

Installation von PhoneGap auf OSX und Windows

Dieser Guide führt einen durch die Installation von PhoneGap auf OSX und Windows, um Android und iOS Applikationen zu entwickeln. Zum Testen von iOS Apps wird ein iOS Gerät benötigt. Das Bauen der finalen iOS App funktioniert entweder nur auf einem Mac oder über einen Cloud Service wie PhoneGap.

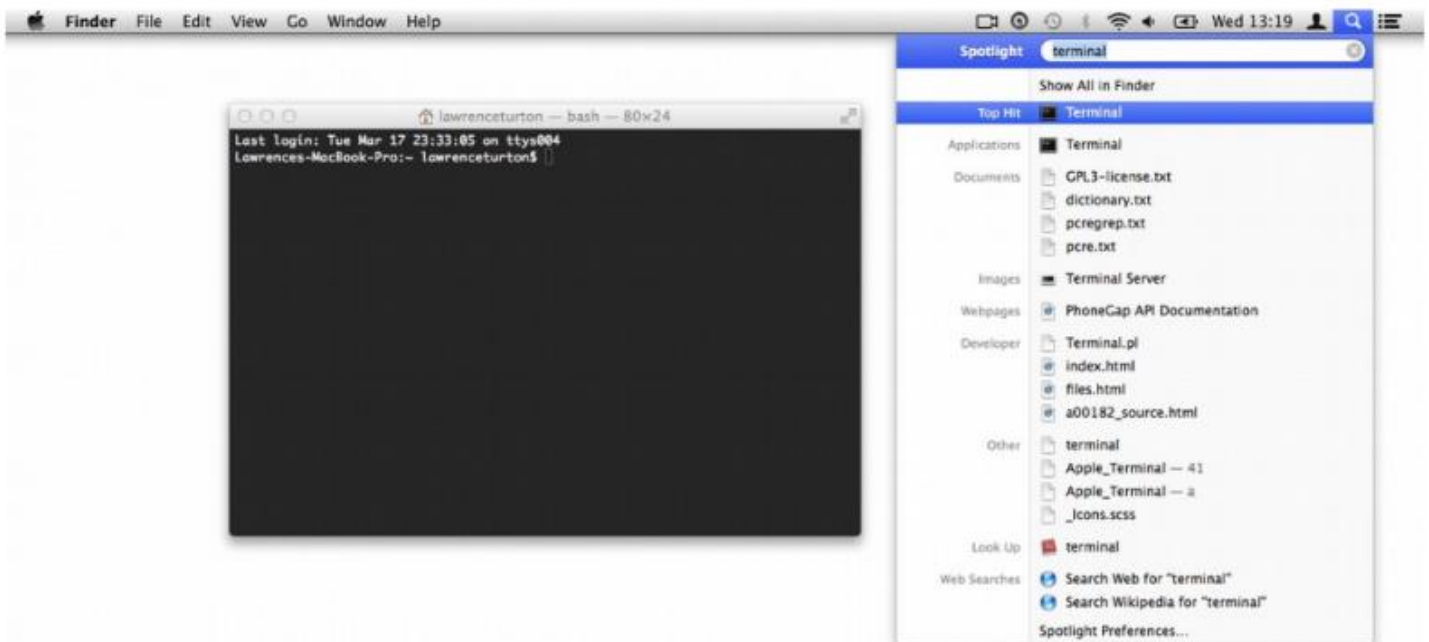
Installation von PhoneGap auf OSX



Der erste Teil von „Installation von Phonegap auf OSX“ setzt die Testumgebung auf. Diese Schritte werden für Android Emulation und IOS Simulation auf OSX benötigt



1. Herunterladen der NodeJS DMG Datei von der Webseite:
<https://nodejs.org/download/>
2. Nach dem NodeJS Setup gebt `node -v` im Terminalfenster ein, um zu prüfen, ob NodeJS erfolgreich installiert wurde. Die Versionsnummer sollte erscheinen.





Installation von Git auf OSX

1. Navigiert zur Git Webseite und ladet die neuste Version von Git für OSX herunter und installiert die heruntergeladene DMG Datei. <https://git-scm.com>
2. Öffnet das Terminal und gibt `cd ~` ein, damit das Terminal auf das Home-Verzeichnis zeigt
3. Gebt `open -e .profile` ein, wodurch eure Profil Datei im Text Editor geöffnet werden sollte
4. Gebt oder kopiert den folgenden Text in die Profile Datei und speichert sie ab
`export PATH=/usr/local/git/bin:$PATH`
5. Gebt im Terminal `..profile` ein
6. Führt den Befehl `git --version` aus, wodurch die Versionsnummer angezeigt werden sollte



Installation von Phonegap auf OSX

1. Um PhoneGap zu installieren, gebt `sudo npm install -g phonegap` ein. Hierfür wird euer Passwort benötigt

```
sudo npm install -g phonegap
```

2. Um eine leere App auf dem Desktop zu erzeugen, gebt `cd desktop` ein und danach `phonegap create myapp`. Statt myapp könnte ihr auch einen anderen Namen wählen.
3. Um die App über PhoneGap zu testen, müssen wir die Dateien über WIFI zur Verfügung stellen. Wechselt hierzu in das Verzeichnis der App (`cd myapp`).
4. Im Verzeichnis unserer App gebt `phonegap serve` ein

```
[phonegap] starting app server...  
[phonegap] listening on 192.168.0.4:3000  
[phonegap]  
[phonegap] ctrl-c to stop the server  
[phonegap]
```

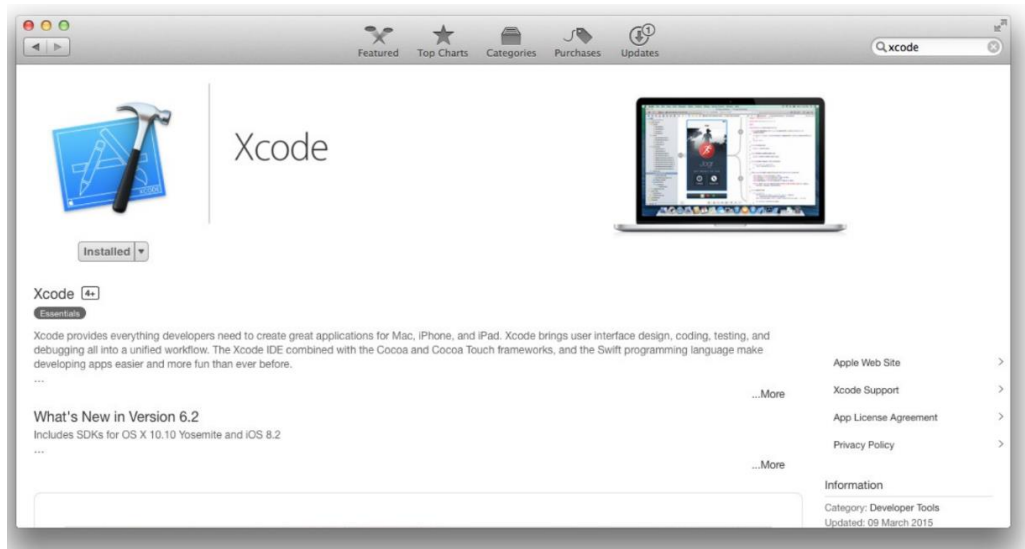
5. Installiert auf eurem Handy die Phonegap Developer App und gebt die IP Adresse (in meinem Fall 192.168.0.4:3000) in das vorgegeben Fenster ein.



Das Terminalfenster auf dem Rechner muss während dem Testen offenbleiben, da sonst die Übertragung abgebrochen wird.

Setup von iOS Simulation auf OSX

1. Installiert die neuste Version von Xcode aus dem App Store.



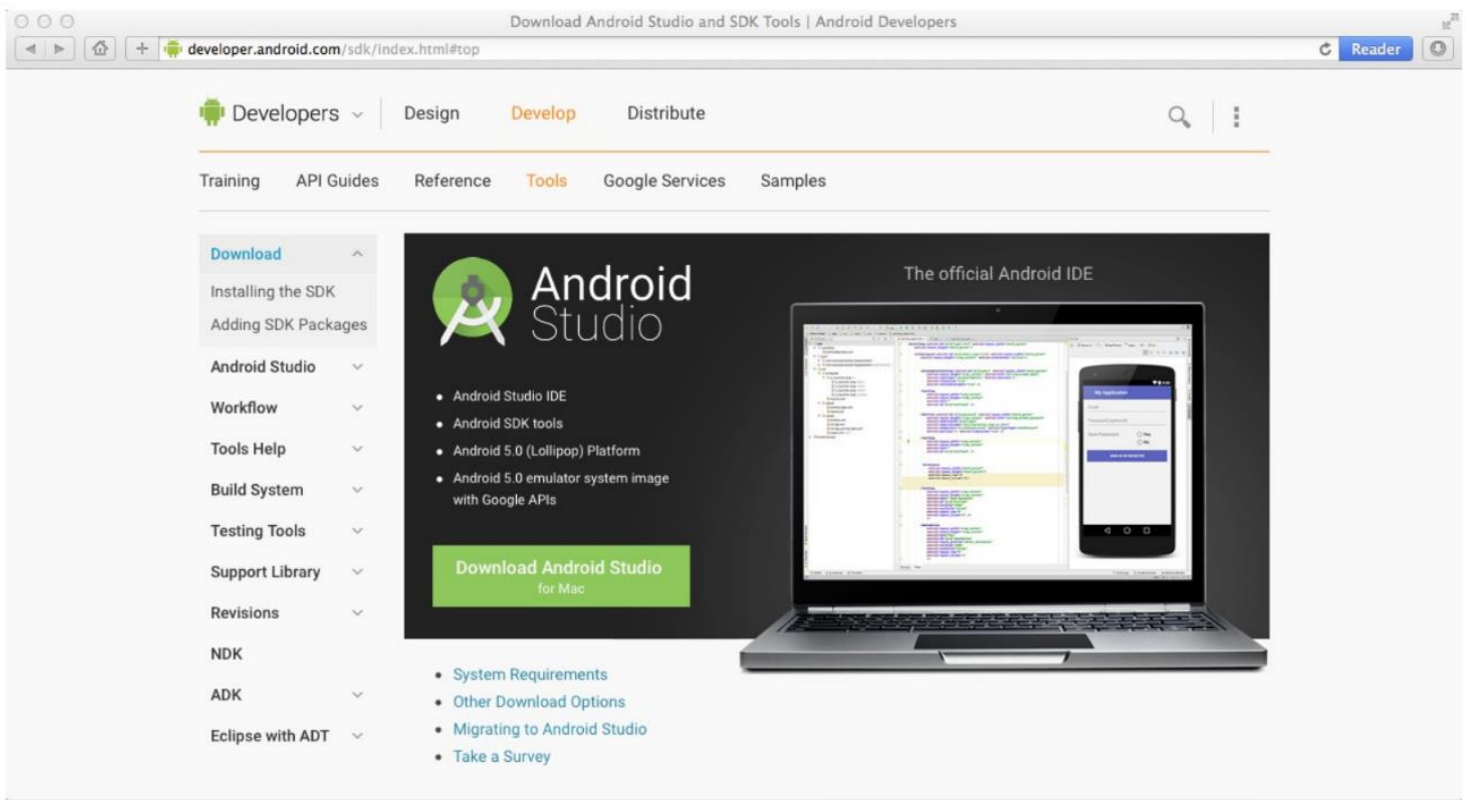
2. Öffnet euer Terminal und gebt `sudo npm install -g cordova` ein
3. Navigiert zu dem Verzeichnis eurer App (`cd myapp`)
4. Im Verzeichnis eurer App gebt den Befehl `cordova platform add ios` ein.
5. Um die App jetzt im IOS Simulator auszuführen, führt den Befehl `cordova run ios` im Terminal aus. Das wird paar Minuten in Anspruch nehmen.



6. Statt des Standardgeräts können auch andere Geräte verwendet werden. Um alle verfügbaren Geräte anzuzeigen, führt den Befehl `ios-sim showdevicetypes` aus. Um z.B. jetzt das iPhone 4s zu verwenden, gebt `cordova run ios --target="iPhone-4s"` ein

Setup des Android Emulators auf OSX

1. Ladet Android Studio von <http://developer.android.com/sdk/index.html> herunter



2. Wenn heruntergeladen, öffnet die DMG Datei und zieht das Android Studio über den Applications Ordner.



3. Öffnet den Applications Ordner und führt die Installation von Android Studio aus
4. Öffnet euer Terminalfenster und gebt folgenden Befehl ein

open -e ~/.profile

5. Gebt die folgenden Befehle ein (Wichtig: Zeile bei Zeile kopieren und nicht alle Zeilen auf einmal!)

export ANDROID_HOME=~/.Library/Android/sdk

export PATH=\$PATH:\$ANDROID_HOME/tools

export PATH=\$ANDROID_HOME/platform-tools:\$PATH

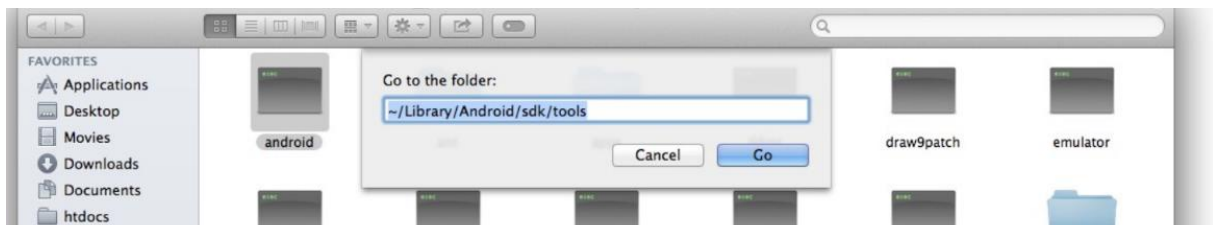
- a. Wechselt zum Terminal und aktualisiert euer Profil durch die Eingabe von:

..profile

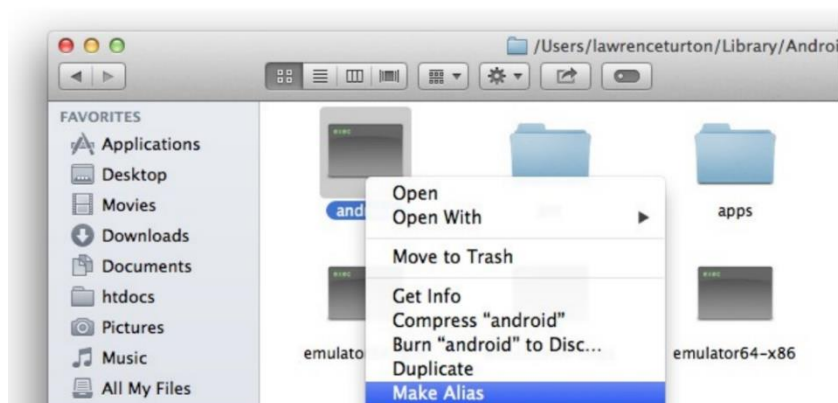
- b. Um zu prüfen, ob alles funktioniert, gebt ein:

Android

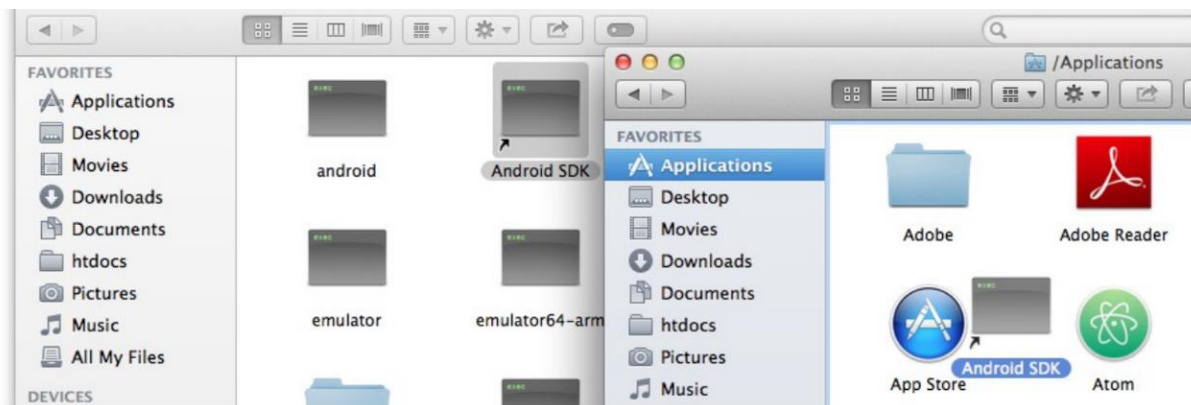
6. Das Android SDK befindet sich im Library Verzeichnis. Damit es einfacher ist den SDK Manager im Applications Ordner zu finden, erzeugen wir einen Alias. Öffnet das „Go-Menu“ im Finder und wählt „go to folder“ aus. Und gebt **~/Library/Android/sdk/tools** ein.



- a. Rechts klickt auf die Android Exe und wählt „Make Alias“ aus.

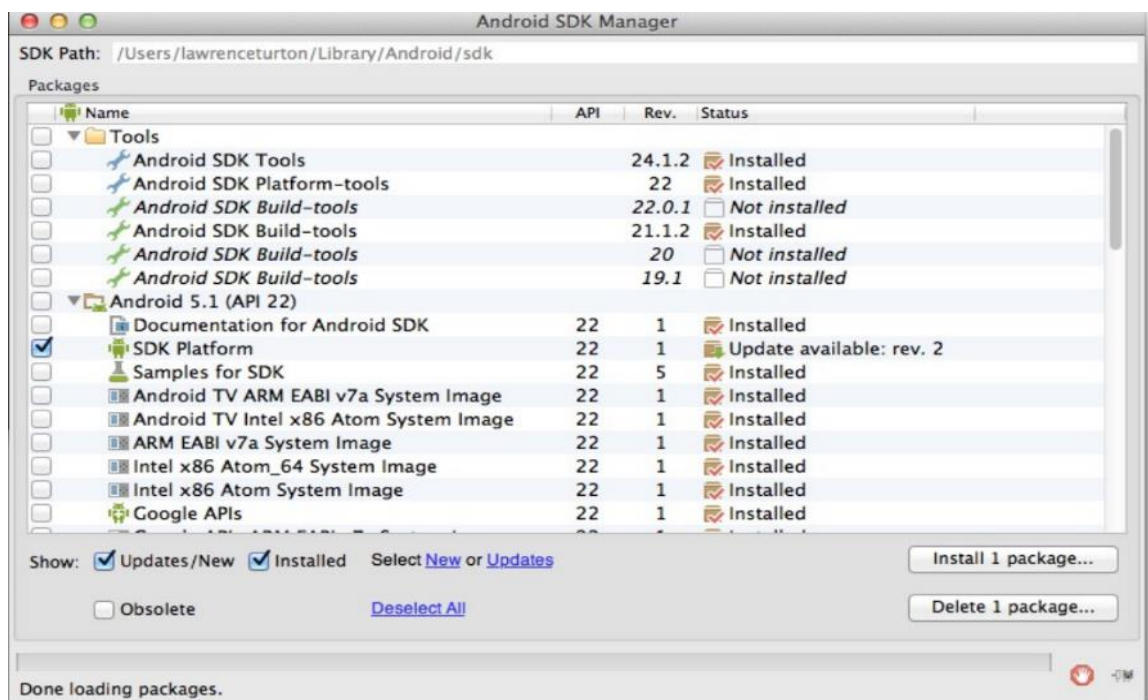


- b. Zieht den Alias in euren Applications Ordner

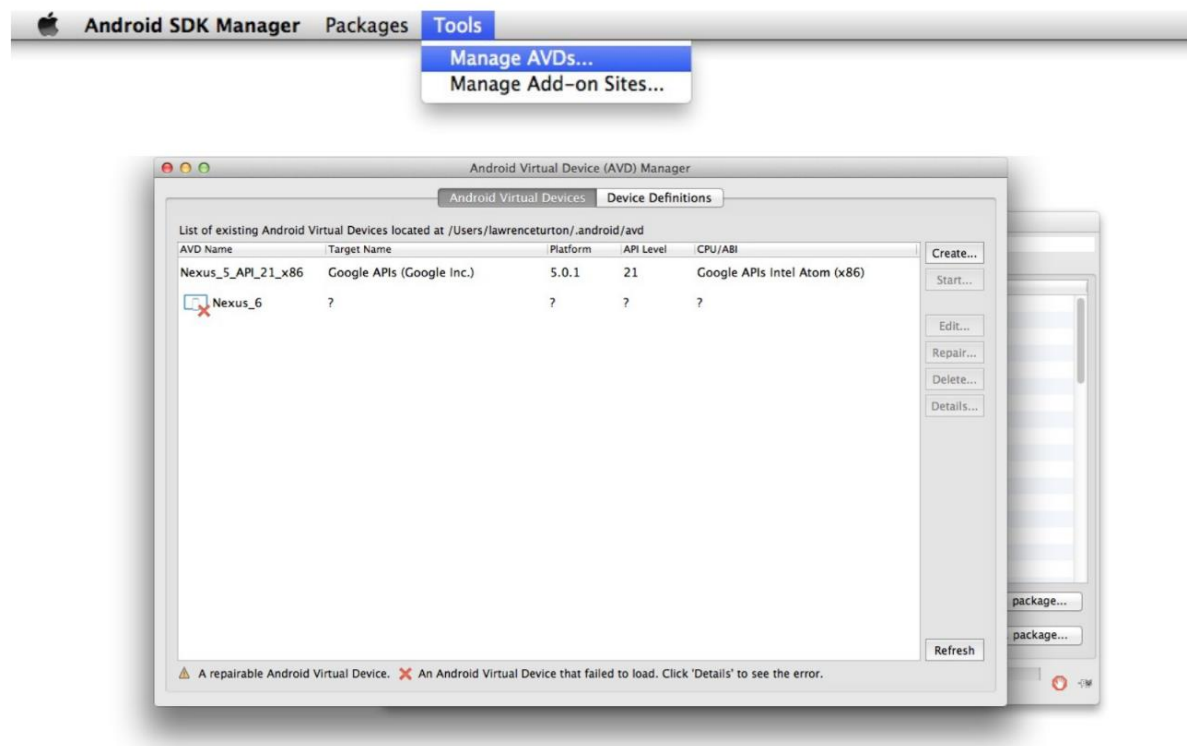


7. Doppel klickt auf den Alias in eurem Applications Ordner. Hierdurch wird der Android SDK Manager gestartet.

Von hier aus können wir zusätzliche SDK's installieren (Installiert den API Level 25).



Über das Tools-Menü kommen wir zum AVD Manager. AVD steht für Android Virtual Device. Von hier aus könnt ihr eure Emulation Umgebung konfigurieren.



8. Öffnet nach der Erstellung des AVDs die Konsole, wechselt in das Verzeichnis eurer App und gebt den Befehl `cordova run android --target="Nexus_6"` ein. Der target-Parameter ist optional und benötigt den Namen des AVDs, welches ihr verwendet möchtet. Es kann bis zu 10min dauern bis sich der Emulator öffnet!