>Renault 4</ion-label>
37        </ion-item>
```

Diese ionic-tags finden Sie unter <https://ionicframework.com/docs/components> mit Beispielen!



Erstellung der Apps



Phonegap build ist Geschichte...

<https://appresser.com/phonegap-build-is-dead-here-are-some-alternatives/>

Ausblick

Die Entwicklung von Ionic 2 auf Basis von AngularJS 2 zeigt, dass Ionic langfristig eine Rolle spielen wird und erlaubt bereits einen Blick in die Zukunft.

Alternativen der neuen Cross-Compile-Tools sind **Native Script** oder **React Native**, die einen sehr interessanten Ansatz verfolgen:

Sie übersetzen Webcode in komplett native Apps.

Viele Projekte setzen auch verstärkt auf eine Mischung aus einer grundsätzlich nativen App, die aber große Teile an webbasierten Inhalten via WebView einbetten.