

## 03 Leistungspaket

### Arbeitstitel

Graw Sounding Console

### Generelles

#### Technisch

#### Anwendung

- Webbasiert
- PHP
- Laravel
- Laravel Nova - Unlimited license
- Vue.js
- Websockets

#### Hosting

- Cloud: containerbasiert (z.B. Docker, Kubernetes)
  - nach möglichkeit: komplett [laC](#) via Terraform
  - wenn serverless, dann evtl. <https://vapor.laravel.com/>
- On-Premises
  - evtl. <https://forge.laravel.com/>

#### Inhaltlich

- Aufstiegsdatenempfang von Bodenstationen
- Echtzeitverarbeitung von Aufstiegen
- Archivierung von Aufstiegen
- Echtzeitüberwachung von Bodenstationen
  - Statistiken: z.B. wie viele erfolgreiche Aufstiege
  - Performance-Überwachung: z.B. Durchschnittliche Höhe aller Aufstiege

### Sprache

- Umsetzung: einsprachig in englischer Sprache
- Vorbereitung: spätere Übersetzung

### Models

#### User

name	type	special
id	unsignedBigInteger	autoIncrementing
created_at	timestamp	nullable
updated_at	timestamp	nullable
deleted_at	timestamp	nullable
created_by	unsignedBigInteger	nullable
updated_by	unsignedBigInteger	nullable
deleted_by	unsignedBigInteger	nullable
email_verified_at	timestamp	nullable

name	type	special
rememberToken	string	nullable; length(100)
email	string	unique
password	string	
role	string	
name	string	
timezone	string	

Datenquelle

- Erstellung im Backend durch Admin

Station

name	type	special
id	unsignedBigInteger	autoIncrementing
created_at	timestamp	nullable
updated_at	timestamp	nullable
deleted_at	timestamp	nullable
created_by	unsignedBigInteger	nullable
updated_by	unsignedBigInteger	nullable
deleted_by	unsignedBigInteger	nullable
wmo_id	unsignedBigInteger	nullable
publicly_readable	boolean	
name	string	
city	string	
country	string	
latitude	decimal	
longitude	decimal	
altitude	decimal	
<i>weitere Stammdaten</i>	various	

Datenquelle

- Erstellung im Backend durch Admin

Flight

name	type	special	comment
id	unsignedBigInteger	autoIncrementing	
created_at	timestamp	nullable	
updated_at	timestamp	nullable	

name	type	special	comment
deleted_at	timestamp	nullable	
created_by	unsignedBigInteger	nullable	
updated_by	unsignedBigInteger	nullable	
deleted_by	unsignedBigInteger	nullable	
station_id	unsignedBigInteger	relationTo(Station)	
sonde_serial	string		
set_frequency	unsignedDecimal	between 400 & 406	
sonde_firmware_version	string		
max_altitude	decimal	nullable	maximale Höhe
min_pressure	unsignedDecimal	nullable	minimaler Luftdruck
max_distance	unsignedInteger	nullable	größte Distanz zur Bodenstation
duration	unsignedInteger	nullable	Flugdauer
last_altitude	decimal	nullable	Höhe der Sonde beim letzten empfangenen Datenpaket
max_wind_speed	unsignedDecimal	nullable	maximale Windgeschwindigkeit
avg_wind_speed	unsignedDecimal	nullable	durchschnittliche Windgeschwindigkeit
avg_ascent_speed	decimal	nullable	Durchschnitt aller positiven vertical_speed
avg_descent_speed	decimal	nullable	Durchschnitt aller negativen vertical_speed

#### Datenquelle

- Erstellung im Backend durch Admin
- Erstellung im Backend durch StationAdmin
- Erstellung durch Bodenstation über HTTP Endpunkt
- Import aus einer oder mehreren XML/JSON
  - evtl innerhalb einer ZIP, z.B. mit <https://gildas-lormeau.github.io/zip.js>

#### Measurement

*Achtung: Unterscheidung Real-Time Daten und Profildaten: Feuchtigkeit ist bis zu 120 sek verspätet (humidity\_raw > humidity\_timelag)*

name	type	special
id	unsignedBigInteger	autoIncrementing
created_at	timestamp	nullable
updated_at	timestamp	nullable
deleted_at	timestamp	nullable
created_by	unsignedBigInteger	nullable
updated_by	unsignedBigInteger	nullable
deleted_by	unsignedBigInteger	nullable
flight_id	unsignedBigInteger	relationTo(Flight)

name	type	special
time_after_launch	unsignedInteger	
utc_time	time	
pressure	unsignedDecimal	
temperature	decimal	
humidity_raw	unsignedInteger	
humidity_timelag	unsignedInteger	will be sent as an update each 60-120 sec
wind_speed	unsignedDecimal	
wind_direction	unsignedInteger	
latitude	decimal	
longitude	decimal	
altitude	decimal	
vertical_speed	decimal	either calculated from altitude or sent, not clear yet
geo_potential	unsignedInteger	
dew_point	decimal	
elevation	unsignedDecimal	
azimuth	unsignedInteger	
distance	unsignedInteger	

#### Datenquelle

- Import über einen Flight
- Erstellung durch Bodenstation über HTTP Endpunkt
- Update durch Bodenstation über HTTP Endpunkt

## HTTP Endpoints

*Unterstützung in GRAWMET muss noch gebaut werden, die Endpunkte werden aus Sicht des Web-backends definiert*

### Technisch

- HTTPS
- Auth (noch zu klären)
- Content- & Response-Type JSON

### Inhaltlich

- create Flight
- create Measurement
- update humidity\_timelag in Measurements

## Roles & Permissions

if a Station is publicly\_readable=true, every role can read the Station, its Flight & Measurements

### Admin

- CRUD User
- CRUD Station
- CRUD Flight
- CRUD Measurement

### StationAdmin

- C User (with Role User and min one Station)

## Limited by assigned Station

- RUD User
- RU Station
- CRUD Flight
- CRUD Measurement

## Limited to own User

- RU User

## User

## Limited by assigned Station

- R Station
- R Flight
- R Measurement

## Limited to own User

- RU User

## Visualisierung & Berechnung

---

- Flight
  - Dashboard
    - during flight: flight dashboard
      - Status der Sonde: noch im Aufstieg oder bereits geplatzt
      - Startzeit & bisherige Flugzeit
      - Frequenz
      - Seriennummer
      - Typ der Sonde (aus firmware version abgeleitet)
      - Station
      - Aktuelle PTU/Wind Messwerte
        - Pressure
        - Temperature
        - Humidity
        - Wind direction
        - Wind speed
      - Aktuelle Position
        - Lat
        - Long
        - Höhe (geopot)
        - Entfernung zur Bodenstation
    - after flight:
      - Status der Sonde: finished
      - Statistiken (eindimensional)
        - max altitude
        - max distance
        - *weiteres folgt*
  - Karte
    - Auf Basis von [sondehub-tracker](#)
    - Echtzeitdarstellung via *WebSockets* oder *Resource Polling* (über Prototyp klären)
    - *Zu Klären und Präzisieren: Durch die Echtzeitdaten werden selektierte Flights auch live auf der Karte getrackt. Neben der Kartendarstellung von Sondendaten wird innerhalb einer Station auch die Karte mit allen Measurements bereitgestellt. Ebenso kann mit der gleichen technischen Basis eine Karte für alle Flights einer Station oder alle Flights des gesamten Systems implementiert werden.*
  - Tabelle aller Measurements (Messwerte & Datenprodukte)
    - Echtzeit durch [Resource Polling](#)
  - Diagramme (mehrdimensional)
    - Liniendiagramme
      - alle Messwerte nach Zeit oder nach Höhe
      - filterbar nach Messwert
      - eventuell filterbar nach Zeit
    - eventuell auch Skew-T
- Station
  - Stammdaten
  - Zusammenfassung der statistischen Indikatoren der zugehörigen Flights in Durchschnittswerten, filterbar nach Zeit
    - eventuell auch in einigen *noch zu definierenden* Zeitabschnitten immer angezeigt

## Weiteres

---

- Test-Umgebung auf Kiwis & Brownies Cloud-Architektur
- Finales Produkt wird einmalig unverschlüsselt auf einem passenden Server bei Graw eingerichtet
- Codeverwaltung in GitLab Organisation von Graw, andere Plattformen nur als Mirror
  - Dies ermöglicht auch eine Mischform aus Closed Source Entwicklung, bei der man manuell nach Prüfung von einem closed GitLab Repo in ein open GitHub Repo mirrored.