

# **Watch face erstellen**

## Kurzbeschreibung

### **I. So sollte man anfangen :**

1. Installation der [Watchfaces app](#) auf der Pine und die app starten. Diese app ist eine Evaluierungsversion, die man testen und verwenden kann, in der sich aber alle 15 Sekunden die Uhr ändert. Mit einem Doppelklick auf dem Display kommt man in die Settings und kann per Buyapp die app kaufen. Nach dem Kauf kann man den Parameter ( alle 15 Sekunden neue watch ) abwählen und hat auch einen developer mode ( der aber auf der pine nicht richtig funktioniert siehe Punkt C.ii.)).
2. Nach der Installation kommt die komplette Grafik und Uhrenlogik von der APK selbst.. Wir werden jetzt das Aussehen ändern. Als erstes mit dem usb Kabel die pine mit dem PC verbinden Dann das [ZIP archive of the project](#) herunterladen und unzippen. Kopieren des Watchfaces Ordners zum top-level Directory der SD card Ordner auf der Pine, die Struktur muss dabei erhalten bleiben.

Achtung : Das gemountete device sollte das erste auf dem windows PC sein(598 MB). Falls man den sd swap gemacht hat, ist das natürlich die ( 16/32/64 GB ) SD Card. Die sd card auf der pine sollte dann einen Ordner watchfaces enthalten, inclusive aller Beispiel Ordner mit watchfaces und inclusive Unterordner example , welcher das Tutorial watchface enthält.

3. Tutorial jetzt durchführen, um die grundsätzlichen Dinge zu verstehen :  
<https://github.com/kurthuwig/faces/blob/master/docs/tutorial.md>  
und einige Änderungen an dem Beispiel vorzunehmen.

### **II. Eigenes watchface bauen:**

1. png Dateien erstellen ( Hintergrund ( kann auch jpeg Format haben), Zeiger, Symbole, Zahlen, Buchstaben )
2. Font erstellen ( für die Darstellung von Zeichen, Zahlen, ...) mit Tool gdx\_fontpacker
3. alle Dateien in ein leeres Directory ( auf dem PC )
4. tool GDX\_Texture:packer zum pack aufrufen und packen
5. \*.pack in \*.atlas umbenennen
6. \*.lua editieren
7. Pine mit PC verbinden ( usb mass storage )
8. komplettes directory nach /watchfaces auf der sd card kopieren
9. watch app stoppen und neu starten ( vorher das usb Kabel entfernen)
10. Ergebnisse auf der pine prüfen

### **III. Links :**

Tutorial :

*<https://github.com/kurthuwig/faces/blob/master/docs/tutorial.md>*

Tool zum Packen : GDX Texture Packer :

*<https://code.google.com/p/libgdx-texturepacker-gui/>*

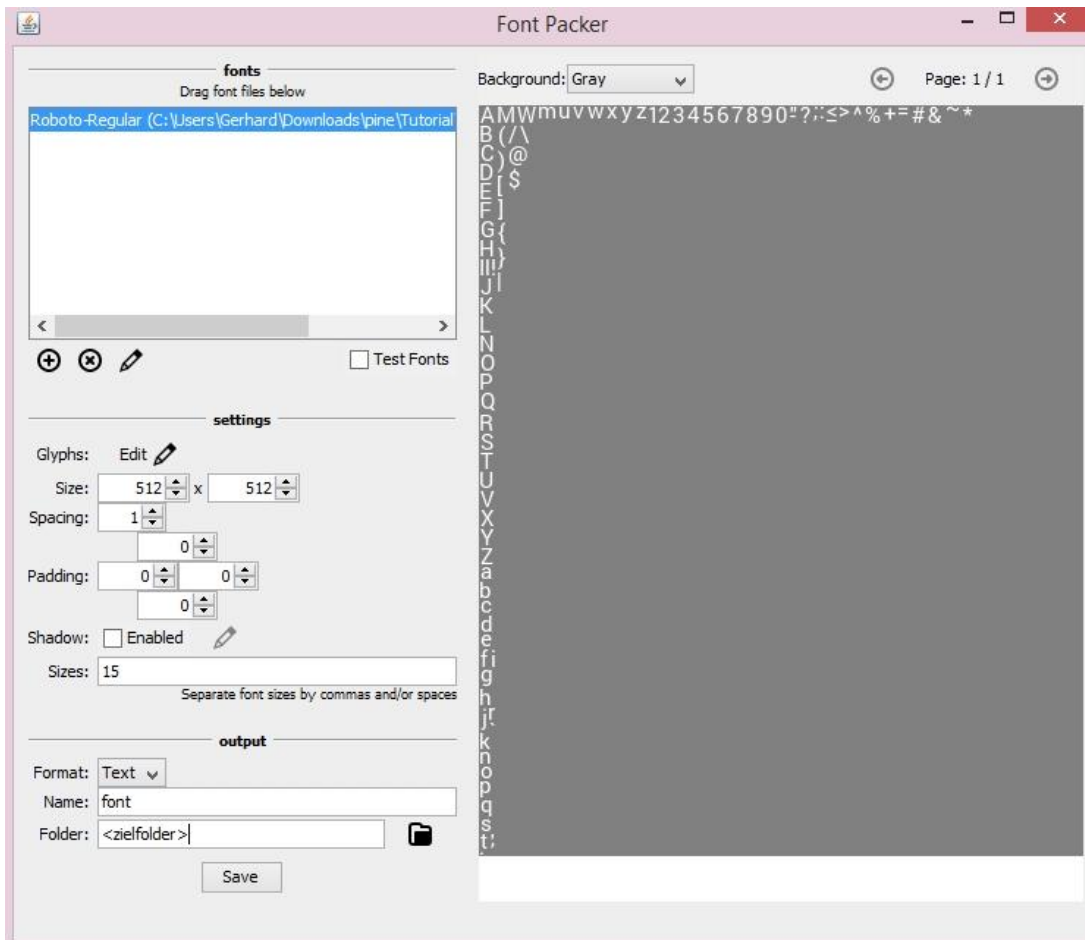
Tool zum Erstellender fonts : gdx\_font\_packer :

*<https://github.com/mattdesl/gdx-fontpack/blob/master/README.md>*

## IV. Tipps :

### A. Zu 2: gdx\_font\_packer :

Programm starten



Aus dem Internet geladene „Roboto-Regular.ttf“ ( hier als Beispiel ) in das obige Fenster ziehen ( drag+drop) und per Edit ( falls nötig ) bearbeiten( löschen oder hinzufügen). Hinweis : ich musste das Grad zeichen für Grad Celsius dazutun.

Unter sizes: die anzuzeigenden Formate eingeben ( hier nur eins: 15)

Dann unter output : Parameter = „Text“ setzen und den Namen und das Zieldirectory eingeben : hier : <name> = font und <Zielfolder> =Projektfolder ( z.B. ../../../pine3

Save

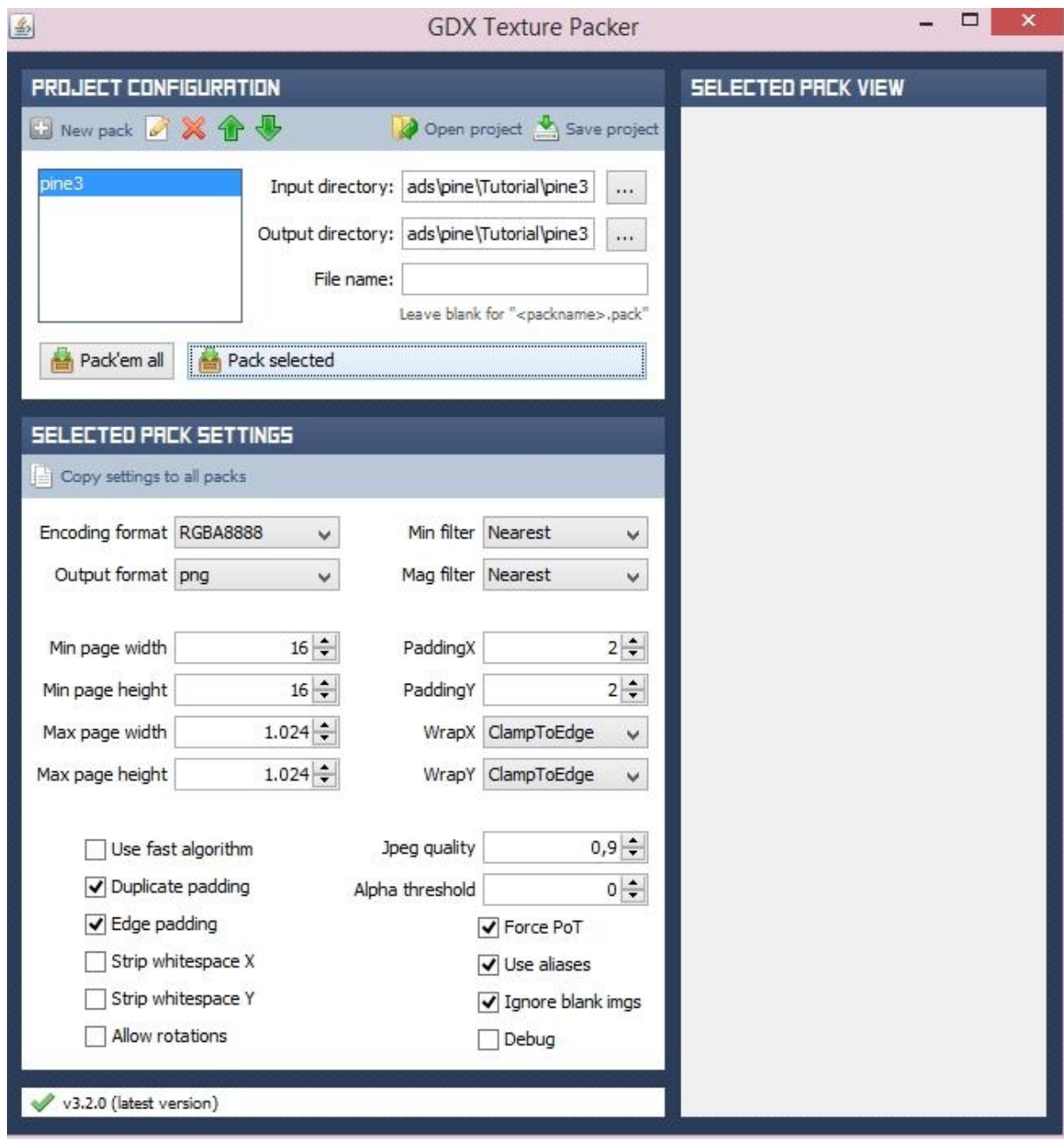
Damit werden 2 Dateien erstellt :

- font.png* : die Datei, die verwendet werden soll um Zeichen darzustellen
- roboto-regular15.fnt* : die Datei, die die Infos enthält , am welcher Stelle die Zeichen zu finden sind.

Beide Dateien jetzt ins Projektverzeichnis kopieren. Sie gehören immer zusammen!

Hinweis : hat man mehr als ein size ausgewählt ( zb. 15,21), um in unterschiedlicher Grösse ( 15 und 21 )anzuzeigen, werden zwei fnt-Dateien erstellt, die man dann ins ZielDir kopieren muss.

## B. Zu 4. Gdx\_texture\_packer :



- a. mit +New pack neues Packet einrichten :  
Name vergeben : z.B. pine3  
Input und output directory auswählen ( das Projektdir )
- b. pack selected wählen

nun wird ein entsprechendes Paket und eine entsprechende <name>.png ( in der alle \*.png enthalten sind ) erstellt. Die \*.pack umbenennen in \*.atlas , ----- fertig ----

## C. Allgemein :

- i. Will man ein **neues Projekt** aus einem vorherigen erstellen, dann daran denken:

In der face.lua folgende Zeilen ändern :

- a. Name + Author
- b. <name>.atlas in der ersten Zeile der Init Funktion anpassen

### ii. **Reload Button : funktioniert auf der pine nicht !!!!!!!!!!!**

Um einen reload durchzuführen bitte die app beenden  
Settings/apps/manage apps : app suchen und beenden  
Und wieder neu starten. Damit werden die Änderungen gültig.

**Mein Tipp:** Folgende Schritte :

- a. Auf dem PC die Projekte bearbeiten, bis alles fertig ist, dann
- b. die pine mit dem PC per USB verbinden ( usb mode ) und
- c. das Projekt auf die sdcard unter watchfaces kopieren.
- d. USB- verbindung trennen !!!! Wichtig, da sonst ein Test unmöglich oder fehlerhaft !
- e. Testen