

# Achievements

Nach den "Complaints" dass ich die Doku auf Englisch gemacht habe (beim Kalender) hier eine Deutsche

zum Achivement

## Wie man achievements macht:

Erstelle ein "meta\_achievements.py" file. Dies ist wo du alle achievement klassen definierst. Jedes Achiement ist eine unterklasse von "achivements.MetaAchievment"

### Beispiel

```
import achievements
from hasi.models import UserProfile

class Autobiographer(badges.MetaAchievement):
    id = "autobiographer"
    model = UserProfile
    one_time_only = True

    title = "Autobiographer"
    description = "Completed the User Profile"

    progress_start = 0
    progress_finish = 2

    def get_user(self, instance):
        return instance.user

    def get_progress(self, user):
        has_email = 1 if user.email else 0
        has_bio = 1 if user.get_profile().bio else 0
        return has_email + has_bio

    def check_email(self, instance):
        return instance.user.email

    def check_bio(self, instance):
        return instance.bio
```

Das achievement wird über post\_save signal awarded. Deshalb, wenn eine teilnehmer des models (in diesem fall UserProfile) gespeichert wird wird überprüft ob der user ein achievement awarded wird.

Wenn ein profil (in diesem beispiel) gespeichert wird überprüft die app die konditionen und wenn alle wahr sind dann wirds awardet

Die "get\_user"methoder methode wird die gleiche instanz gepasst wie in den konditionen und muss den benutzer zurückgeben der den award "gewinnt". (Einfach so lassen)

Die "one\_time\_only" attribut ist da um zu prüfen ob der user mehr als einmal gewinnen kann.

Alle mehtoden mit check sind checks!

## Signals

Wenn ein achievement awarded wird wird ein signal gesendet (in signals.py). Das "sender" keyword argument is das metaAchieement das du definiers (Autobiographer, und nicht die achievement model instance. "user" keyword argument wer sie bekommt und das "achievement" keyword argument is die model instance des achievements ind der Datenbank.