

Tutorial

IFTTT-fähige LED mit der JTL-Wawi steuern

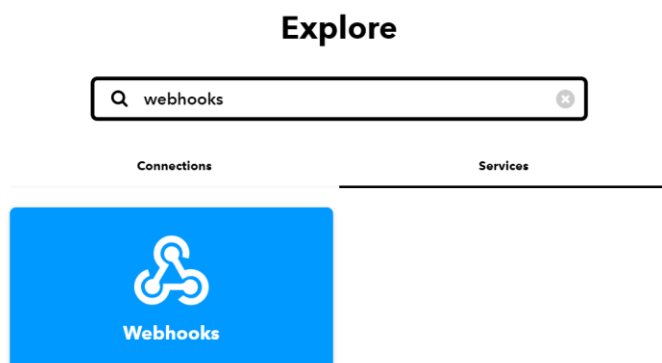
Das Szenario dieses Tutorials ist denkbar einfach: immer, wenn eine offene Amazon Prime Bestellung in der Wawi-Datenbank ist, soll eine Lampe im Lager rot leuchten, damit niemand diese priorisierte Bestellung übersieht.

Teil 1: IFTTT

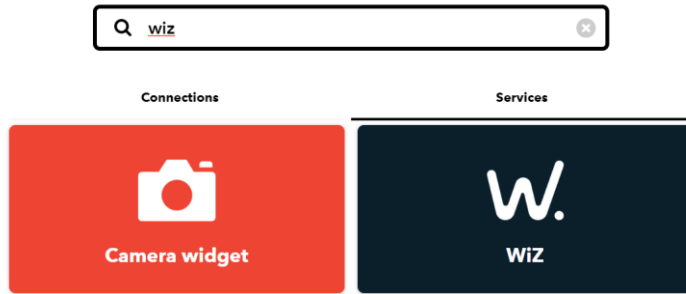
1. IFTTT-fähige Lampe besorgen. Bekommt man in jedem gut sortierten Baumarkt oder im Internet. Für dieses Tutorial verwenden wir eine WiZ LED A60 E27.
Ggf. Fassung und Kabel mit dazu kaufen.

Klinkt die LED in Euer W-Lan ein und installiert Euch die zur Lampe gehörende App auf Eurem Smartphone.

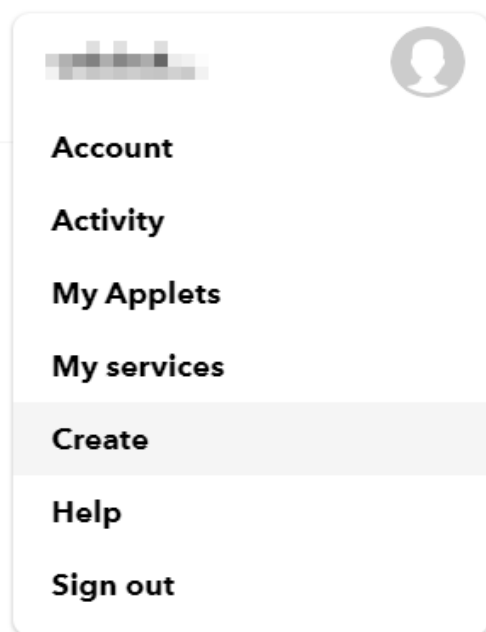
2. Auf www.ifttt.com einen Account anlegen.
3. In der App der LED-Lampe IFTTT autorisieren (in der WiZ-App unter Einstellungen)
4. Auf ifttt.com rechts oben auf „Explore“ klicken. Gebt „Webhooks“ in das Suche-Feld ein und klickt darunter auf „Services“. Durch einen Klick auf das Webhooks-Icon aktiviert Ihr den Service.



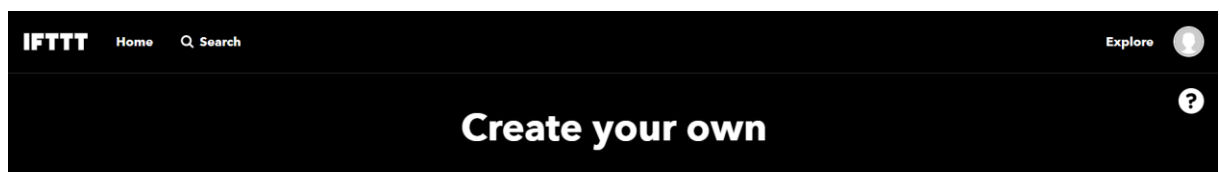
5. Statt webhooks sucht Ihr nun nach dem Hersteller Eurer LED, in unserem Fall also nach „WiZ“.



6. auf ifttt.com rechts oben neben Explore auf das Profil-Icon klicken -> Create

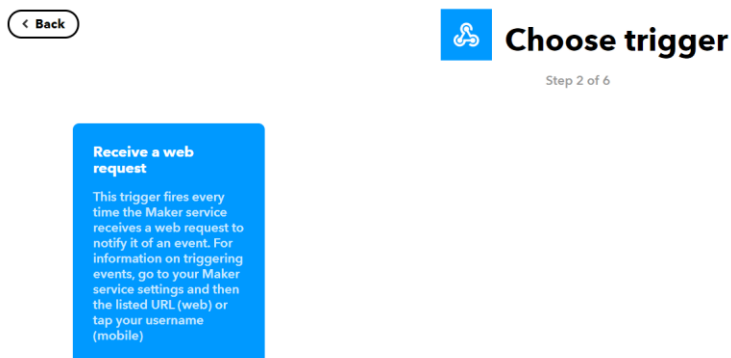
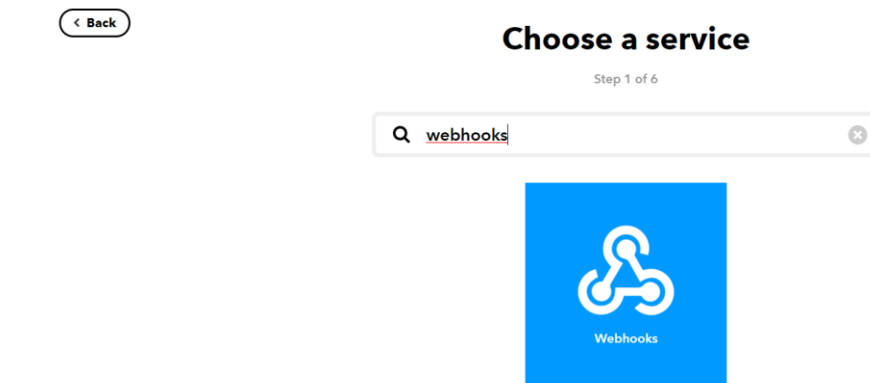


7. Beim "If + This Then That" auf das + klicken

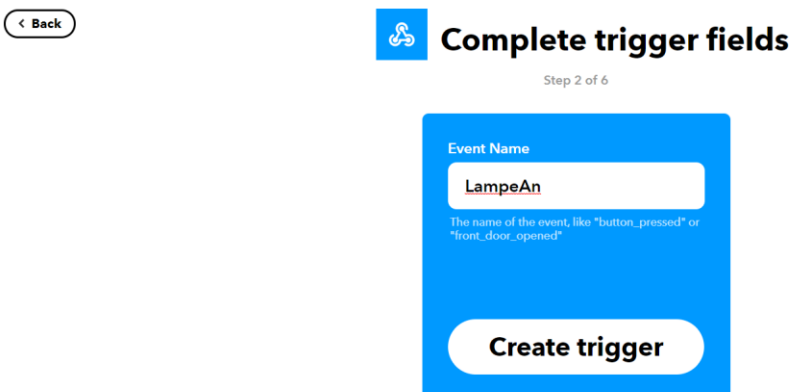


If This Then That

8. nach webhooks suchen und draufklicken, danach auf „receive a web request“.



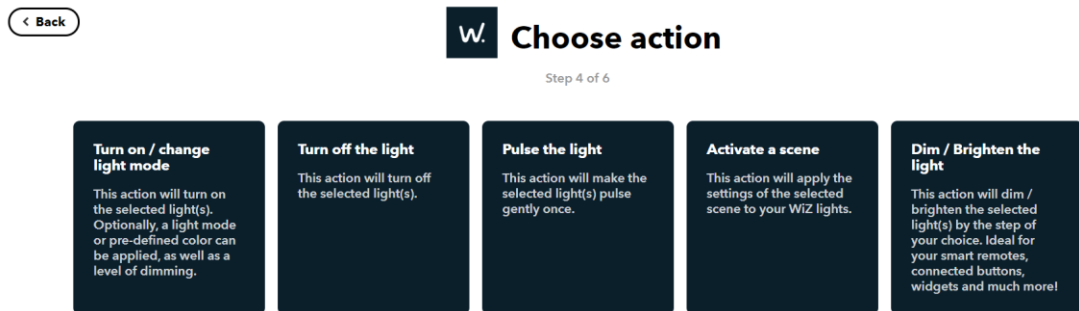
9. Vergebt einen Namen für das Event, das Ihr anlegen wollt. Zum Beispiel „LampeAn“. Verwendet hier keine Leerzeichen und merkt Euch den Namen des Events, denn den brauchen wir später.



10. Jetzt auf das + zwischen Then und That klicken



11. sucht nach der App für Eure Lampe, klickt darauf und die möglichen Aktionen für die Lampe erscheinen. Klickt darauf und wählt dort die Aktion, die Ihr wünscht. Wir klicken in diesem Fall auf „Turn on/change light mode“.



12. Im folgenden Dialog wählt Ihr die betreffende LED aus (alternativ können auch ganze Gruppen an- und ausgeschaltet werden), die Farbe und wie hell die LED leuchten soll. Klickt abschließend auf „create action“.

W. Complete action fields

Step 5 of 6

Which light(s)?

Prime Lampe (Lager) ▼

This can be the whole home, a room, a group, or a light.

What light mode / color?

Red ▼

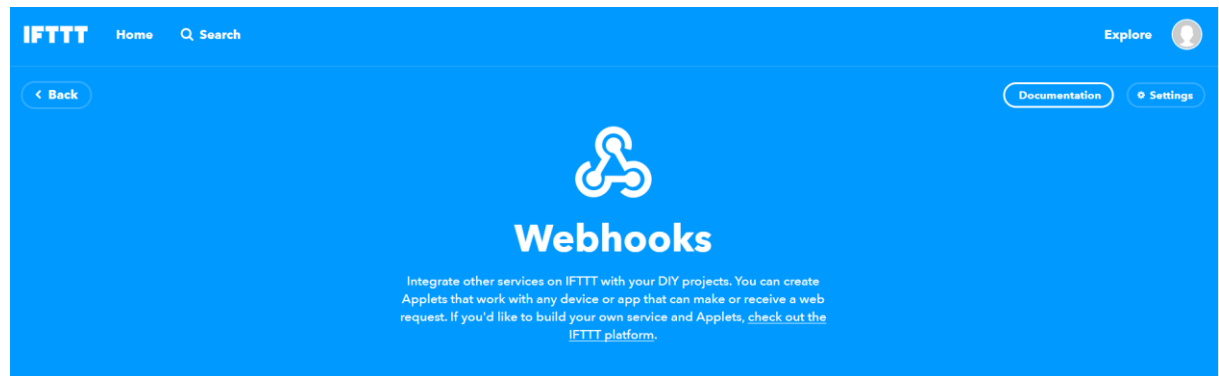
What brightness?

100% ▼

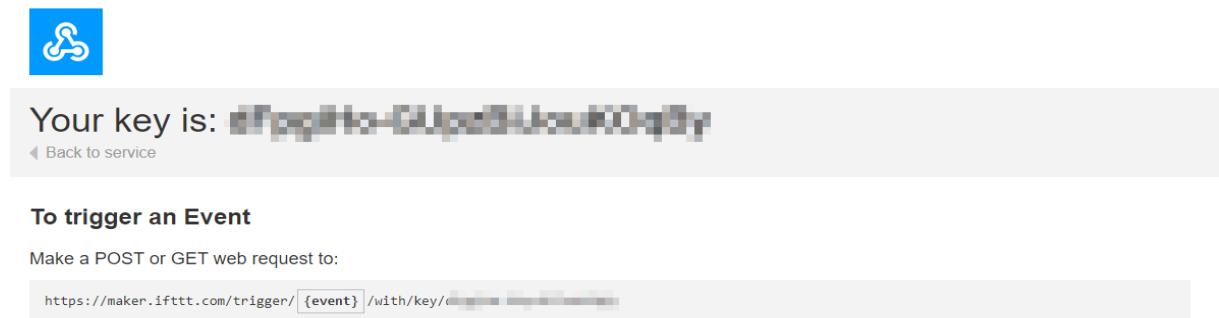
Select the brightness level of your choice.

Create action

13. Im letzten Schritt könnt Ihr noch eine Beschreibung vergeben. Benachrichtigungen solltet Ihr abschalten, ansonsten bekommt Ihr jedes Mal eine Nachricht, wenn das Event ausgelöst wird. Klickt dann auf „Finish“.
14. Nun benötigt Ihr die Trigger-URL, mit der Ihr das Event auslösen könnt. Klickt rechts oben wieder auf Euer Profil-Icon -> my Services -> Webhooks -> und rechts auf „Documentation“.



15. Wir benötigen die url in der ersten Zeile:



Bei Event tragt Ihr den Namen der auszulösenden Aktion ein, in unserem Fall also „LampeAn“.
Um unsere Lampe anzuschalten, sollte Eurer URL also so aussehen:

`https://maker.ifttt.com/trigger/LampeAn/with/key/$euergeheimerKey$`

16. Kopiert Euch obige URL in Euren Browser und drückt Enter. Eure Lampe sollte angehen.
Woohoo! Hexenwerk!

17. Jetzt wiederholt Ihr das für weitere gewünschte Aktionen, z.B. Lampe aus.

Teil 2 – Workflow basteln

Falls Ihr noch nie mit Workflows gearbeitet habt: RTFM! :-)

<https://guide.jtl-software.de/jtl-wawi/jtl-workflows/einleitung-zur-workflow-verwaltung/>

Insbesondere auch, wie erweiterte Eigenschaften angelegt werden.

1. Erweiterte Eigenschaft „PrimeOffen“ anlegen.

```
{% assign primeoffen = false %}\n{% assign result = \"SELECT COUNT(cBestellNr) FROM tBestellung AS t1\nWHERE [nKomplettAusgeliefert]='0' AND [nStorno]='0' AND\n[kRueckhalteGrund]='0' AND [tVersandArt_kVersandArt] = (SELECT\nkVersandArt FROM tversandart WHERE cName = 'Amazon Prime')\" |\nDirectQueryScalar -%}\n{%if result > 0 -%}\n{% assign primeoffen = true %}\n{% endif -%}\n{{ primeoffen }}
```

Dieser Code schaut direkt in der Datenbank nach, ob es nicht ausgelieferte, nicht stornierte und nicht zurückgehaltene Aufträge mit der Versandart „Amazon Prime“ gibt. Sollte Eure Versandart anders heißen, müsst Ihr sie im Code natürlich entsprechend ändern.

Je nachdem gibt der Code nun ein „wahr“ oder „falsch“ zurück.

Den folgenden Code könnt Ihr natürlich nach Belieben modifizieren, z.B. direkt die ID der Versandart hinterlegen oder auch nur die Anzahl der Bestellungen ausgeben lassen und diese statt dem Boolean auswerten.

2. Legt Euch nun einen Workflow unter Auftrag -> Erstellt wie folgt an:

The screenshot shows the 'Workflowverwaltung' (Workflow Management) interface. The main window is titled 'JTL-Workflows anlegen / bearbeiten'. The 'Aufträge' (Orders) tab is selected. The workflow being edited is named 'Amazon Prime Lampe'. The 'Bedingungen' (Conditions) section is active, showing a condition: 'offenePrime gleich True'. The condition is configured with the property 'Erweiterte Eigenschaften/offenePrime' set to 'Gleich' and the value 'True'. The 'Aktionen' (Actions) section shows a 'Web-Request (POST)' action with the URI 'https://maker.fttt.com/trigger/LampeAn/with/key/'. The 'Bei Fehler Workflow abbrechen' checkbox is checked. The 'Workflow testen' (Test Workflow) button is visible at the bottom.

Als Bedingung soll der Wert PrimeOffen wahr sein, also wenn offene Prime-Bestellungen im System sind.

Als Aktion legt Ihr einen Web-Request (POST) an und tragt dort Eure Trigger-URL ein.

3. Wiederholt 2. entsprechend für Euren Lampe-AUS-Trigger
4. Die Workflows sollten ebenso unter „Geändert“ und „Ausgeliefert“ erstellt werden, da die Lampe ansonsten nur bei neuen Aufträgen getriggert wird.

Und das war es schon. Viel Spaß!

Matthias „Matze“ Vogel

AIMPLIFY Distribution Inc.
Erfurter Str. 22
99706 Sondershausen
Deutschland
mv@aimplify.com