



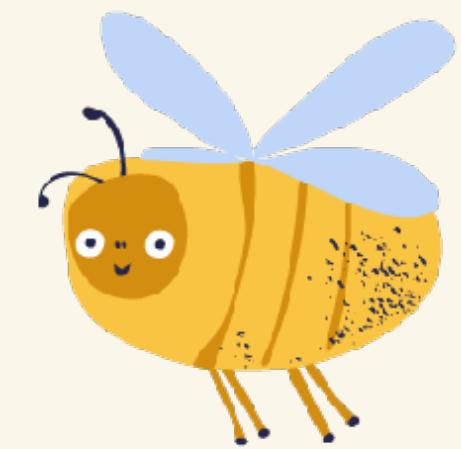
# bloom

## Automatisiertes Bewässerungssystem

11.02.22 // Berliner Hochschule für Technik

Albert Kaminski  
Dominik Domonell  
Jelena Mirceta  
Sahiram Ravikumar  
Simon Aschenbrenner

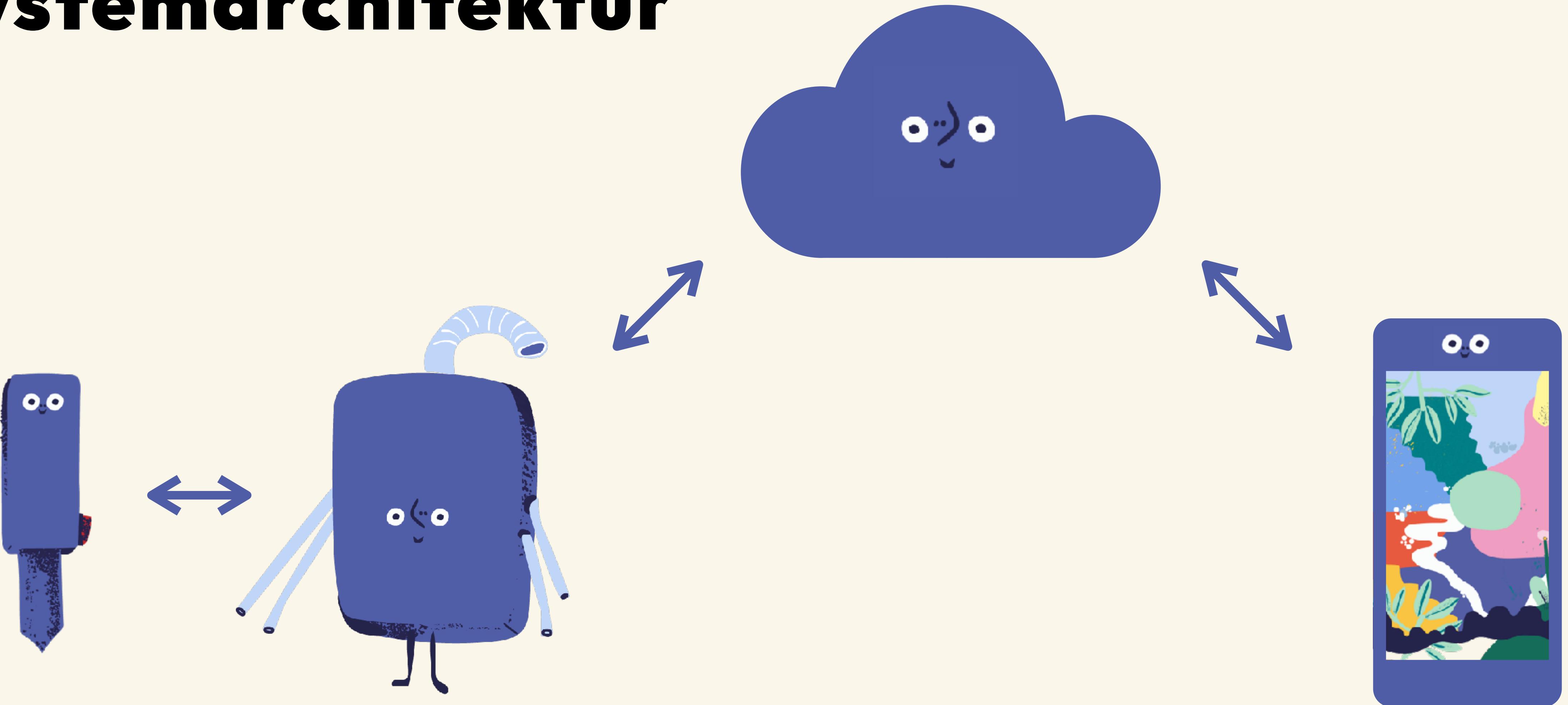
# Bloom



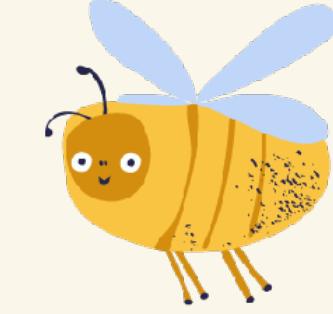
- Unzählige Pflanzen gehen jedes Jahr ein, weil ihre Besitzer\*innen im Urlaub sind oder sie wegen mangelnder Fachkenntnis zu wenig oder zu viel gegossen werden
- Unser benutzungsfreundliches System kann mithilfe von Magnetventilen und drahtlosen Feuchtigkeitssensoren eine optimale und ressourcenschonende Bewässerung aller Pflanzen sicherstellen
- Mit einer Android-App können Nutzer\*innen die Bewässerung überwachen und steuern
- Skalierfähiges System – Für das Projekt Fokus auf den Use Case Balkonbewässerung



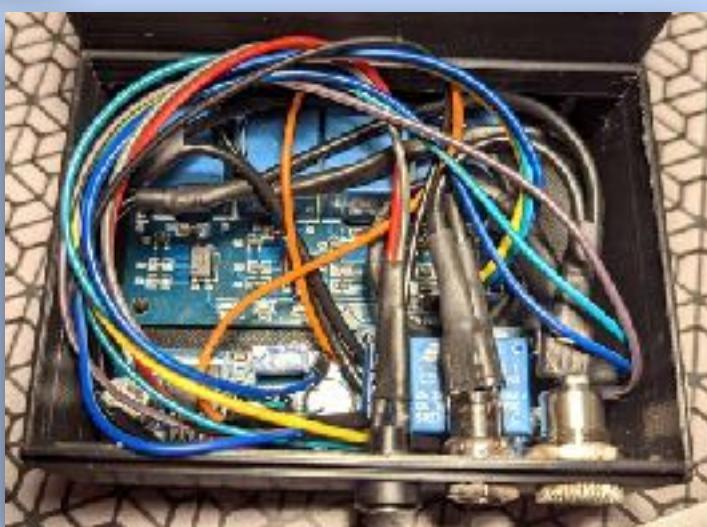
# Systemarchitektur



# Bloom Box



- Die Bloom Box ist eine 3D-gedruckte Box, die Hardwarekomponenten zur Steuerung des Gießvorgangs beinhaltet
- Dazu gehören: Microcontroller (Hub), ein über Relais steuerbares Magnetventil mit 4 Ausgängen, Spannungswandler, Pumpe sowie ein Wasserstandssensor
- Außerdem besitzt die Bloom Box die Möglichkeit über ein Display und 2 Knöpfe Funktionen abzulesen und zu bedienen
- Alle angeschlossenen Geräte können an- und abgesteckt werden
- Auf der Oberseite, sowie an den Seiten befinden sich hilfreiche Beschriftungen



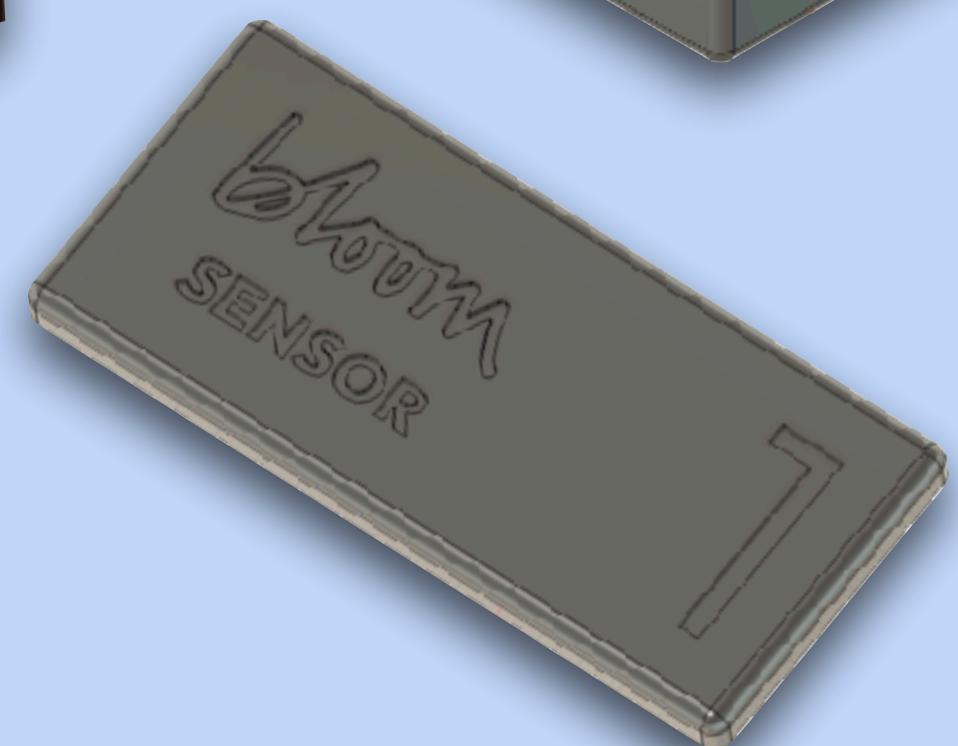
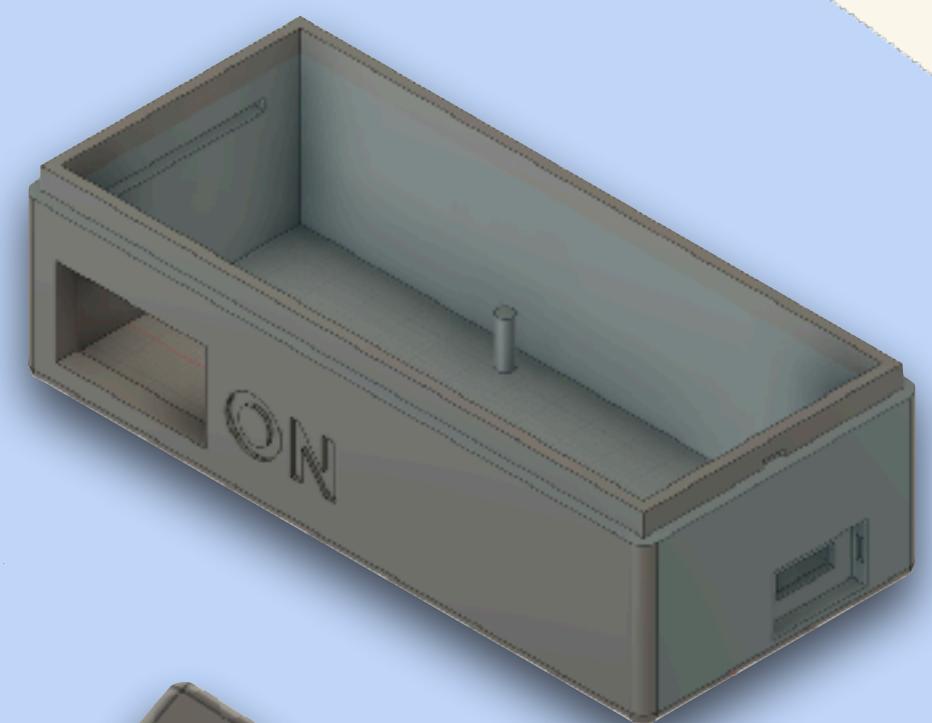
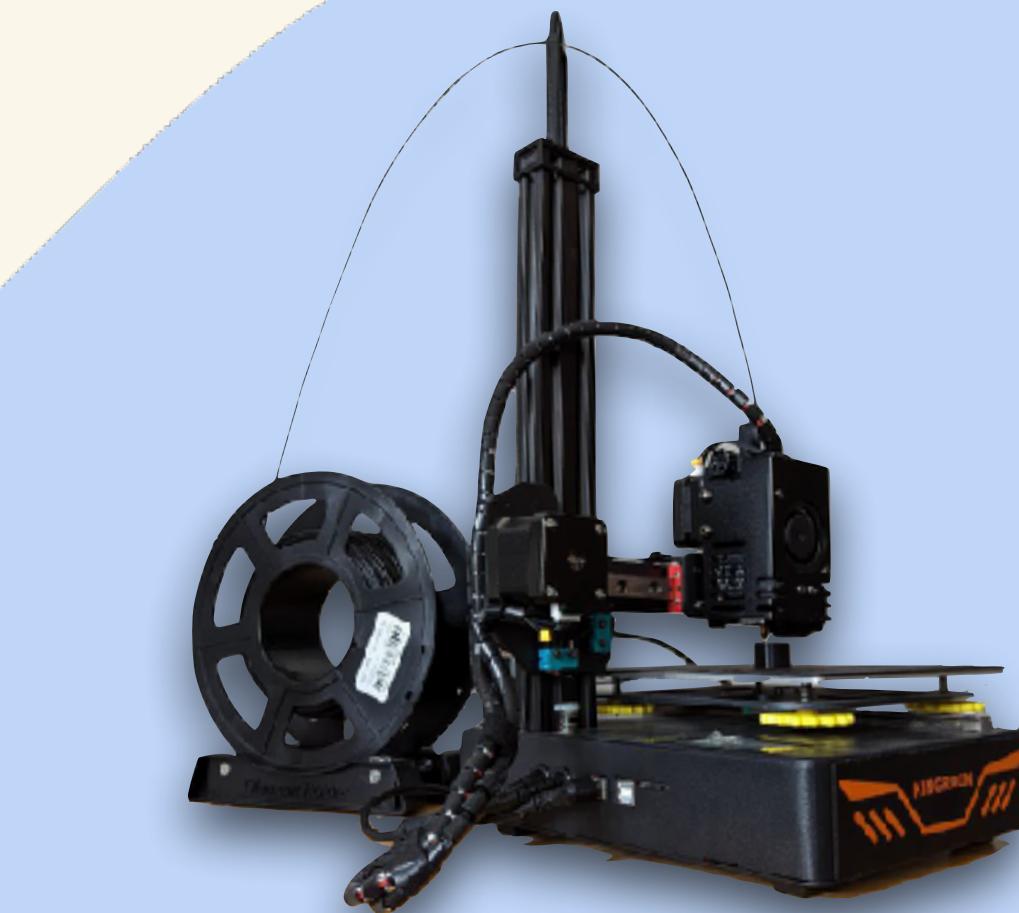
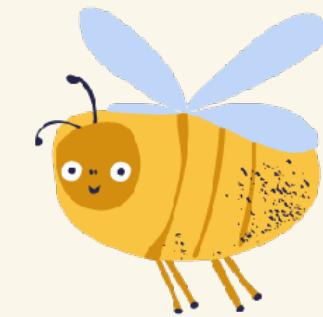
# Bloom Sensor



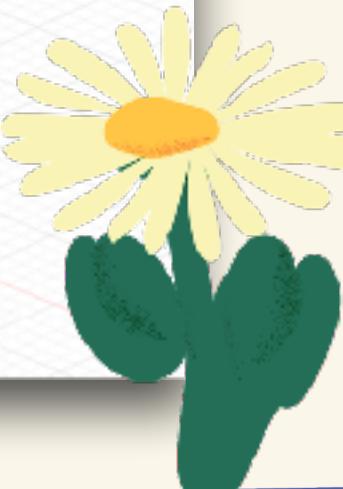
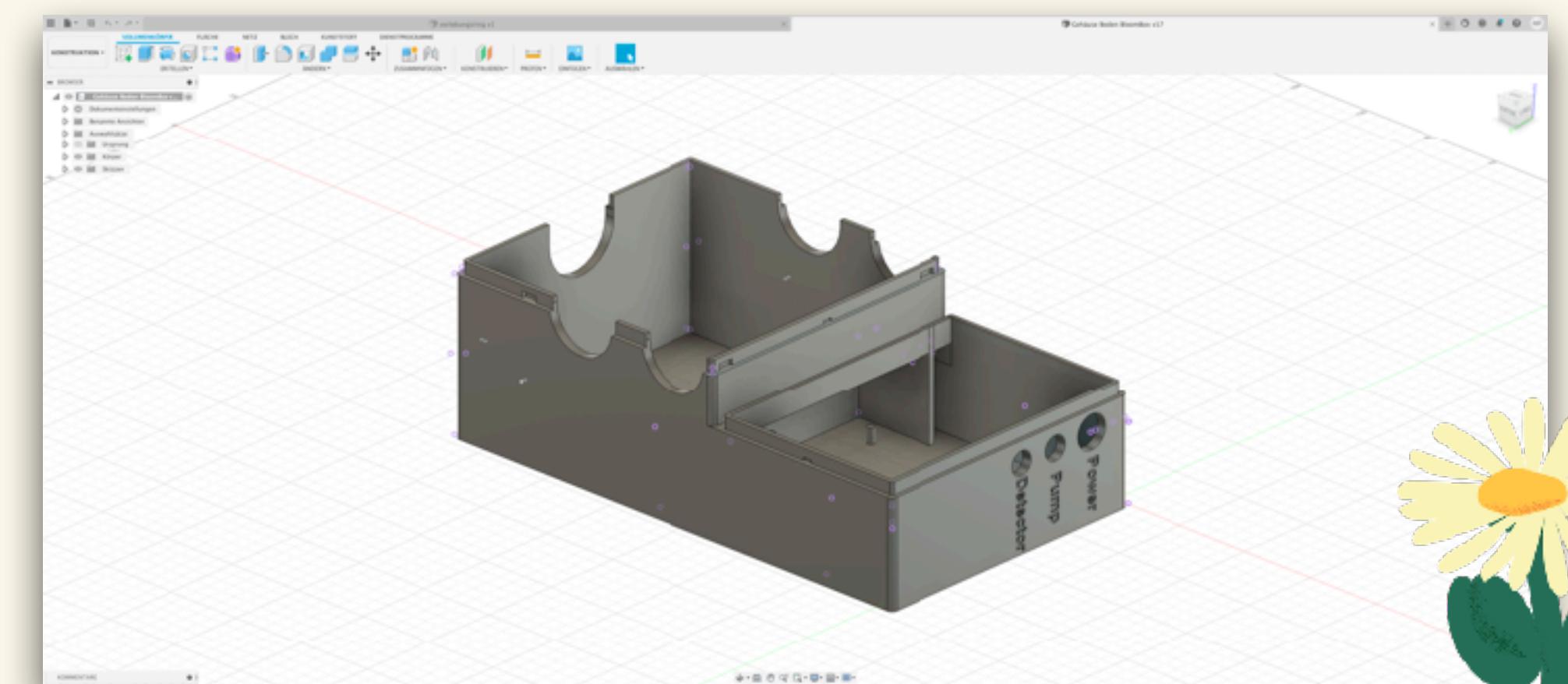
- Der Bloom Sensor misst die Feuchtigkeit in der Erde und gibt diesen Wert an die Bloom Box weiter
- Er beinhaltet einen Micro-Controller mit LoRa-Modul, den Feuchtigkeitssensor und einen wiederaufladbaren Akku mit 9 Monaten Akkulaufzeit
- Außerdem einen Schalter zum an-und ausschalten, sowie einen USB-Anschluss
- Das Gehäuse ist zudem Spritzwassergeschützt
- Auf dem Deckel, sowie an den Seiten befinden sich hilfreiche Beschriftungen



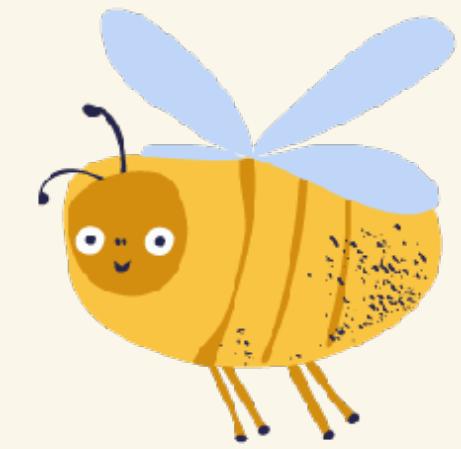
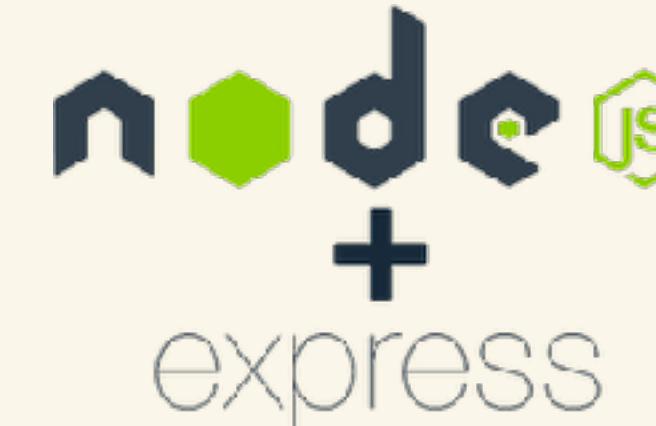
# 3D-Drucker (Kingroon KP3S)



- 3D-Druck ist ein Druckverfahren, bei dem Material in Schichten aufgetragen wird und somit ein dreidimensionales Objekt erzeugt wird.
- PLA (polylactic acid) ist ein stabiler Kunststoff aus nachwachsenden Stärke-Rohstoffen (zersetzt sich nach 80 Jahren)
- Mit einer CAD-Software wird der Körper konstruiert (Fusion360)



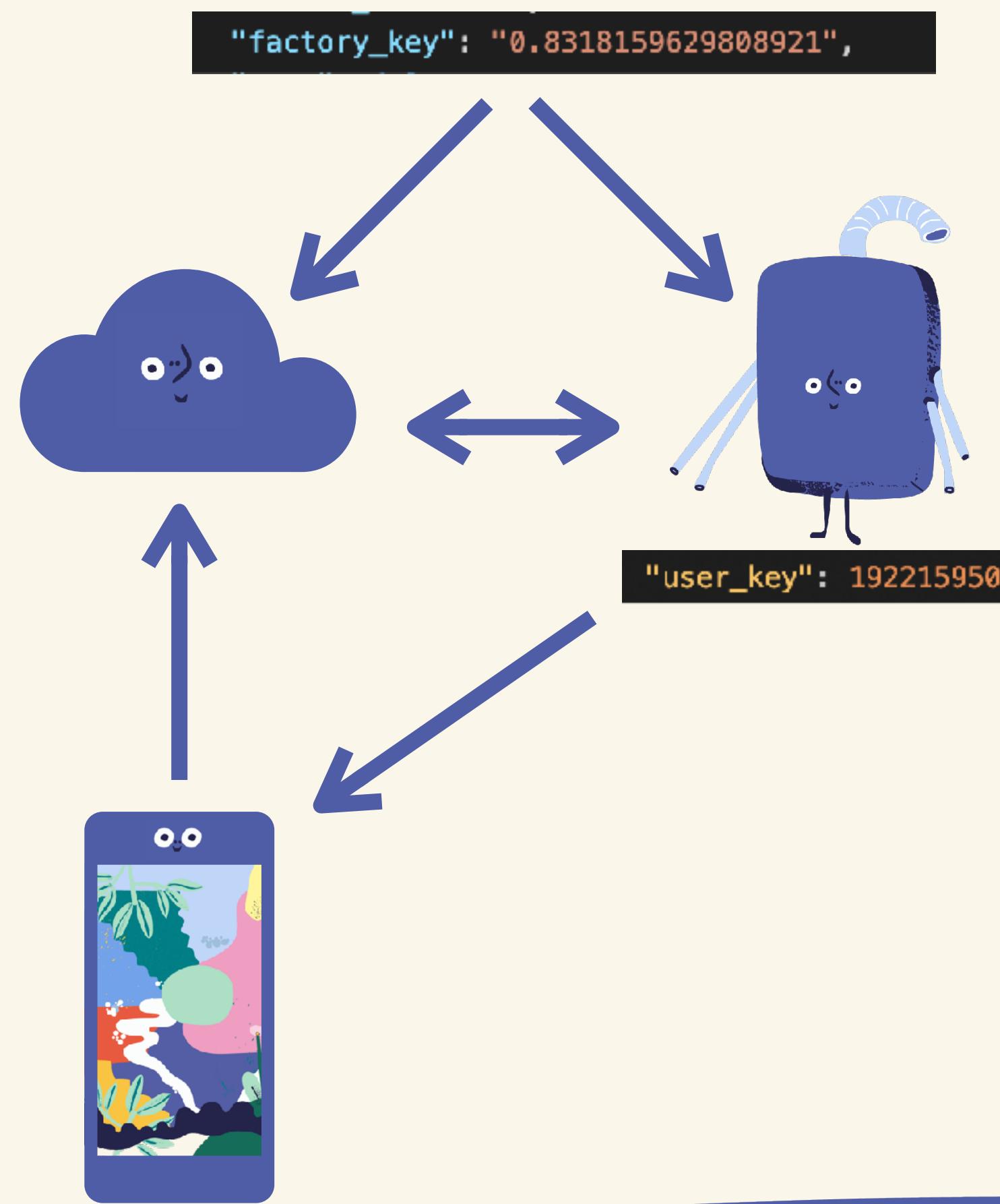
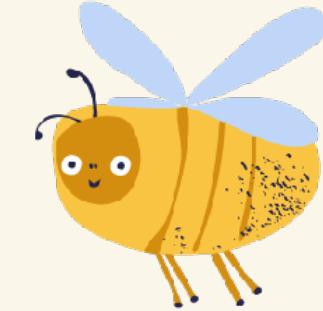
# Backend



- Umsetzung eines Servers der als Schnittstelle zwischen App und BloomBox dient.
- Stack bestehend aus Ubuntu 20.04.3, Node.js , Express für das Routing sowie MongoDB zum verwalten und persistieren der Daten.
- Lose Kopplung vom Client zum Server durch Erzeugung von Tokens für die UserSession.
- Sicherheit gewährleistet durch Autorisierung sowie durch SHA-256 Verschlüsselung und Hashing mit Salt bei Passwörtern.
- CRUD Methoden zum Erzeugen,Verändern und Löschen von Daten.
- Load Balancer mit Docker Compose und Nginx



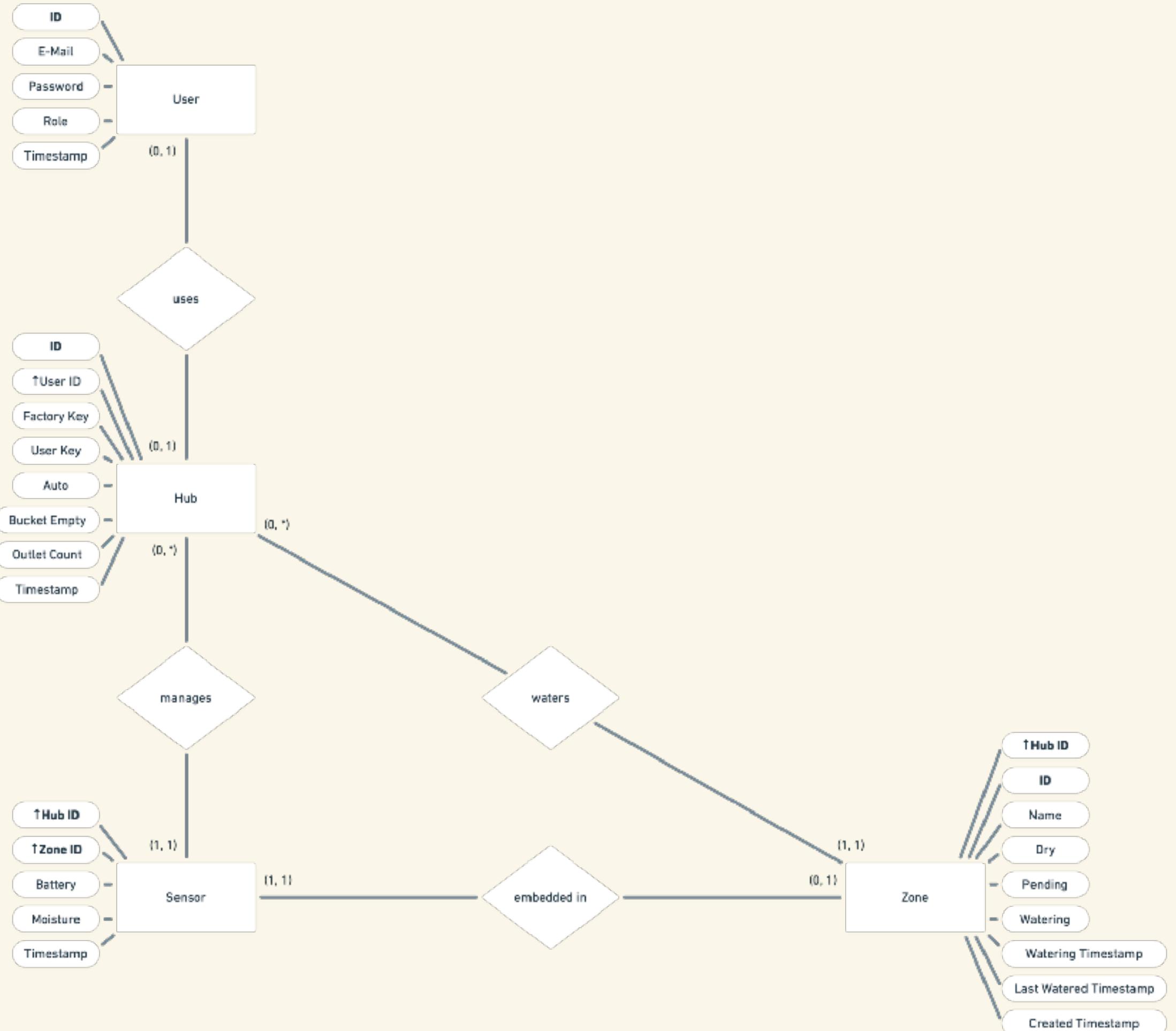
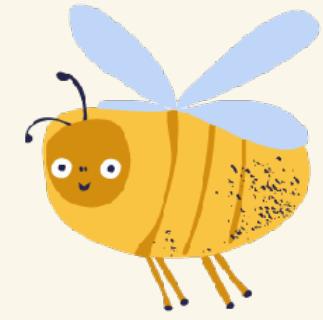
# Kopplung von App und BloomBox



- Algorithmus erzeugt aus einem String und Rauschen einen Double-Wert zB: 0.8412482804380098
- Wert ist auf dem Server & der BloomBox gespeichert und wird beim initialisieren der BloomBox dem Server übergeben.
- Verifizierung das es sich um eine Originale BloomBox handelt.
- Rückgabe von Token sowie zufällig generiertem User\_key.
- User\_key wird auf dem Display der BloomBox gezeigt und muss in App eingetragen werden.
- User\_key von BloomBox und App werden abgeglichen und bei MongoDB zusammengeführt.



# Datenmodellierung



- Skalierbares System
- NoSQL
- Dokumentdatenbanken
- Kombiniere jeden Schlüssel mit komplexen Datenstrukturen
- Jedes Dokument kann viele Schlüssel-Wert-Paare
- Sowie eingebettete Dokumente
- Benutzerfreundliche und flexible objektorientierte Programmierung



# Design



## Analyse

- Zielgruppenanalyse, z.B. Sinus-Milieus
- Wettbewerbsanalyse

## Produktdefinition

- Schlüsselaspekte und Geschäftsziele
- Frühe Wireframes

## Research

- Interviews
- Preference Testing

## Design

- Wireframes
- Prototypen
- Illustrationen
- Styleguide

## Testing

- Funktioniert das Design auf der App?
- Moderiertes Testen
- Umfragen

## Design Thinking

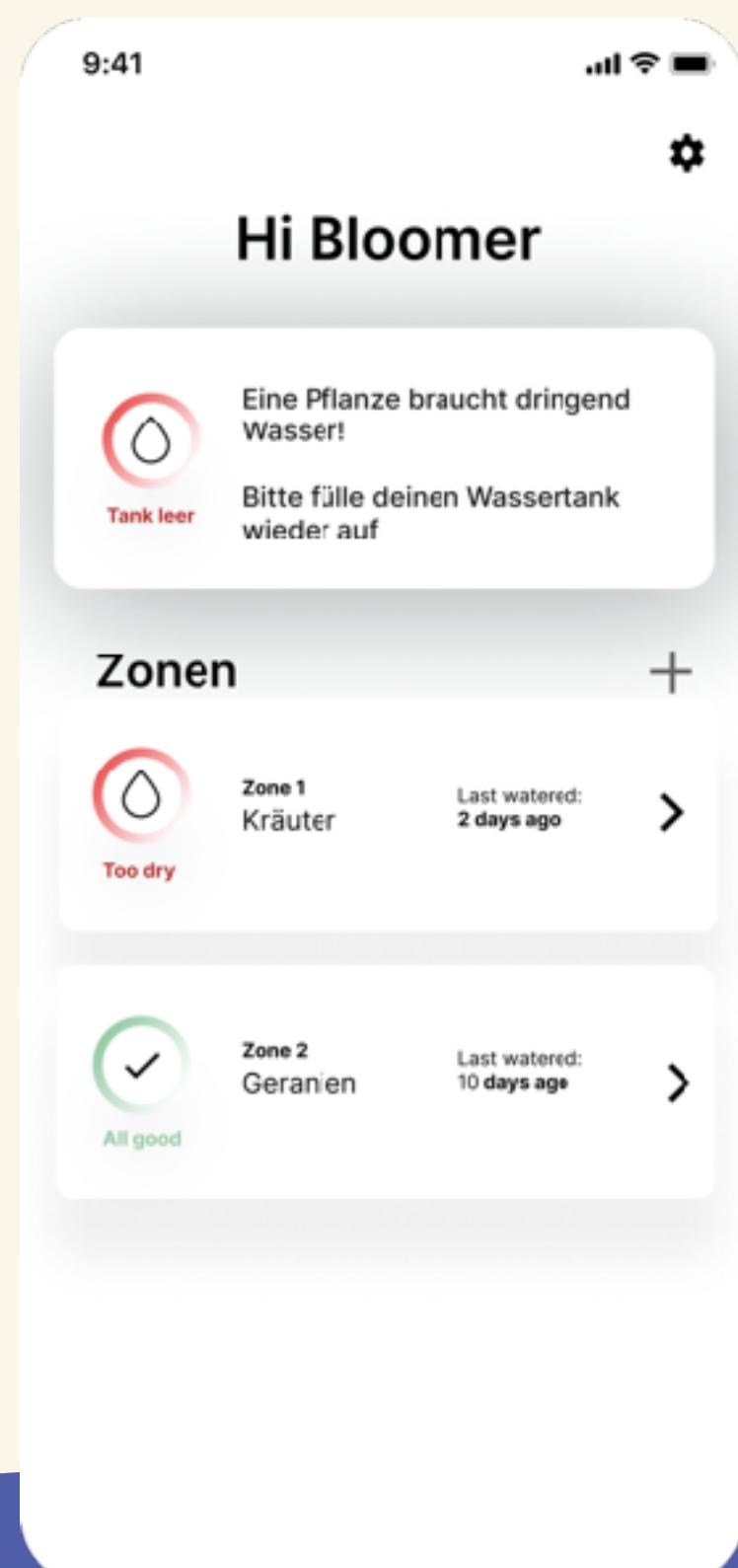


# Design

Prototypen

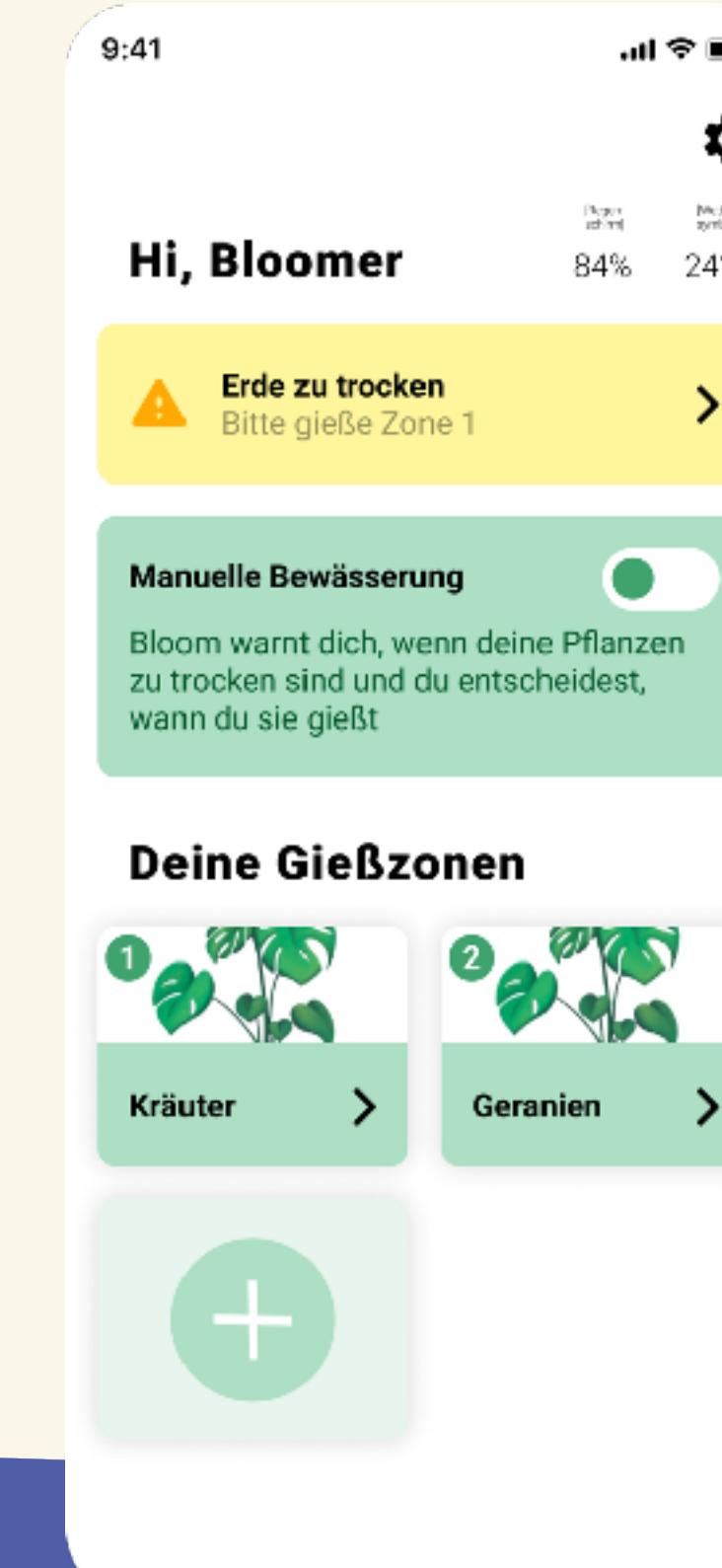
## 1. Iteration

Umsetzung der App-Architektur, klare Strukturen



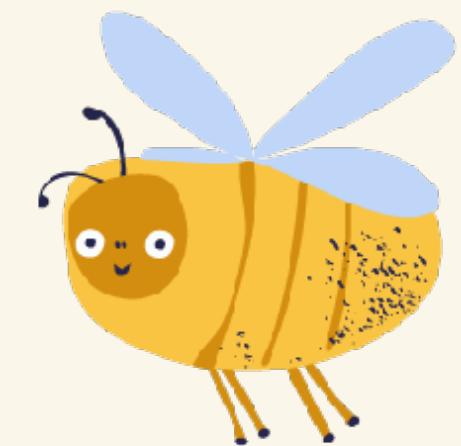
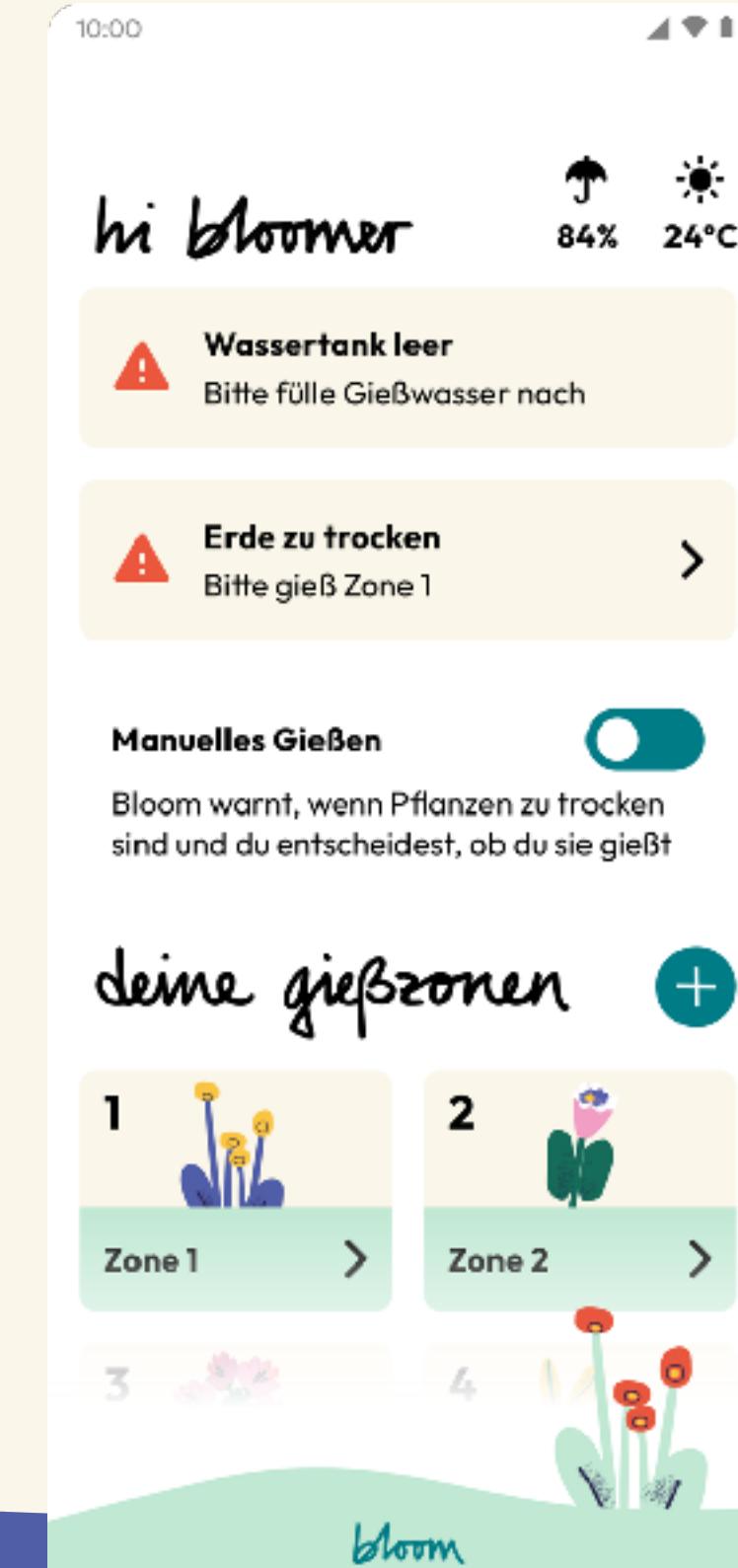
## 2. Iteration

Umsetzung nach UX-Analyse und Tests

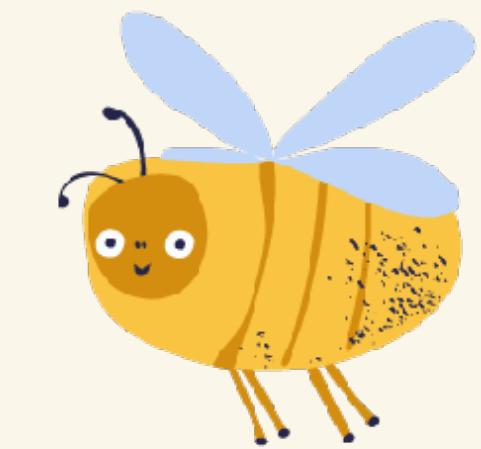


## Finaler Prototyp

App soll ein Erlebnis bieten und einfach zu nutzen sein



# Design



- zeitgemäß, freundlich, reduziert, vermittelt Lebensfreude
- mit Storytelling Spaß am Gärtnern vermitteln
- Komplexität des Produkts anhand komplexer Farbpalette und Illustration
- konsistentes, technisches Produkt so präsentieren, dass der User Freude hat beim Nutzen und es spielerisch leicht geht
- Organisch, haptisch, weich
- Gute Work-Life-Fun-Balance, pragmatisch aber Spaß und Unterhaltung nicht zu kurz

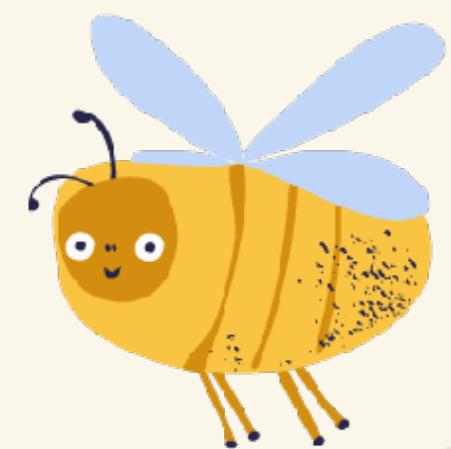
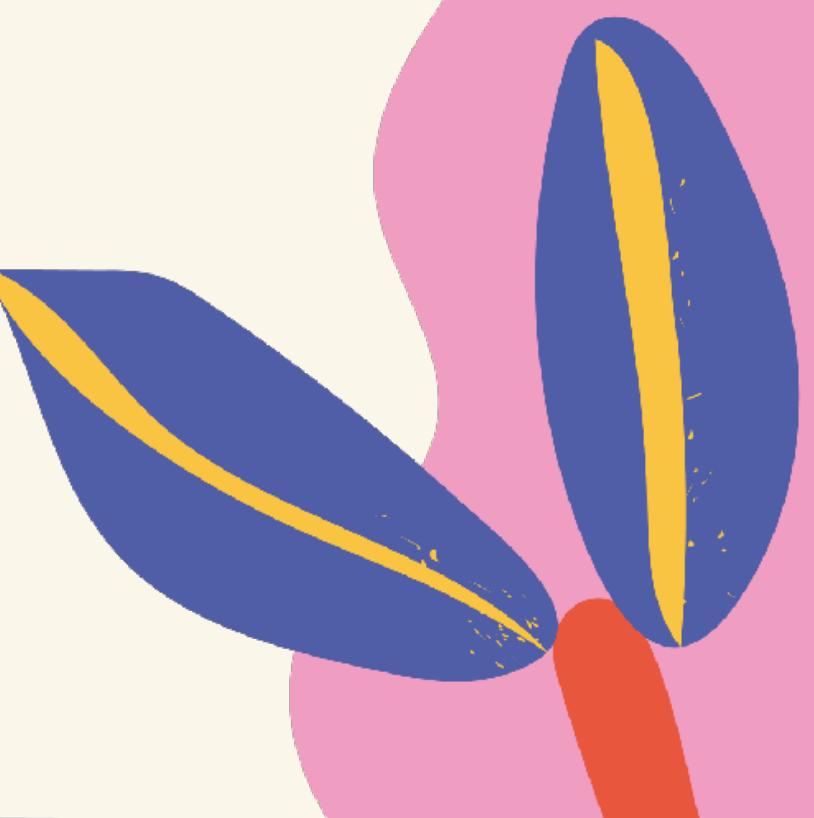
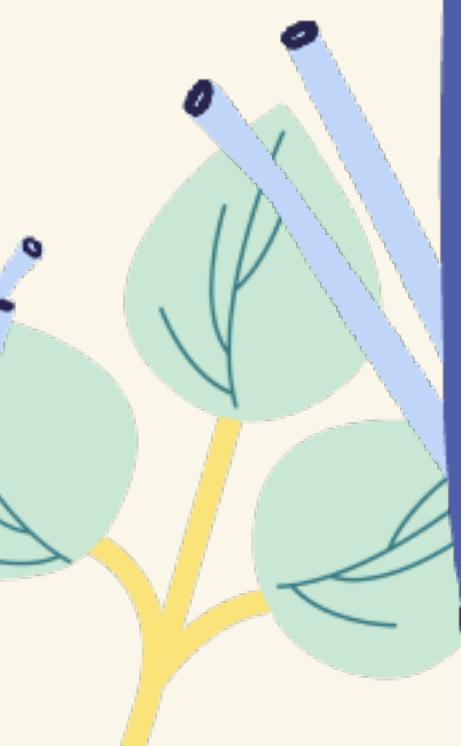
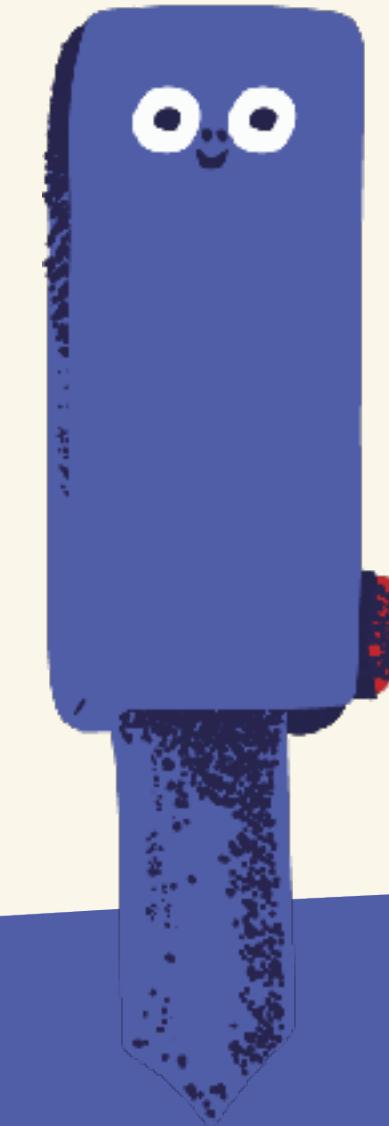
# Design

Illustrationen

hi bloomer

bloom

deine gießzonen



# Design

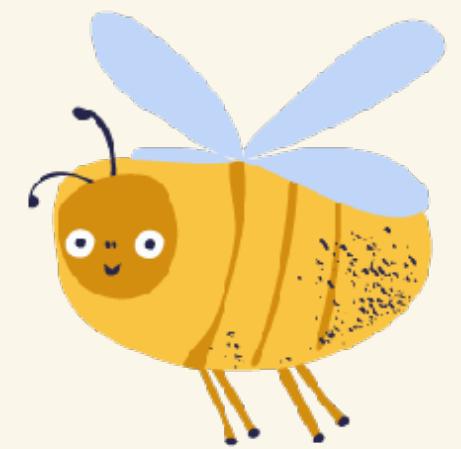
- Das Farbschema ändert sich, je nachdem ob der User das manuelle oder das automatische Gießen einstellt



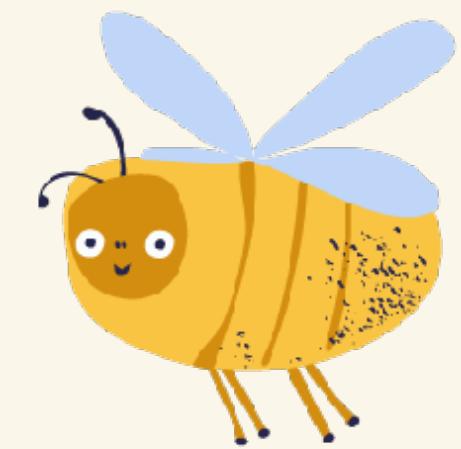
Manuell



Auto



# App



- Über Retrofit schicken wir mithilfe eines Interfaces, welches unsere HTTP Requests enthält, Anfragen an den Server
- Um diese Anfragen verarbeiten zu können, haben wir die Schemas unserer Model-Klassen vom Server in GSON umgewandelt
- Mit einem Interceptor können wir Tokens benutzen, um somit User Sessions zu gewährleisten
- In der App haben wir das MVVM Entwurfsmuster umgesetzt und auch Observer umgesetzt
- Die Daten werden in Intervallen vom Server geholt



# App

Login

Email  
bloom@mail.de

Passwort  
.....

E-Mail oder Passwort sind leider falsch

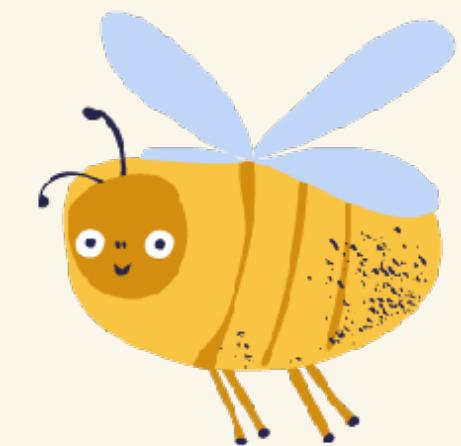
Einloggen

Du bist noch kein Bloomer?  
Zur Registrierung

Credentials in base64

Fehlermeldung

POST



Weather API

GET

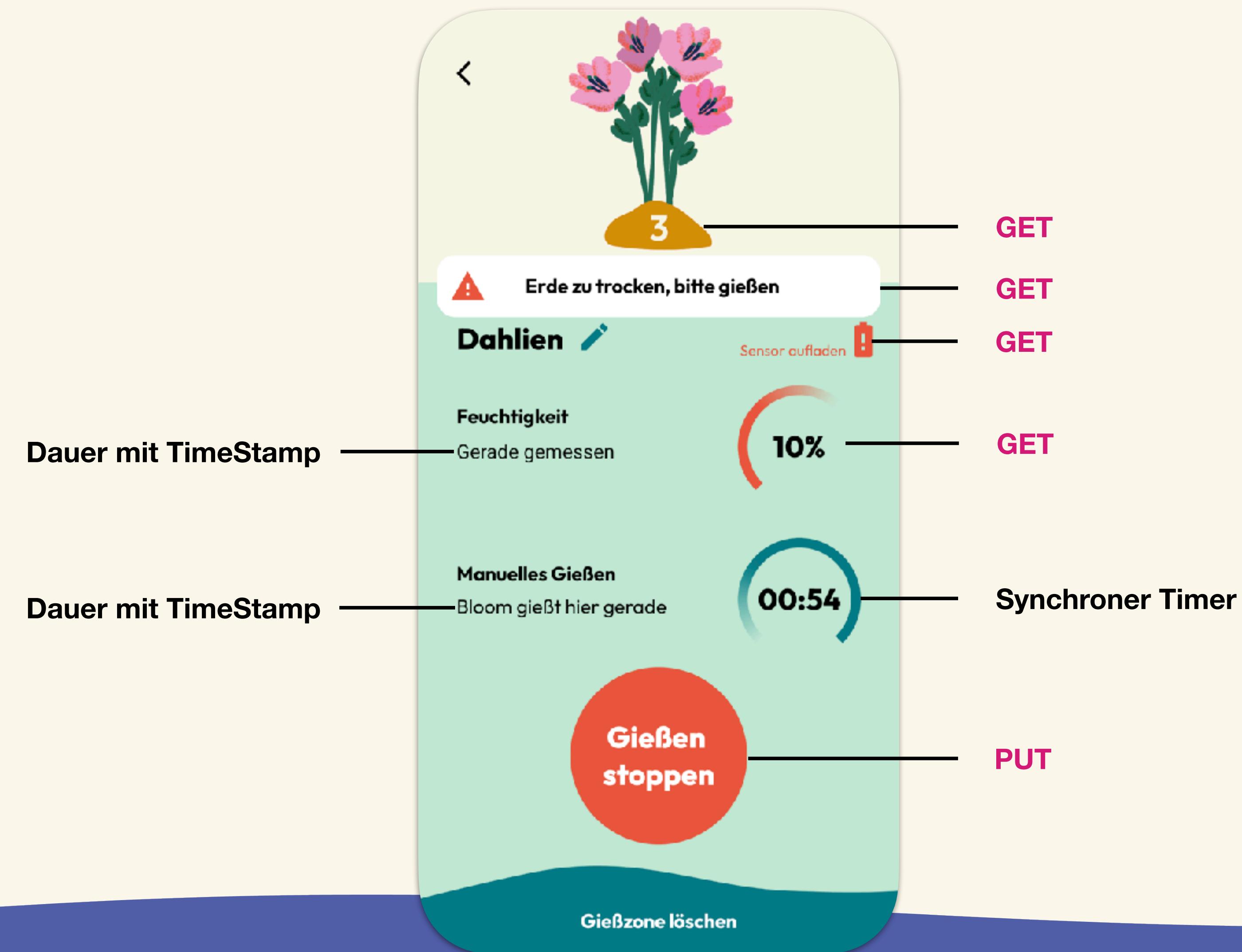
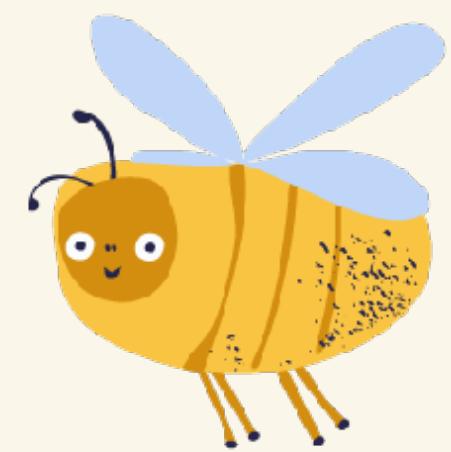
GET

PUT

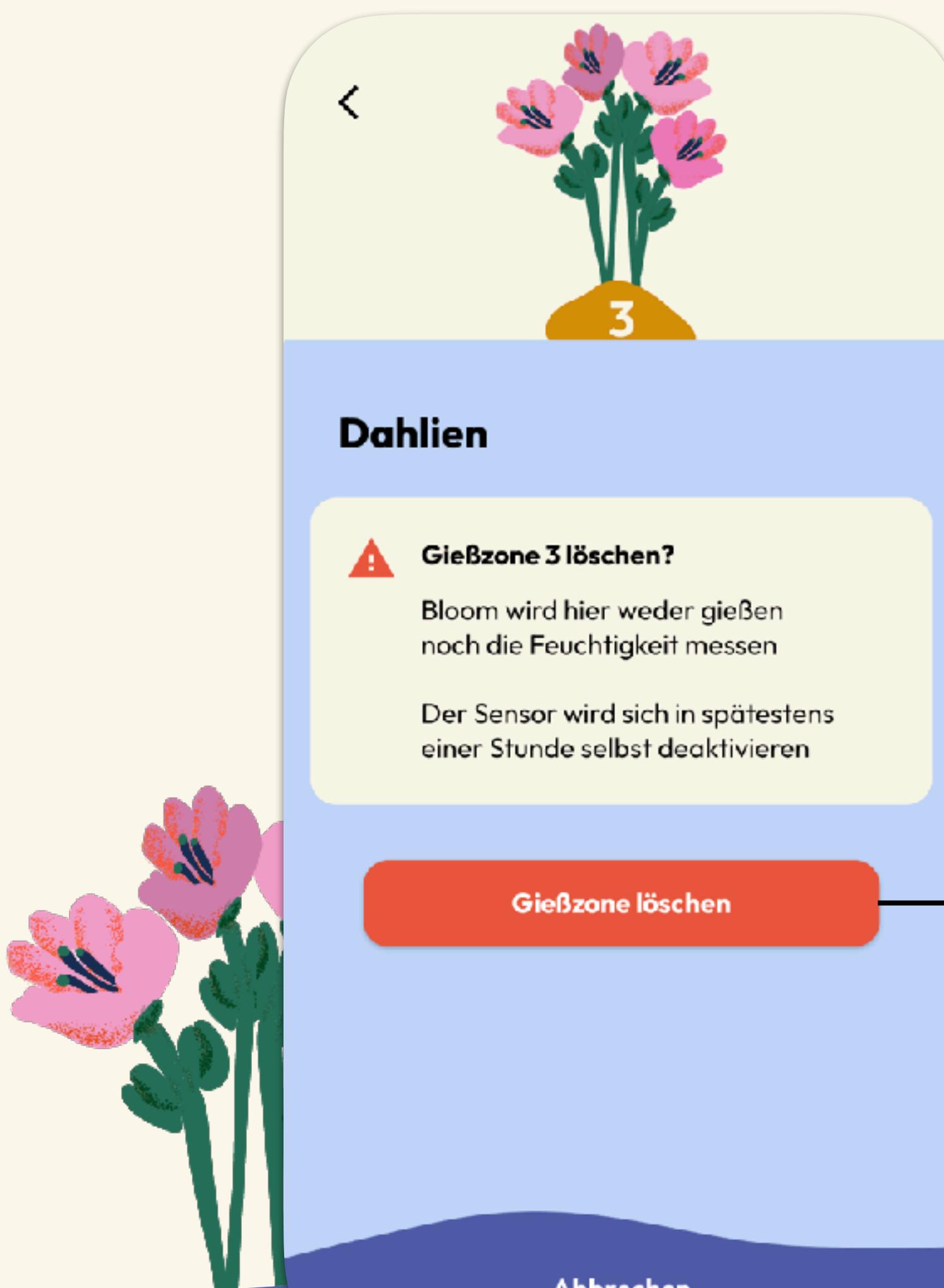
POST

GET

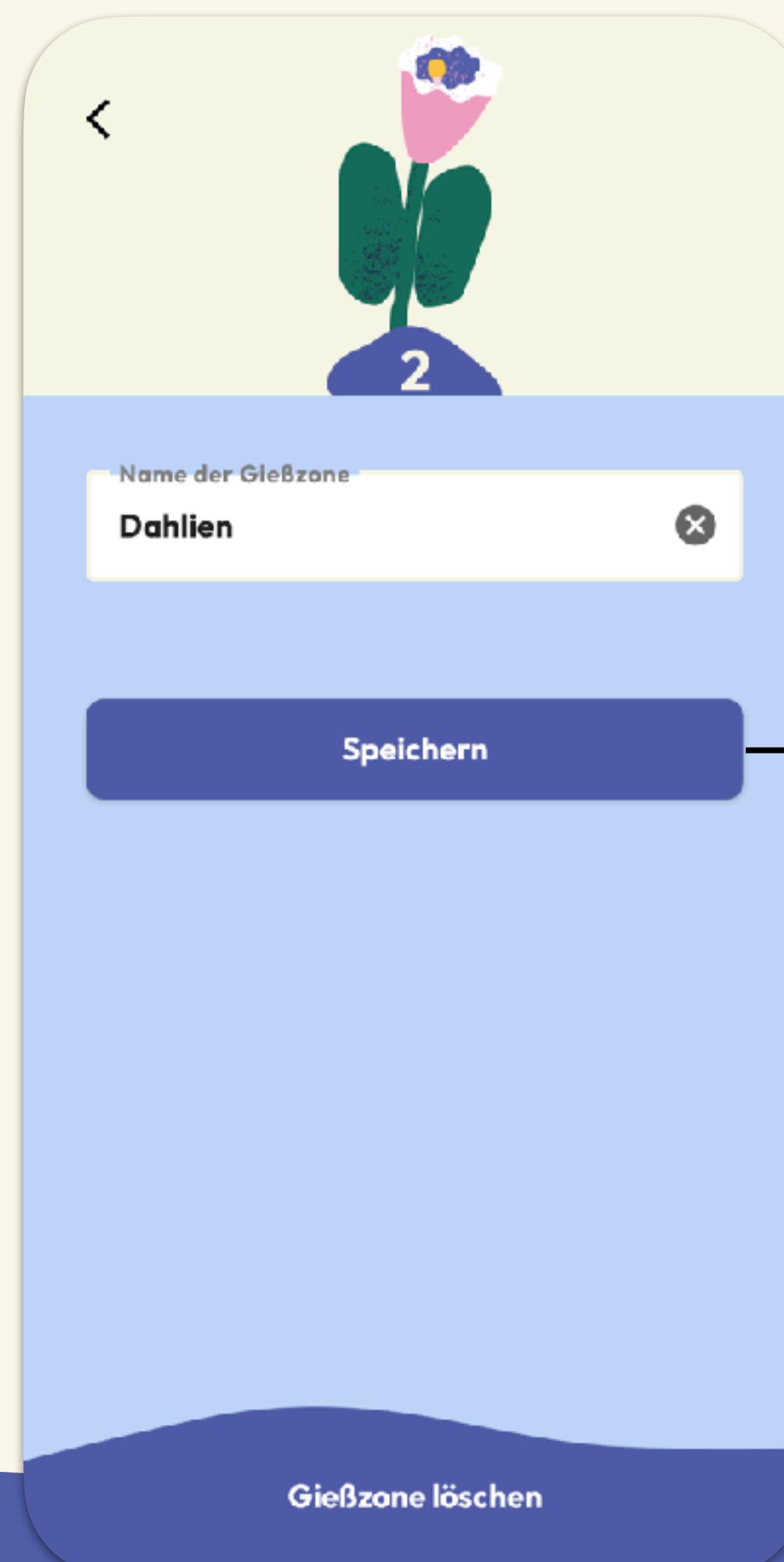
# App



# App

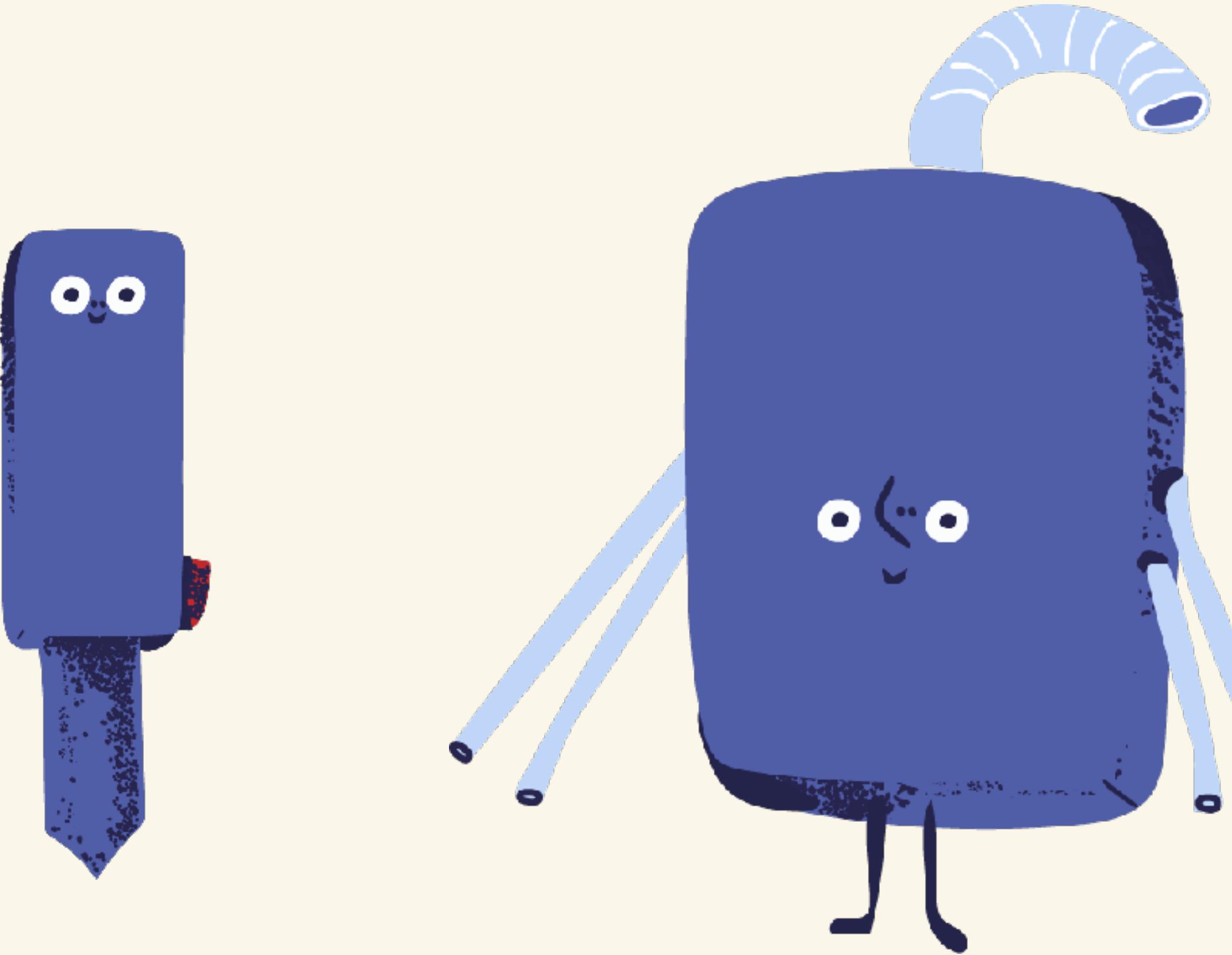
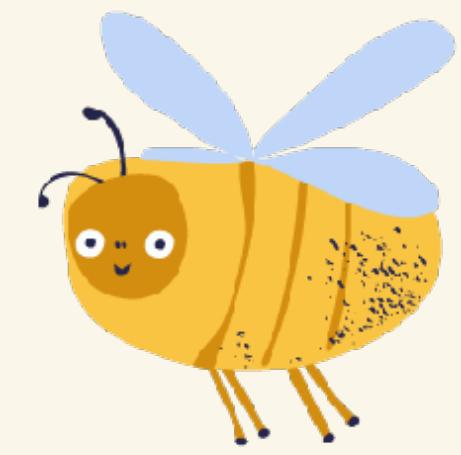


DELETE



PUT

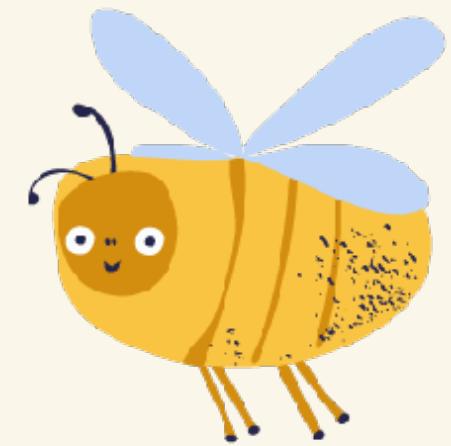
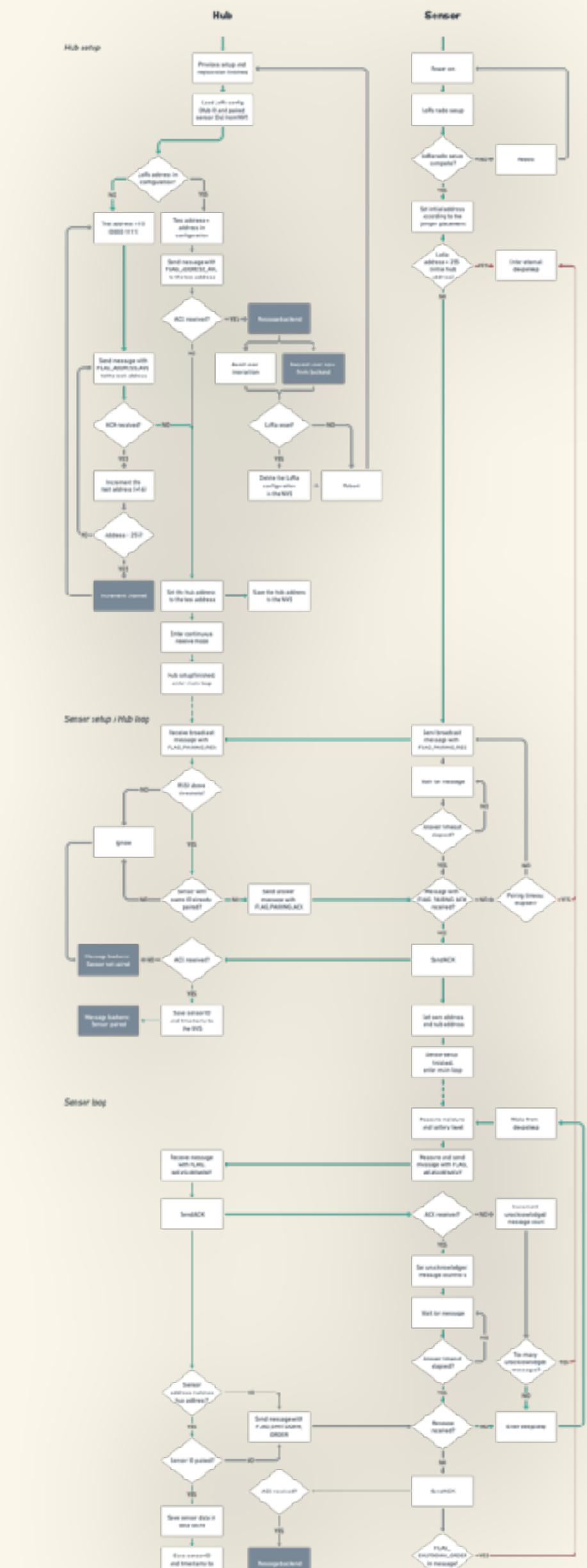
# Microcontroller



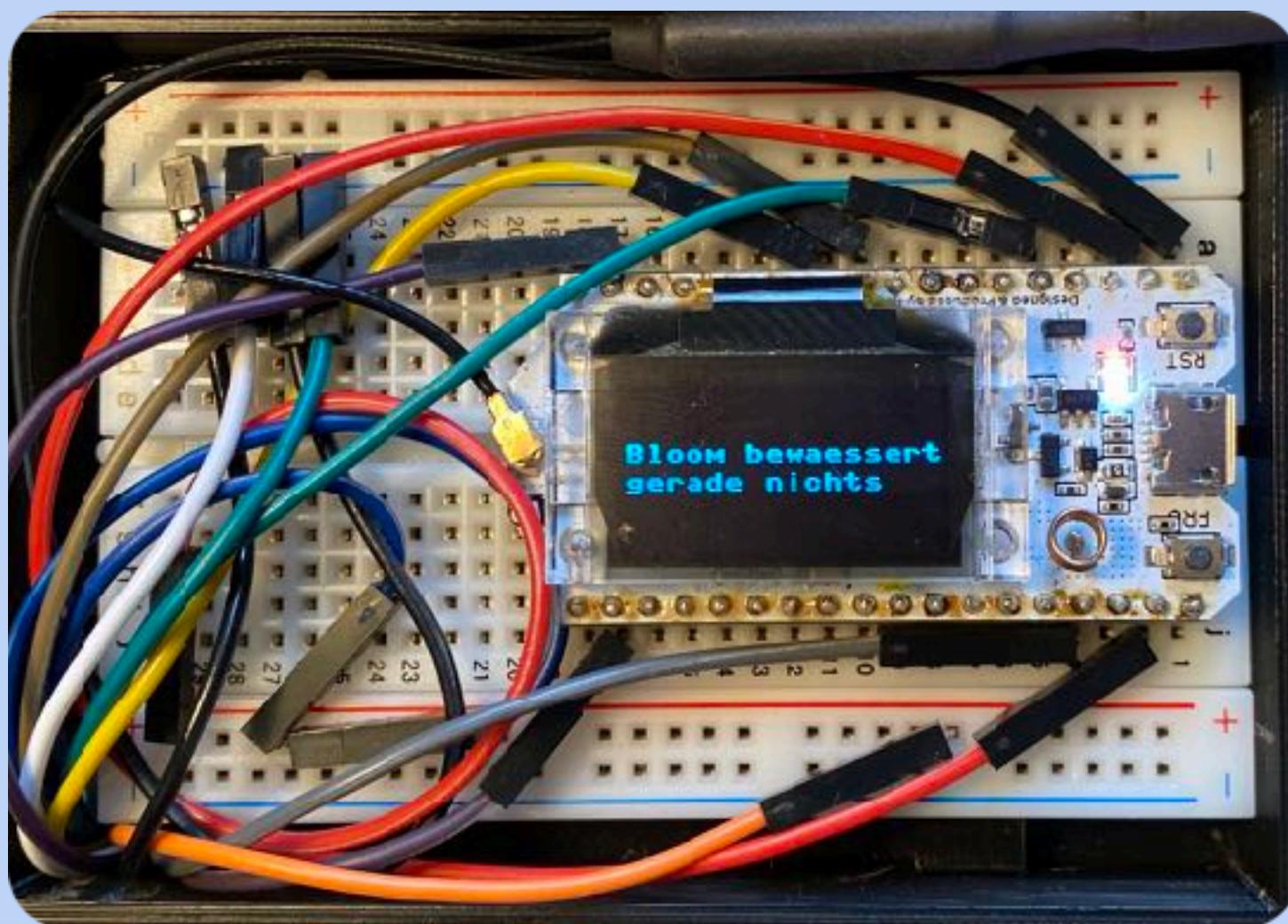
# Lora



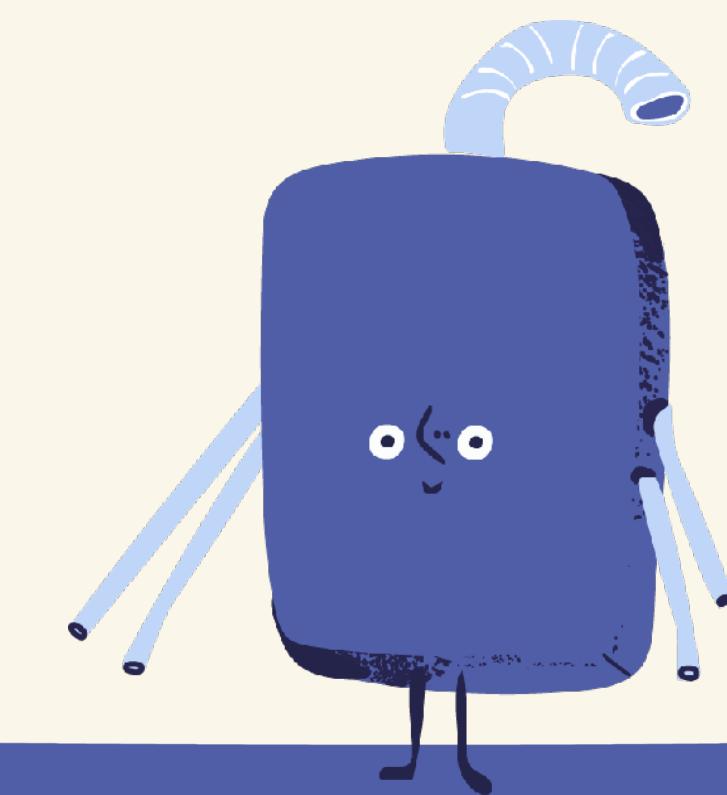
- IOT Kommunikationsprotokoll für lange Strecken
  - Long Range, Low Energy, Low Bandwidth
  - Skalierbar: Vom Balkon über Gärten bis hin zu Farmen
  - Protokoll für automatischen Setup und Messdatenaustausch
  - Bis zu 15 Bloom Boxen und 225 Sensoren/Gießzonen gleichzeitig



# Bloom Box



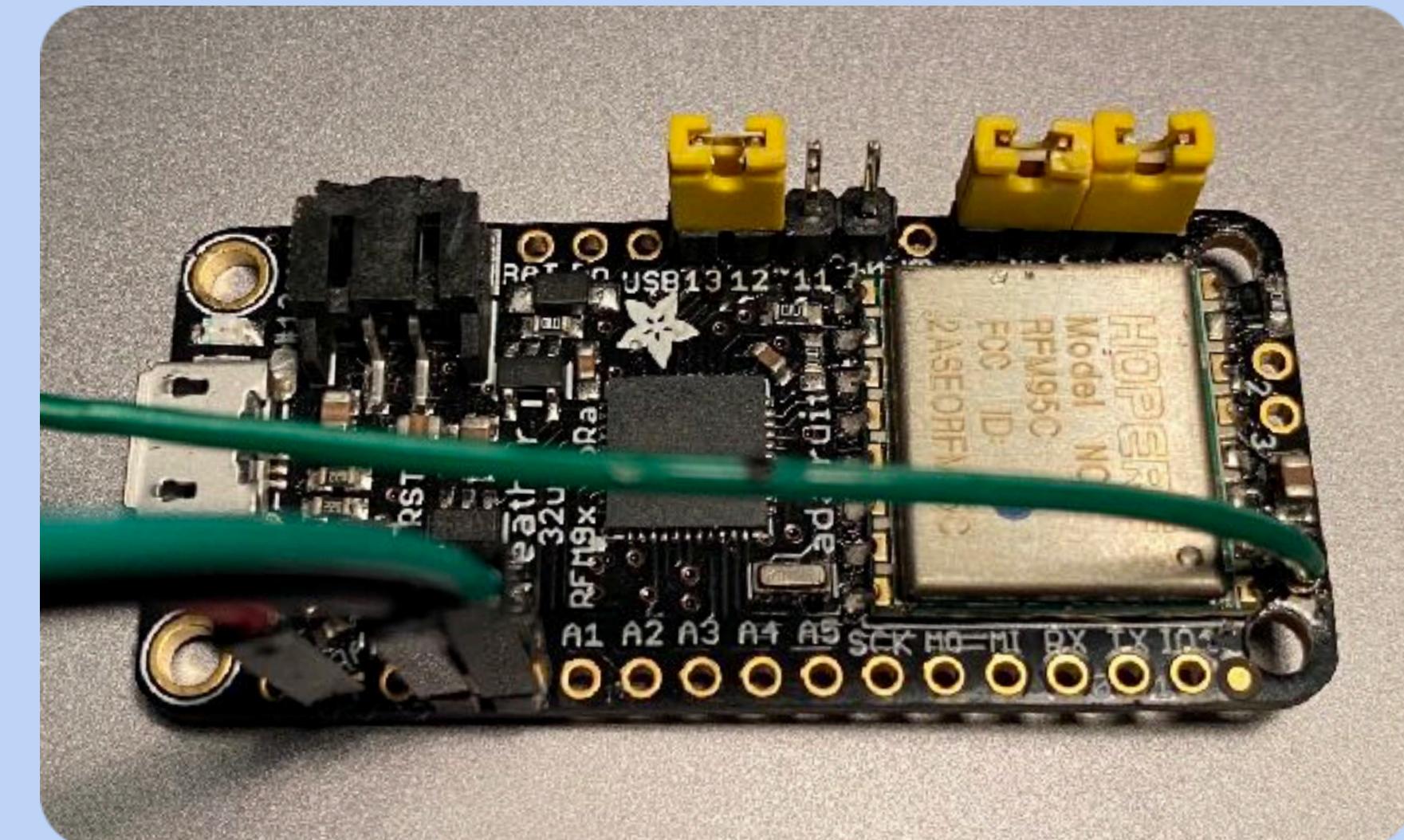
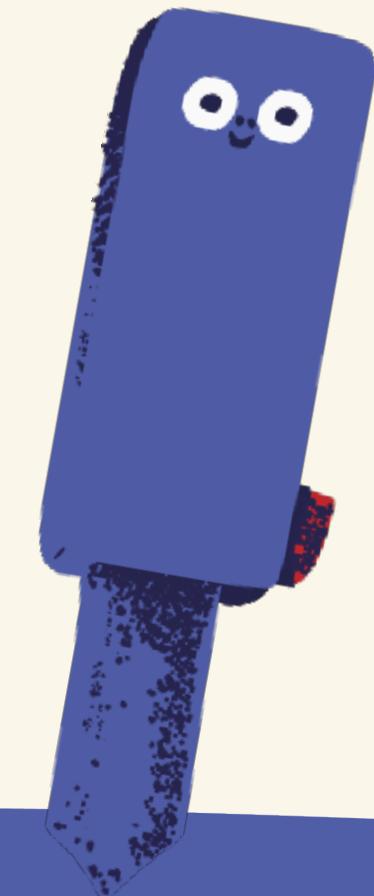
- Ventil- und Pumpensteuerung, Sensormanagement, Backendschnittstelle
- Heltec Entwicklungsboard mit ESP32 Microcontroller, WLAN, OLED, LoRa
- Steile Lernkurve, da keinerlei Vorerfahrung mit MCUs und Signalelektronik
- **Micropython**  
Effiziente Python Implementierung mit eingeschränkter Funktionalität
- Eigener HTTP-Handler, komplette Überarbeitung des LoRa-Treibers
- Wiederverwendbarkeit durch Modularisierung
- Ausfallsicher mit Resetroutinen
- Transparenter Setuptoolsprozess (außer WLAN über WPS)



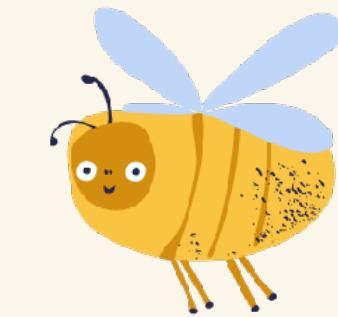
# Sensor



- Adafruit Feather mit ATmega32u4 8-bit Microcontroller, Batterieladefunktion und LoRa
- Kommunikation zwischen unterschiedlicher LoRa-Hardware komplex
- Automatisches Setup mit hardwarecodierten IDs
- Extrem energiesparend
- C++



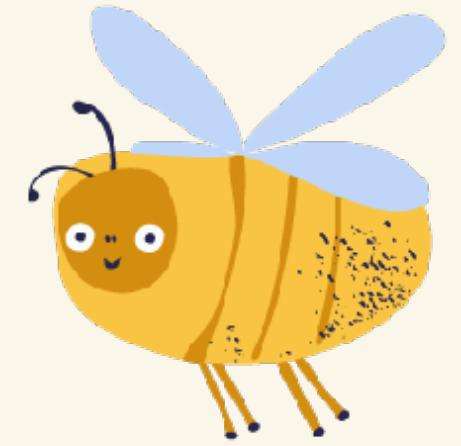
# Ausblick



App	Backend	Microcontroller	Hardware
Feuchtigkeitshistorie als Diagramm, Filterung der Feuchtigkeitsdaten	Pflanzprofile mit Feuchtigkeitspräferenzen	Benutzungsfreundliches WLAN Setup, z.B. mit WPS	Wassertank und Bloom Box in einem Gehäuse
Ändern der Userdaten über die App, Löschen des Users und der Bloom Box über die Einstellungen	(Automatische) Bodenprofile anhand von geologischen Daten zum Zonenstandort	Over-the-air Softwareupdates und Konfigurationsänderungen	Integration in Balkommöbel oder Pflanzkästen
Push-Benachrichtigungen	Prognostizieren des Wasserverbrauchs und ggf. Warnung zum rechtzeitigen Auffüllen des Wassertanks	Solarzelle für Dauerbetrieb und Lichtmessung	Gehäuse und Wassertank aus recyceltem Material
Dark Mode	Gießplan für halbautomatischer Betrieb, Flexible Einstellung der Gießzeit	Barrierefreiheit, z.B. mit Sprachausgaben der Displayanzeigen	Bloom Box ohne Tank und Pumpe mit direktem Anschluss an Wasserleitung
Pflanzenerkennung mit Google Lens	Gießempfehlung, bzw. automatische Gießentscheidung anhand der Wettervorhersage	Temperatursensor zur Frostwarnung	Bloom Box mit weniger oder mehr Gießausgängen
Gießzonen in schematischer Darstellung, Satellitenkarte oder AR-Umgebung anlegen und verwalten	Verschiedene Arten von Usern mit unterschiedlichen Rechten zur Verwaltung einer oder mehrerer Bloom Boxen	Dynamische Veränderung der Messintervalle (z.B. für Wintermodus)	Bloom Box mit LTE-Modul
Animation der Illustrationen	Messengerdienst unter Usern, die die gleichen Gießzonen betreuen	Offlinemodus ohne App, bzw. Backend	
Barrierefreiheit: Steuerung über Sprachassistenten	KI-gestützte Auswertung von Messdaten zur Optimierung der Gießempfehlungen, bzw. des automatischen Gießens	Anbindung an ein LoRa-WAN wie TTN	



# Team



Albert Kaminski // **Projektleitung, Webserver-Programmierung, Administrator des Servers**

Dominik Domonell // **Hardwarekonstruktion und -design, App-Programmierung**

Jelena Mirceta // **Webserver- und App-Programmierung, UI-Design**

Sahiram Ravikumar // **Webserver- und App-Programmierung, Systemarchitektur**

Simon Núñez Aschenbrenner // **MCU-Programmierung, Design, Systemarchitektur, Dokumentation**

