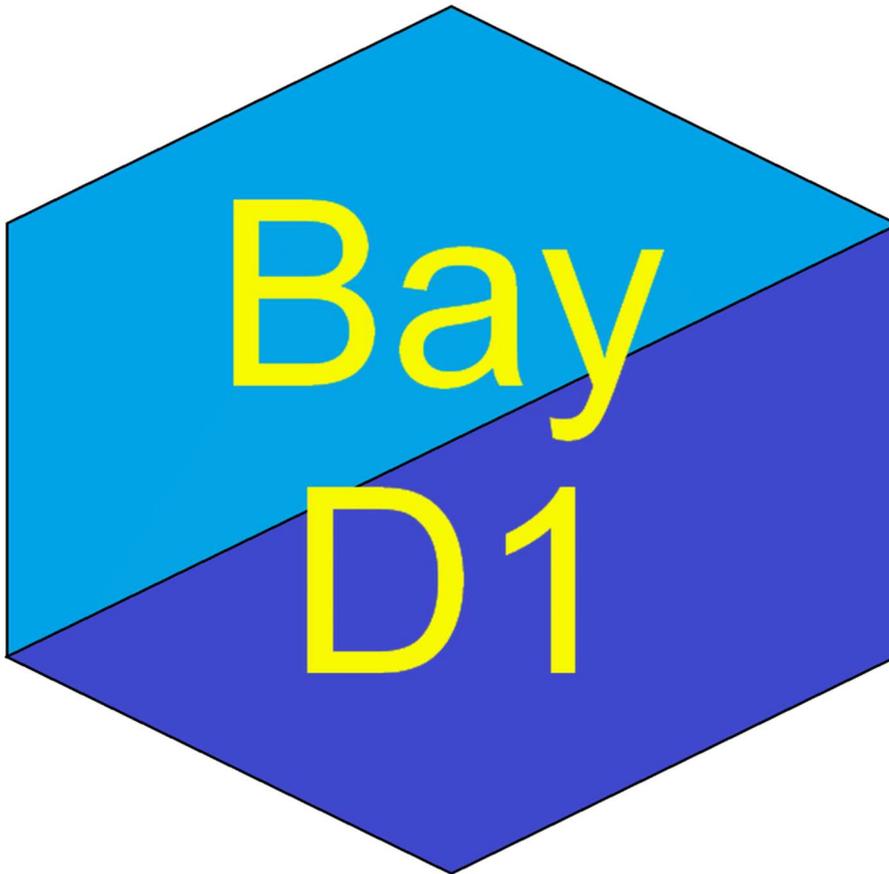


App Icons

Dazu braucht man zunächst ein gewünschtes Bild, dass als Icon erscheinen soll. Es sollte möglichst groß sein, damit aus diesem alle anderen Größen generiert werden können. Mein Icon für die erste Demo:



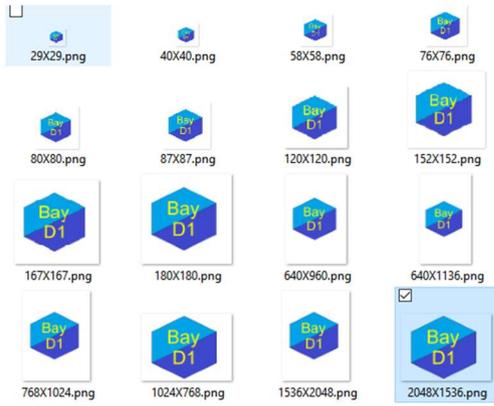
Dann gibt es auf Github ein Tool, mit dem man fast alle anderen gewünschten Größen erzeugen kann:

<https://github.com/techierathore/ReSizer>

(Achtung: in moodle gibt es jetzt eine von mir verbesserte Version in Ordner „Icons für Apps“ namens ReSizer-master Bay.zip)

Dieses Github-Tool habe ich benutzt, um fast alle möglichen Größen automatisch zu erzeugen:

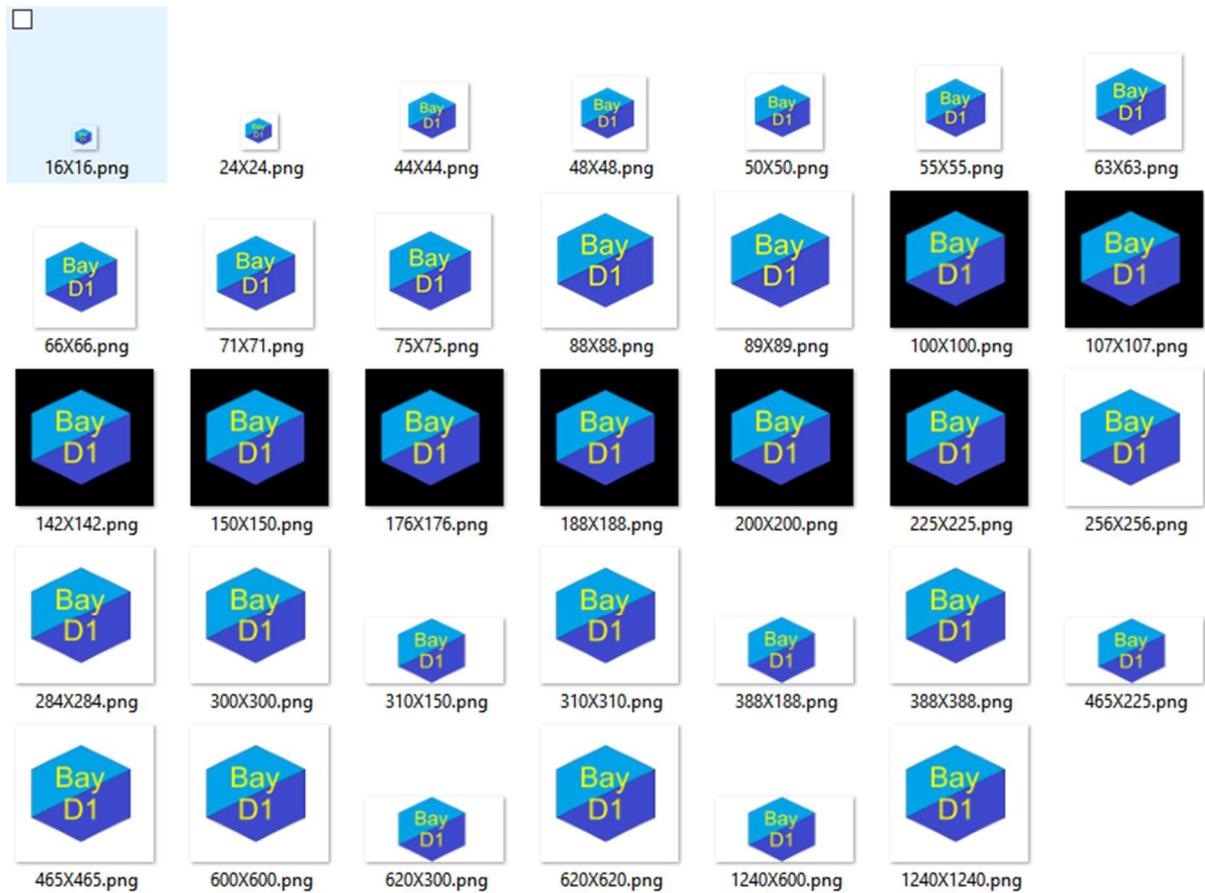
IOS- Images:



Android Images:



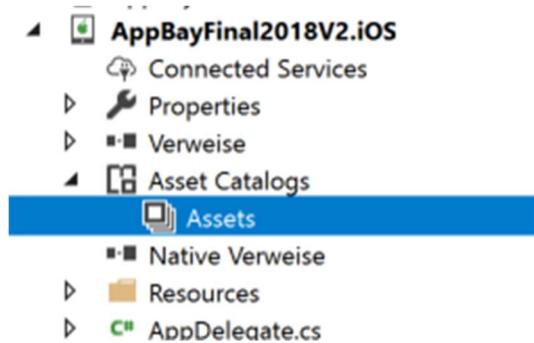
Und UWP- Images:



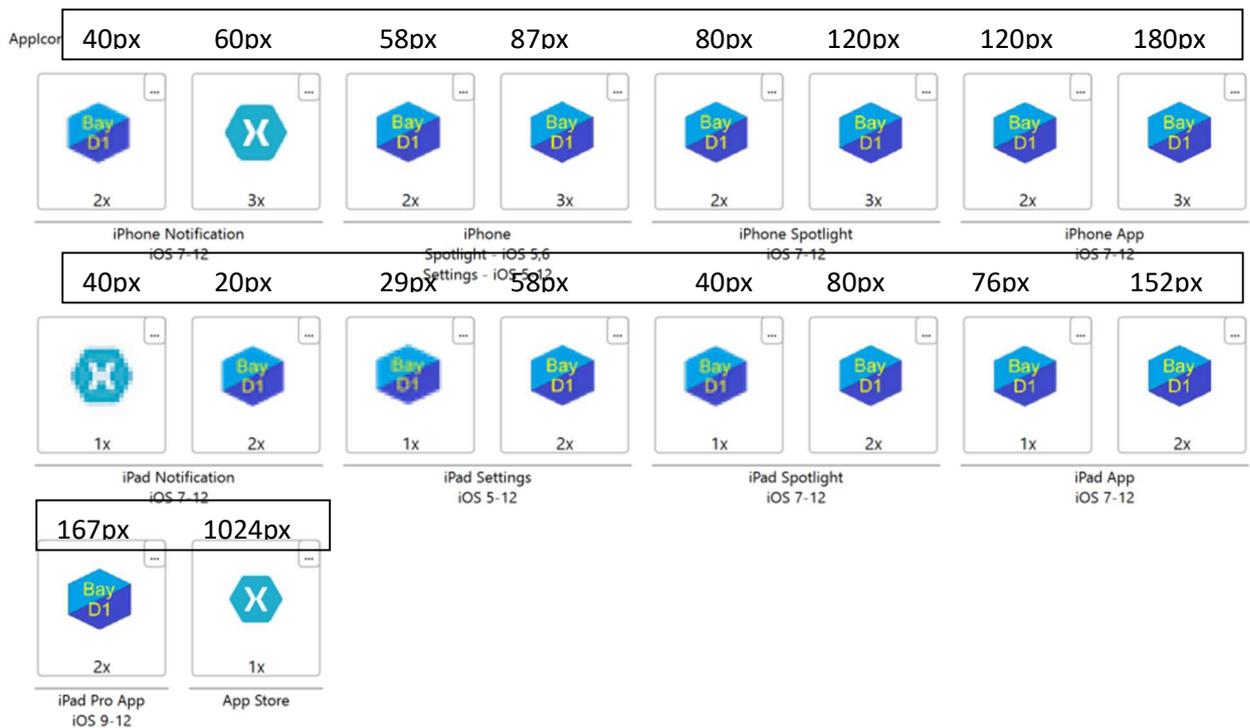
Das sollte man sich nicht mit Hand antun sondern so automatisch machen.

Dann werden diese manuell an die richtigen Stellen im Projekt geladen:

IOS: Im Projektmappen -Explorer auf Assets doppelklicken:

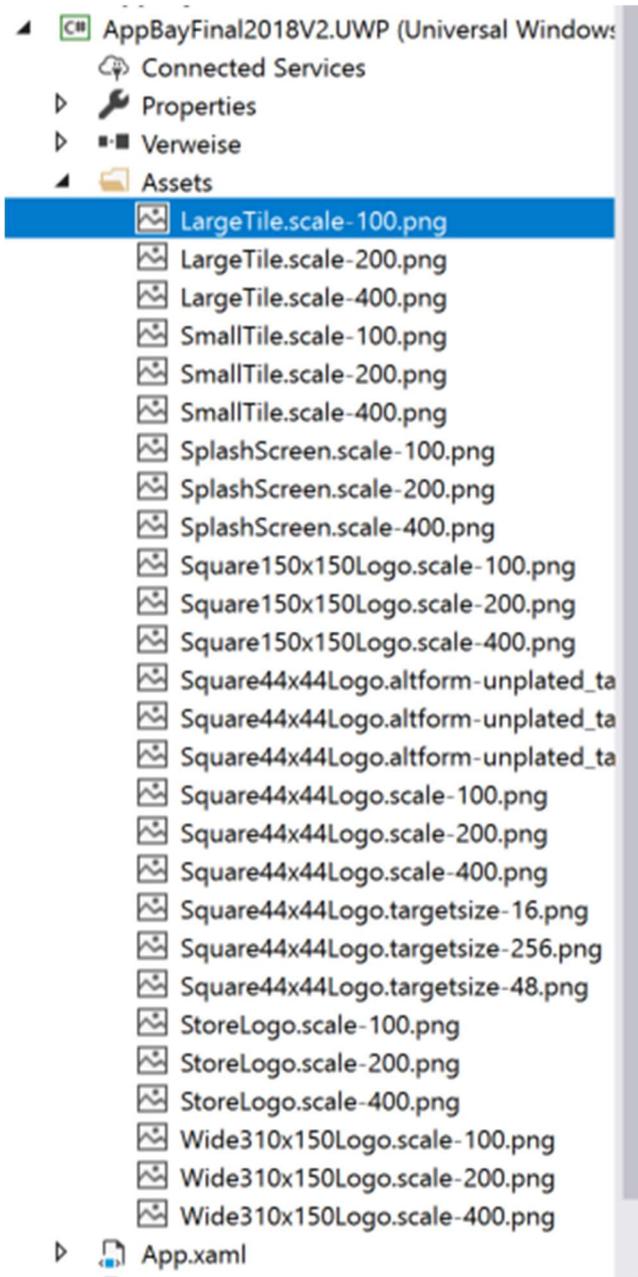


Dann öffnet sich der Editor zum Laden (Feld mit den Punkten) der Icons:



Drei Bilder fehlten, die musste ich nochmal nachproduzieren. Die Größen sind alle quadratisch und in Pixeln angegeben.

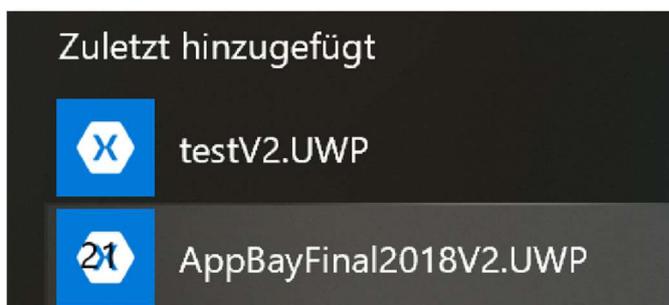
In UWP sind in Assets 27 Bilder zu ersetzen, die in der Größe übereinstimmen sollten.



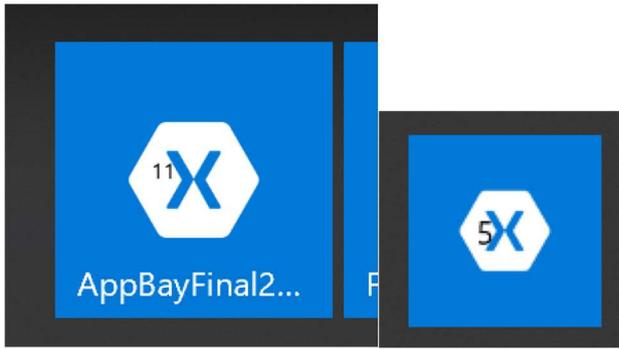
1	310px
2	620px
3	1240px
4	71px
5	142px
6	284px
7	620*300px
8	1240*600px
9	2480*1200px
10	150px
11	300px
12	600px
13	16px
14	256px
15	48px
16	44px
17	88px
18	176px
19	16px
20	256px
21	48px
22	50px
23	100px
24	200px
25	310*150px
26	620*300px
27	1240*600px

Ich habe in die UWP- Bildchen Nummern von 1-27 eingefügt, um zu sehen, wo welches Icon benutzt wird: Beispiele:

In der Startleiste Win10 die Nummer 21:

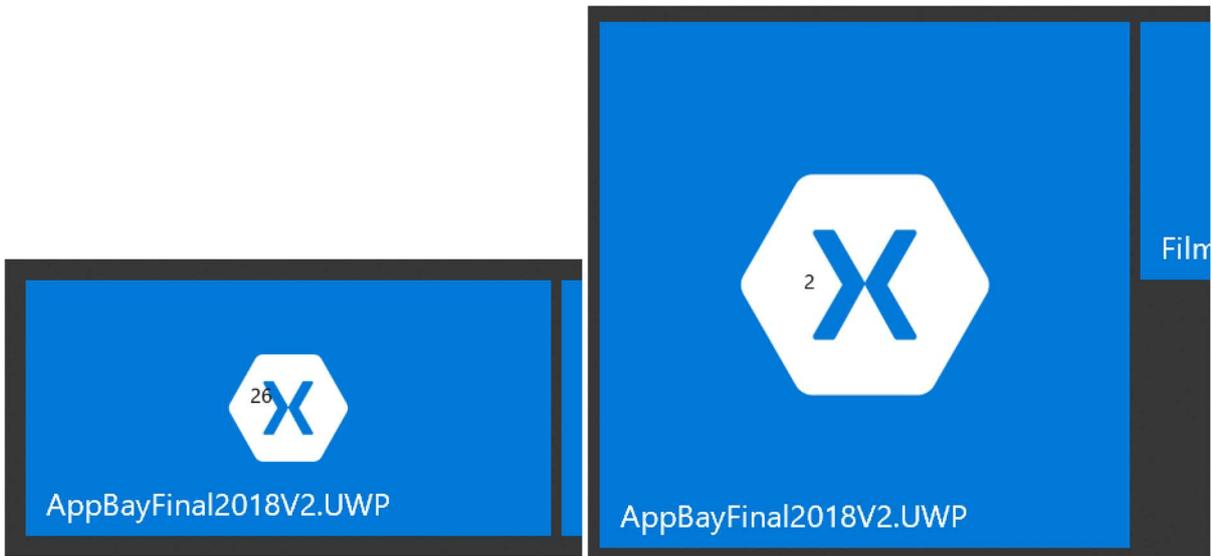


Als „Mitte“-große Kachel 11, klein die 5



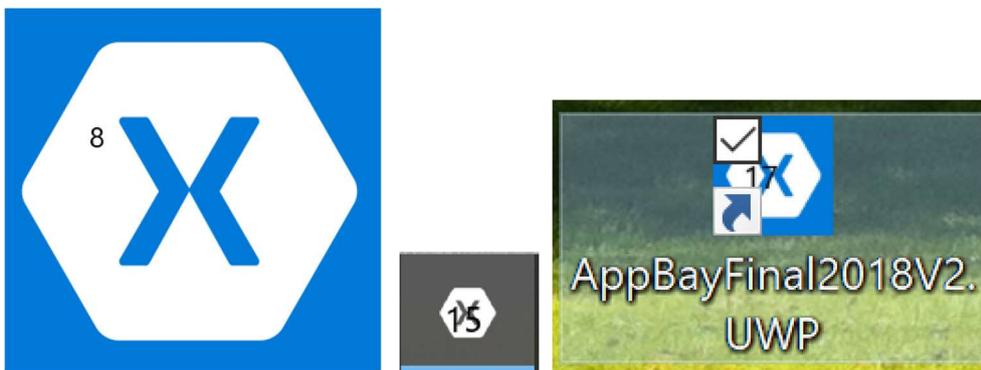
Breit 26 und

Groß die 2



Beim Starten die 8:

in der Taskleiste die 15: Link auf Desktop die 17

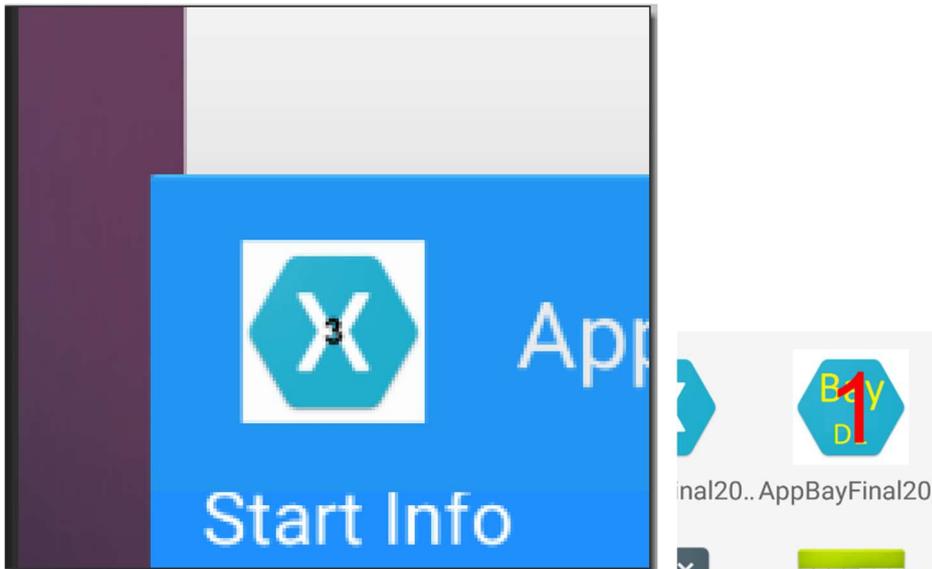


Ebenso in Android, dort in Resources sind 10 Bilder, die ich auch so nummeriert habe:

- ▲  AppBayFinal2018V2.Android
 -  Connected Services
 - ▷  Properties
 - ▷  Verweise
 - ▷  Assets
 - ▲  Resources
 -  drawable
 -  drawable-hdpi
 -  drawable-xhdpi
 -  drawable-xxhdpi
 -  drawable-xxxhdpi
 - ▷  layout
 - ▲  mipmap-anydpi-v26
 -  icon.xml
 -  icon_round.xml
 - ▲  mipmap-hdpi
 -  icon.png
 -  launcher_foreground.png
 - ▲  mipmap-mdpi
 -  icon.png
 -  launcher_foreground.png
 - ▲  mipmap-xhdpi
 -  icon.png
 -  launcher_foreground.png
 - ▲  mipmap-xxhdpi
 -  icon.png
 -  launcher_foreground.png
 - ▲  mipmap-xxxhdpi
 -  icon.png
 -  launcher_foreground.png
 -  xxxHdpi.png

9	72px
10	162px
7	48px
8	108px
5	96px
6	216px
3	144px
4	324px
1	192px
2	432px

Diese findet man dann wieder:

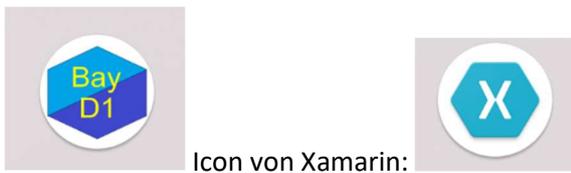


Marshmellow Taskmanager

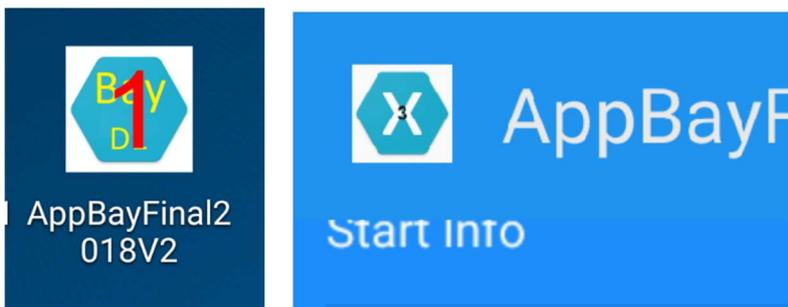
App auf Desktop

Bilder 4 (Taskmanager) und 2 (Desktop) in Android 8 und 9

Bei Android 8 und 9 fügt er das Bild in einen Kreis ein. Zunächst hatte ich mein Bild nicht mittig platziert und es sah doof aus. Aber nach Korrektur ging es dann. Im Emulator:



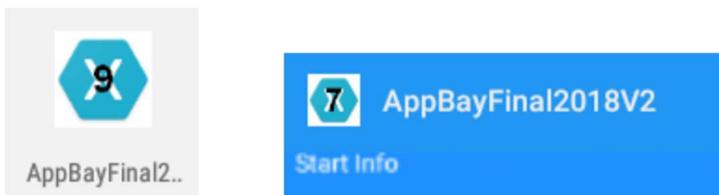
Bei Samsung S4 (Lollipop):



Desktop Nr 1

TaskManager Nr. 3

Bei Nougat 7.1 - Emulator:



Desktop

Taskmanager

Some links to adaptive Icons ab Android 8

<https://forums.xamarin.com/discussion/146458/adaptive-icons-not-working-for-android-in-xamarin-forms-app>

<https://stackoverflow.com/questions/47864285/xamarin-oreo-adaptive-icons>

<https://developer.android.com/studio/write/image-asset-studio#access>

Ab 26.1.2019 ist eine von mir modifizierte Version dieses Resizers zu finden (zip Datei ReSizer-master Bay.zip), die die Arbeit noch einfacher macht und in Android und UWP die Verzeichnisse und Namen gleich richtig erzeugt sowie nun alle ios- Bildchen. Zusätzlich kann man noch eine Farbe wenn nicht schon im Originalbild geschehen als transparent schalten.

Die Background- Icons für Android ab Version 8 habe ich allerdings noch nicht gelöst.

Eine sehr gute Seite für Design Richtlinien aller Plattformen ist diese:

<https://material.io/design/introduction/#>

Design von der Android- Seite: <https://developer.android.com/training/multiscreen/screendensities>

<https://developer.android.com/studio/write/image-asset-studio>

Geht wohl nur mit Android Studio, da werden adaptive icons erzeugt.:

<https://developer.android.com/studio/>

Bayerlein

26.1.2019