

LiveCode-App in den Google Play Store bringen

Das höchste Ziel einer gelungenen und brauchbaren App für Android besteht darin, sie im Google Play Store zu veröffentlichen.

Wie macht man das?

Voraussetzung: Du hast eine App in LiveCode entwickelt, die funktioniert und als APK kompilierbar ist und die du bereits im eigenen Gerät oder im Emulator getestet hast, so dass du bereits weißt, dass sie sich auf dem Android-Gerät so verhält wie sie soll. Dann kann sie in den PlayStore.

Was brauchst du dazu?

Zunächst einmal einen *Google Developer Account*. Wenn du bereits Google-Dienste nutzt und somit schon eine Google-ID besitzt, kannst du den Developer Account zu deiner bestehenden Google-ID hinzubuchen, oder du legst eine ganz neue an.

Dies geht unter:

<https://accounts.google.com>

„Konto erstellen“.

Nachdem du ein Google-Konto angelegt hast, musst du dich als *Google Developer* registrieren. Dies kostet einmalig 25 Dollar und kann nur mit Kreditkarte bezahlt werden.

Du findest die Informationen dazu auch alle hier:

<https://support.google.com/googleplay/android-developer/answer/6112435?hl=de>

Einmal als Google Developer angemeldet, hast du schon einmal alle formalen Voraussetzungen, um deine Apps im Google Play Store veröffentlichen zu können (OpenSource-Apps mit der LC-Community Edition, wohlgemerkt).

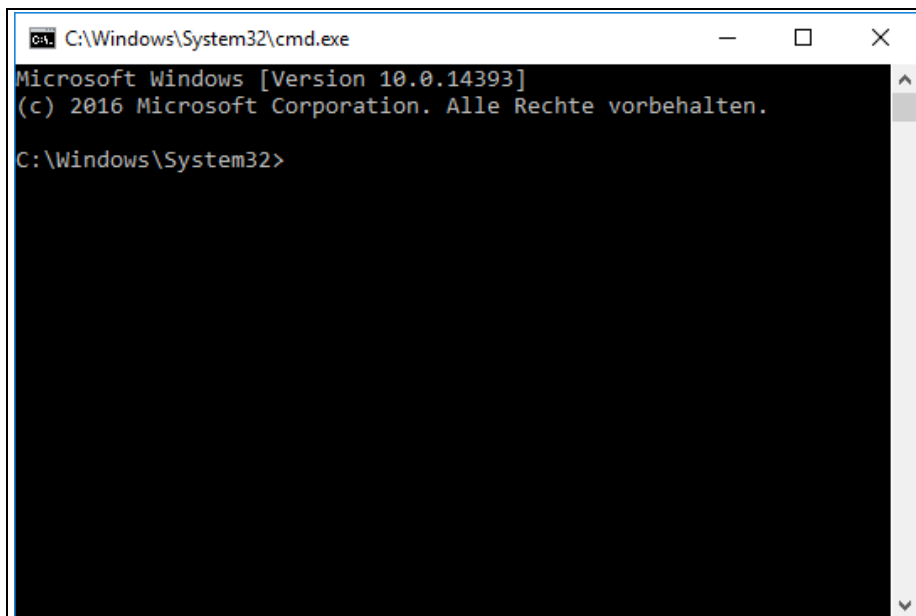
Die App selber benötigt aber noch ein gültiges Zertifikat (Schlüssel) zum Verschlüsseln in Form einer Datei, damit sie in den Store geladen werden kann.

Es gibt einen einfachen Weg, einen *keystore-Schlüssel* für Ihre App zu erstellen. Verwende dazu einfach das praktische **Keymaker-Tool** für

Windows oder Mac (siehe etwas weiter unten). Für Profis, die alles manuell machen möchten und die maximalen Einflussmöglichkeiten haben wollen, ist hier aber zunächst der herkömmliche Weg über die Konsole beschrieben:

Vorbereitung Schlüsselerstellung für Windows:

Du musst unter Windows das Command-Line-Programm keytool.exe verwenden. Das befindet sich im bin-Verzeichnis des Java SDK. Öffne ein CMD-Fenster (Command Window). Du kannst in Windows nach CMD suchen, oder du findest das Programm in C:\Windows\System32 unter cmd.exe und startest es dort – mit Rechtsklick „Als Administrator starten“.



Das Command-Fenster in Windows

Hier musst du in das richtige Verzeichnis des Java SDK navigieren. Das geschieht mit folgendem Befehl (im CMD-Fenster):

```
cd C:\Program Files\Java\jdk1.8.0_91\bin\
```

Wenn deine Java-Version eine andere ist, muss die Nummer jdk1.8.0_91 entsprechend ersetzt werden. Schau mit dem Datei-Explorer sicherheitshalber nach in C:\Programme\Java nach, wie das Verzeichnis bei dir genau heißt. Sonst gelangst du nicht in das richtige Verzeichnis.

Nach Eingabe der RETURN-Taste solltest du im entsprechenden Verzeichnis sein.

Generieren der Schlüsseldatei (Windows und Mac)

Auf dem Mac brauchst du nicht in das Java-Verzeichnis navigieren, sondern du öffnest einfach nur das Terminalprogramm und gibst den Befehl dort direkt ein.

Gib jetzt also im CMD-Fenster bzw. im Terminalfenster auf dem Mac den Befehl zum Erstellen der Schlüsseldatei ein. Das ist beim ersten Mal ein bisschen kompliziert.

Vorher überlegst du dir einen passenden internen Namen für deine App (ohne Leerzeichen und Sonderzeichen). Das ist der Aliasname, den du anschließend gleich benötigst. Dann gibst du exakt das hier ein:

```
keytool -genkey -v -keystore releasekey.keystore -alias  
aliasname -keyalg RSA -keysize 2048 -validity 10000
```

Den Begriff aliasname ersetzt du dabei durch den Namen deiner App, zum Beispiel lottomaschine oder wuerfelapp.

Also zum Beispiel so:

```
C:\Program Files\Java\jdk1.8.0_91\bin>keytool -genkey -v -keystore releasekey.keystore -alias wuerfelapp -keyalg RSA -keysize 2048 -validity 10000
```

Wenn du alles exakt richtig eingegeben hast und RETURN drückst, wird dir das keytool-Programm einige Fragen stellen, die du möglichst korrekt beantworten solltest (wenn du etwas ganz falsch machst, starte das Tool anschließend einfach noch einmal):

Als erstes fragt das Programm nach dem Keystore-Kennwort. Denke dir ein einfaches Kennwort aus (ohne Leerzeichen), das du dir merken kannst, gib es ein und drücke RETURN. Anschließend muss das Keyword noch einmal zur Wiederholung eingegeben werden.

Danach gibst du deinen Vor- und Nachnamen ein, dann den Namen deiner organisatorischen Einheit (Leereingabe mit RETURN geht auch), dann den Namen deiner Organisation oder Firma (auch hier geht Leereingabe), danach den Namen deiner Stadt, dann das Bundesland, dann „de“ für Deutschland als Länderkürzel. Danach fragt das Programm noch einmal, ob alles stimmt, tippe „ja“ ein und drücke RETURN. Zum Schluss musst du noch das

„Schlüsselkeyword“ für die App eingeben. Drücke einfach RETURN, dann ist es identisch mit dem am Anfang eingegebenen Keystore-Kennwort.

Anschließend wird deine Schlüsseldatei erstellt. Du findest sie unter dem Namen `releasekey.keystore` im bin-Verzeichnis, wo auch die `keytool.exe` Datei ist. Kopiere sie heraus in dein Projektverzeichnis der LiveCode-App.

Verzweifle nicht, wenn das Erstellen des Schlüssels nicht auf Anhieb klappt. Gehe langsam Schritt für Schritt vor. Wenn du es exakt so machst wie hier beschrieben, wird es klappen. Wenn du das ganze später erneut machst, geht es schon ganz schnell. Oder du verwendest gleich den Keymaker – dann sparst du dir viel Mühe:

Schlüsseldatei erstellen – der einfache Weg

Mithilfe des Tools „keymaker“, das ich eigens für die Leser meines Buchs erstellt habe, geht die Schlüsselerstellung ganz einfach. Das ist sicherlich vor allem für Anfänger, die nicht mit der Konsole vertraut sind, angenehmer.

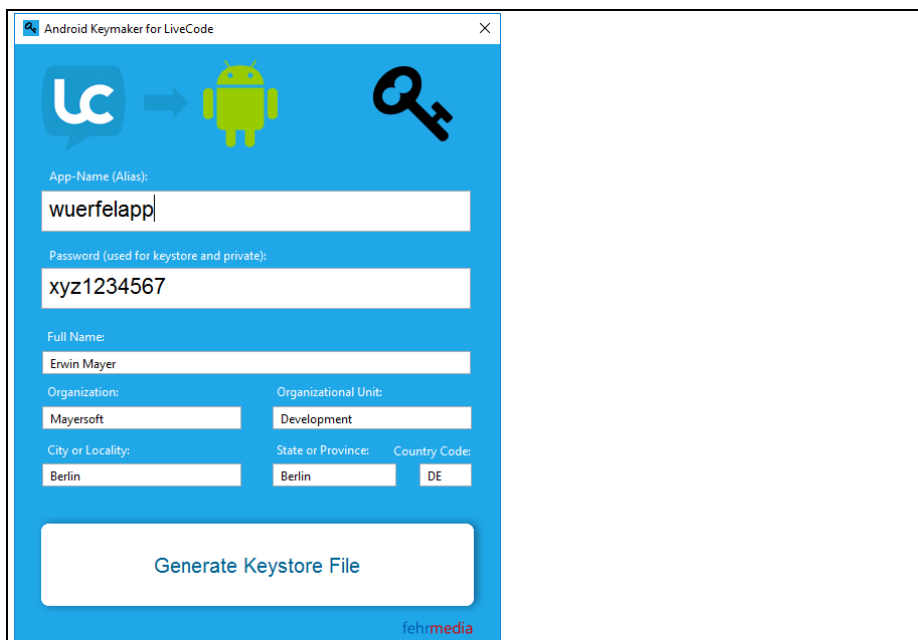
Lade dir dazu folgendes Programm herunter:

http://www.fehr-media.de/Keymaker_Windows.zip

bzw. für den Mac:

http://www.fehr-media.de/Keymaker_Mac.zip

Entpacke die heruntergeladene ZIP-Datei und starte das Programm.



Der KeyMaker vereinfacht das Erstellen einer Keystore-Datei für Android-Apps

Gib in den oberen Zeilen einfach den Alias-Appnamen (ohne Leer- und Sonderzeichen) ein, darunter das Passwort, das sowohl für den Store als auch als Private-Passwort verwendet wird. Diese beiden Einträge sind die wichtigsten, und du solltest sie dir merken oder aufschreiben, denn du brauchst sie später wieder.

Darunter gibst du deine Daten ein – wichtig ist hierbei nur, dass alles ausgefüllt ist. Die Daten werden gespeichert und erscheinen beim nächsten Start des Programms automatisch, so dass du zukünftig nur noch den App-Alias-Namen und das Passwort eingeben musst.

Klicke auf „Generate Keystore File“ – und es erscheint nach 1-2 Sekunden ein Speichern-Dialog, in dem du wählen kannst, wohin die Schlüsseldatei gespeichert werden soll – am besten in das Projektverzeichnis der zugehörigen App. Fertig!

App mit dem Schlüssel zertifizieren

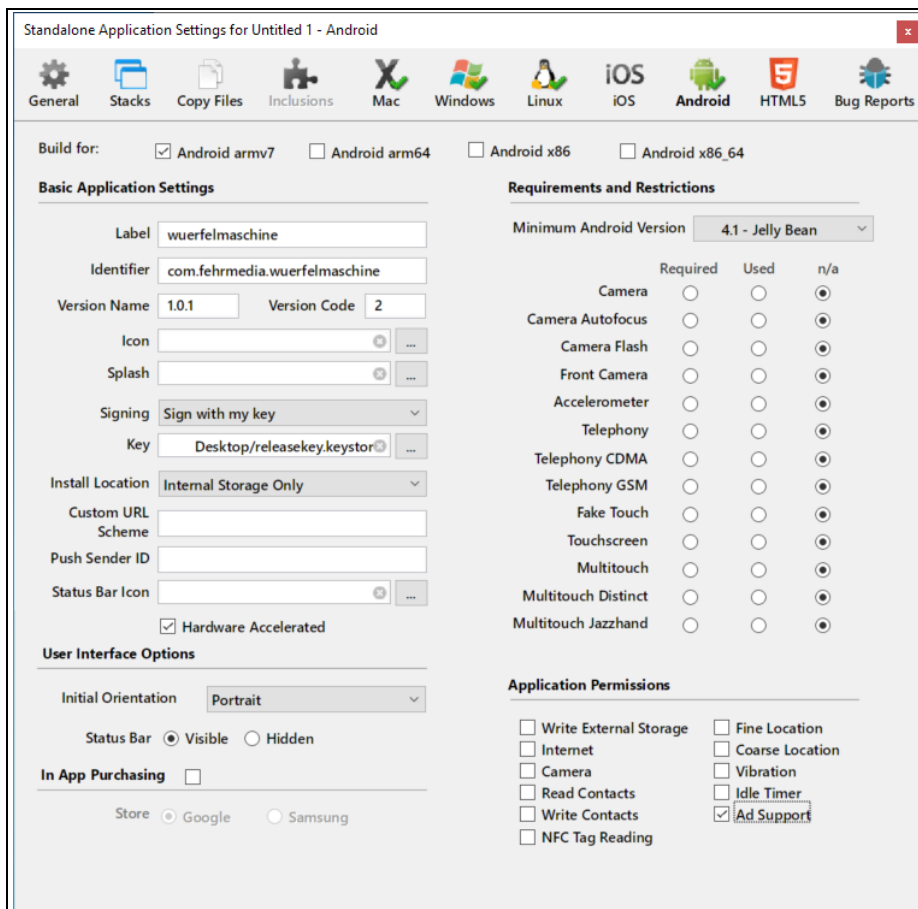
Nun hast du eine Schlüsseldatei und bist Google Developer. Damit ist alles vorhanden, um deine App in den Google PlayStore zu laden.

Dafür musst du den eben erstellten Schlüssel nur noch bei LiveCode für die entsprechende App eintragen, damit sie mit diesem Schlüssel zertifiziert werden kann. Dann vervollständigst du die Informationen für die App in LiveCode und kannst danach ganz einfach über LiveCode eine Standalone-Version deiner App erstellen, die direkt in den Google Play Store hochgeladen werden kann.

Starte LiveCode und lade die App, die in den PlayStore kommen soll.

Gehe auf File – Standalone Application Settings. Dort wählst du Android. Gib unter Label den Namen ein, der unter dem Icon in Android erscheinen soll und als Identifier einen Namen ein, der mit com. beginnt und anschließend deinen Namen (oder Firmennamen) und nach einem weiteren Punkt den internen App-Namen enthält (alles ohne Leer- und Sonderzeichen).

Unter „Signing“ wählst du „Sign with my key“ und unter „key“ wählst du die `releasekey.keystore` Datei an, die du zuvor erzeugt hast.



Beispiel für Standalone-Settings Android für den PlayStore

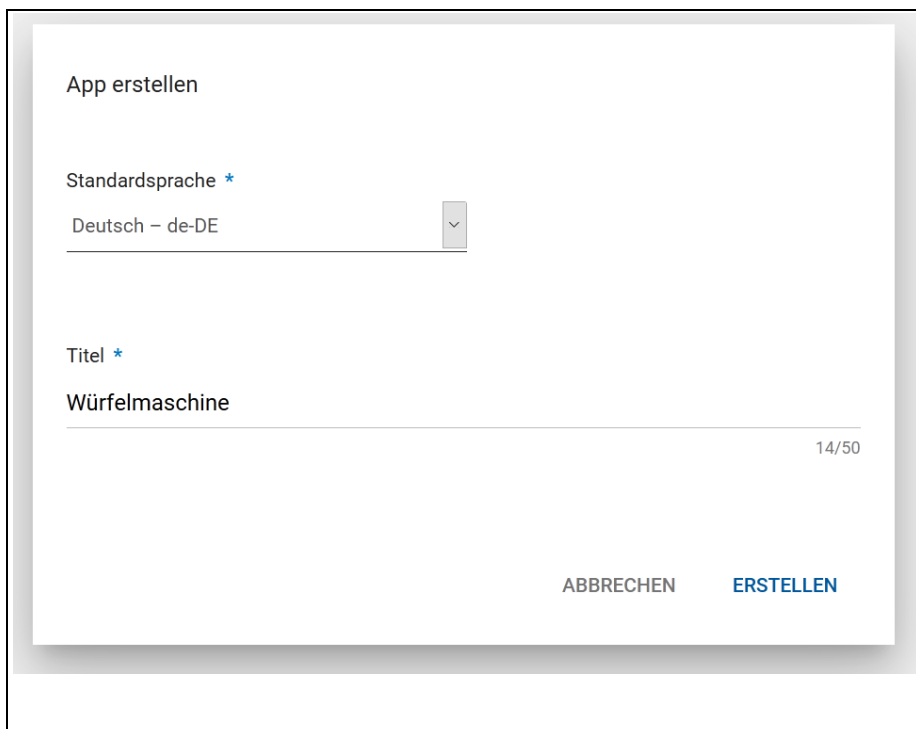
Speichere das Projekt. Anschließend kannst du in LiveCode unter File – SAVE AS STANDALONE APPLICATION dein Projekt als nunmehr vollwertige und zertifizierte Android APK speichern. Bei der Erstellung wird LiveCode nach dem Projektnamen fragen (gib hier exakt den Aliasnamen aus der Schlüsselerstellung an) und zwei Mal nach dem Passwort. Dieses sollte beide Male identisch sein und stammt ebenfalls aus der Schlüssel-Erstellung. Danach hast du eine PlayStore-fähige APK-Datei deiner App. Glückwunsch! Der Rest findet jetzt bei Google statt und hat mit LiveCode nichts mehr zu tun.

Du musst die Datei per Webbrowser in den Google Play Store hochladen und dort entsprechende Einstellungen für die Präsentation der App im Store setzen.

Öffne deinen Webbrowser und navigiere zur Google Play Developer Console.

<https://play.google.com/apps/publish/>

Wenn du eingeloggt bist, klicke auf „App erstellen“. Gib einen Titel für das Google Play Projekt ein (also der Name deiner App).



App erstellen

Standardsprache *

Deutsch - de-DE

Titel *

Würfelmaschine

14/50

ABBRECHEN ERSTELLEN

Nach Klick auf „Erstellen“ muss du eine Menge Infos über deine App eingeben. Du solltest ein Symbol für den PlayStore hochladen (am besten png 512x512 sowie Angaben über den Inhalt machen.

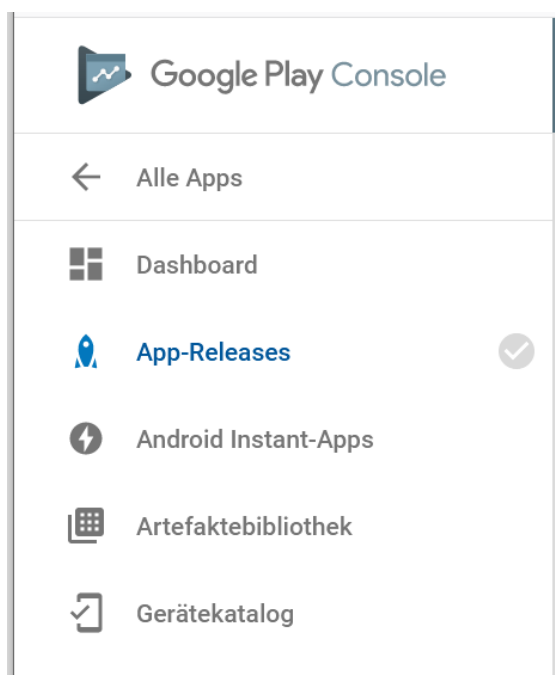
Du musst den Typ der App auswählen, deine Kontaktdaten und Links zum Impressum und Datenschutzerklärung angeben.

Zur Präsentation der App solltest du Screenshots aus der App hochladen (Du kannst die Screenshots direkt auf deinem Mobilgerät erstellen und dir auf den Computer laden oder sie aus LiveCode mittels eines Screenshot-Tools herstellen). Die Screenshots sollten im Format PNG oder JPG sein – ansonsten gibt es keine strengen Formatvorgaben. Den Titel des Programms und die „Funktionsgrafik“ erstellst du vielleicht mit Hilfe von Screenshots oder dem Titelbild der App in einem Grafikprogramm. Das kannst du machen wie du möchtest.

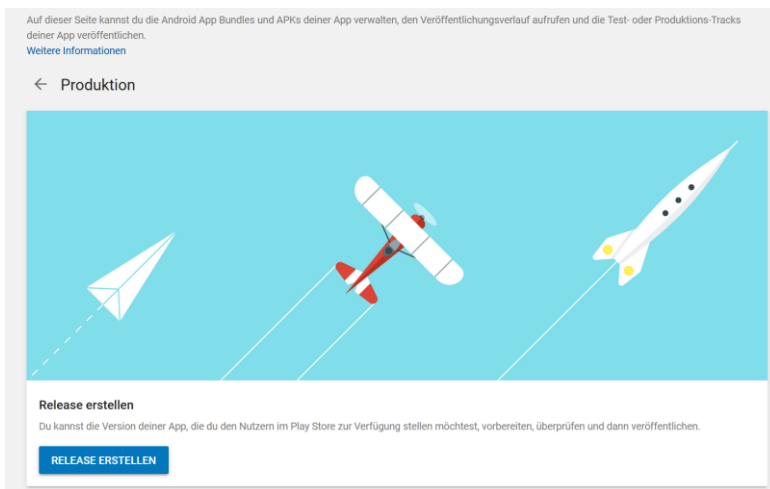
Ebenfalls musst du den Inhalt einstufen, den Preis festlegen (mit LiveCode Community erstellst du eher kostenlose Apps, ansonsten benötigst du eine kommerzielle Version von LiveCode sowie einen Google Merchant Account) und ein paar weitere Einstellungen setzen.

Am Schluss klickst du auf „Entwurf speichern“ und hast somit deine App bei Google angemeldet. Sie muss allerdings auch noch hochgeladen werden.

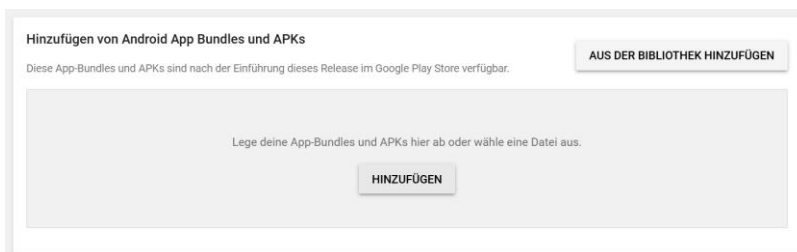
Dazu klickst du links auf „App Releases“:



Hier wählst du „Release erstellen“.



Nun kannst du deine aktuelle APK hochladen:



Klicke auf „Hinzufügen“ und lade deine APK hoch .

Du musst auch noch eine Bezeichnung für die Version der APK eingeben.

Anschließend klickst du auf „Überprüfen“.

Nun ist Google an der Reihe. Deine App wird überprüft, das kann ein paar Stunden dauern, manchmal auch einen Tag – möglicherweise gibt es noch Probleme, die Google dir danach mitteilt... oder die App kommt gleich in den Store.

Bei allen weiteren Fragen, die in diesem Prozess entstehen können, hilft dir (im wahrsten Sinne des Wortes) Google weiter.

Letztendlich wird es irgendwann funktionieren, und du bist App-Autor einer im Google Play Store erhältlichen App! Ich wünsche dir viel Erfolg mit deinen eigenen Apps!