L2.2 Ereignisse (Events)

2.2.1 Ereignisse in Kivy

In Wörterbüchern sind Ereignisse (Events) beschrieben als „etwas, das passiert, insbesondere etwas wichtiges“. Diese Beschreibung passt auch gut auf Ereignisse in Kivy. Kivy feuert ständig und zu jeder Zeit Ereignisse. Wir konzentrieren uns jedoch lediglich auf diejenigen, die wir als wichtig erachten. Kivy stellt einen Event Loop zur Verfügung, der dauerhaft durch die run()-Methode in der App ausgeführt wird. Wenn etwas interessantes passiert, dann wird das notwendige getan, damit der eigene Code merkt, dass das Ereignis stattgefunden hat und über einen Event-Handler die Chance zu reagieren. In Kivy ist ein Event-Handler einfach eine Funktion oder Methode.
Bei Schaltflächen (Buttons) ist on_press ein mögliches Ereignis. Dieses wird ausgelöst, sobald die Schaltfläche mit der Maus oder einer Taste (oder dem Finger) gedrückt wird.

2.2.2 Event on_press in der kv-Datei

Möchte man eine Reaktion erhalten, wenn auf einen Button geklickt wird, muss in der kv-Datei das Attribut *on_press* ergänzt werden. Der zugewiesene Wert muss der Name der Methode (hier: *antworten_gut()*) in der Python-Datei sein.

Aufbau der Datei main.kv:

```
1 Label:
2     id: label
3     text: "Wie geht es dir?"
4 Button:
5     id: button1
6     text: "Gut"
7     font_size: 24
8     on_press: root.antworten_gut()
```

main.kv

Zeile 8: Sobald das Ereignis on_press ausgelöst wird, wird die Funktion antworten_gut() aufgerufen, die in der übergeordneten Klasse definiert ist.

2.2.3 Ereignisverarbeitung in der Python-Datei

Aufbau der Datei main.py:

```
1 …
2 class MyGridLayout(GridLayout):
3     def antworten_gut(self):
4         self.ids.label.text = "Das freut mich!"
```

main.py

Zunächst wird die entsprechende Methode *antworten_gut()* innerhalb der MyGridLayout-Klasse definiert. Möchte man innerhalb dieser Methode andere Widgets aus dem GridLayout ansprechen und ggf. verändern, so müssen diese Widgets vorher in der .kv-Datei eine ID erhalten haben. Das Widget lässt sich dann folgendermaßen ansteuern:
`self.ids.`*`id_des_widgets.attribut_des_widgets`* `= …`