basiered auf Input-File

Meta-Data ◆ Boarding

→ Name Prüfer
→ Datum
→ ... (siehe Daten-Doku)

Kategorie

Iteration ◇ ParticipantList

-deleted : Bool

Meta-Data

→ Name : String

Participant

→ Name : String, Sirname : String
→ Matr. Nr. : Int
→ Anwesend : Bool
→ Vermerk/Anmerkung : String
→ LastTimeModified : Date
→ FirstTimeModified : Date
→ deleted : Bool

Backup

▲ Kann estellt werden mit

LocalBackup

- Boarding
- TimeOfBackup : Date
- checksum?

als MAC realisirt

RemoteBackup

- anonymised Boarding
- TimeOfBackup : Date
- checksum?

(De-)
Serialize
in json
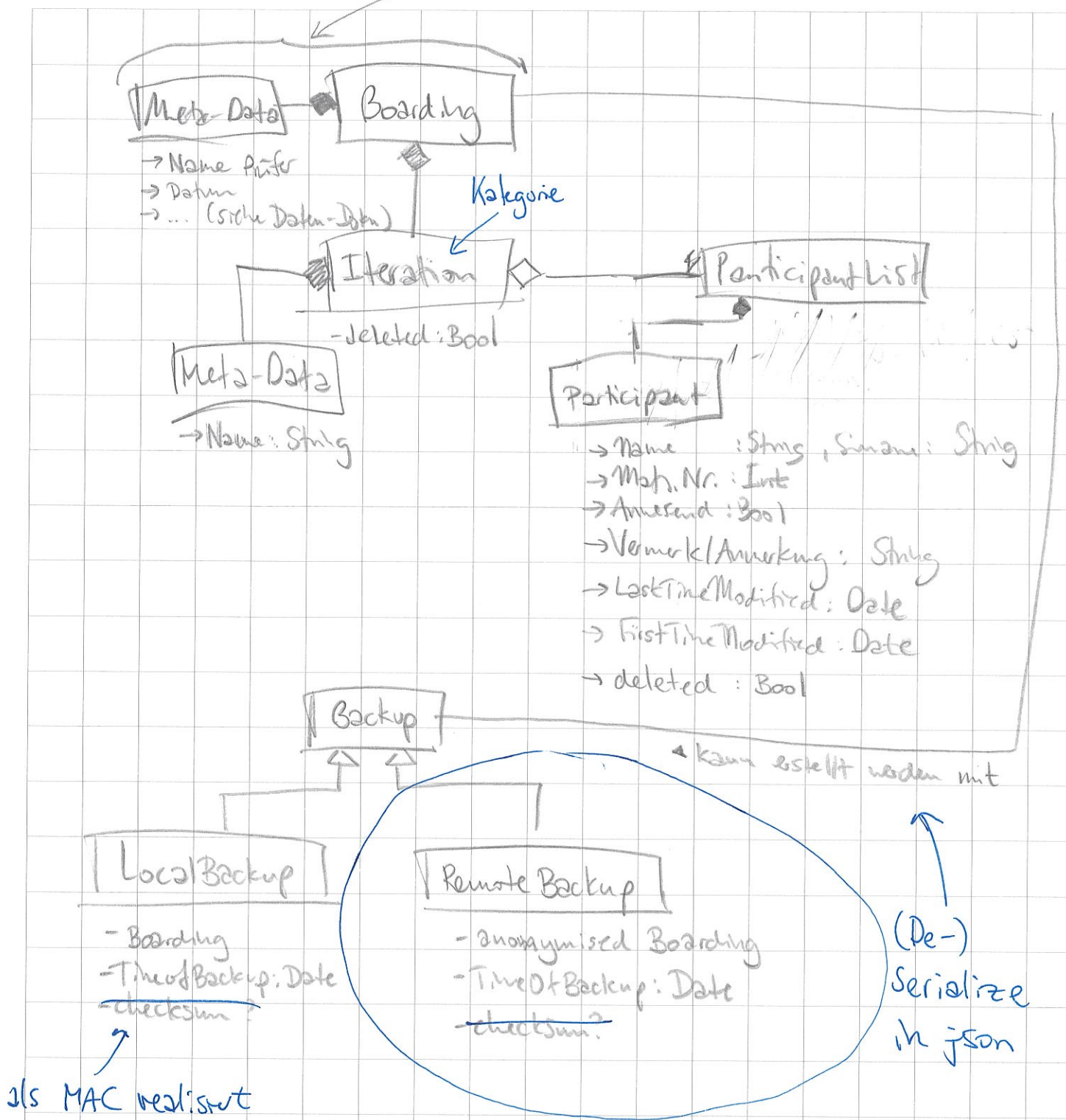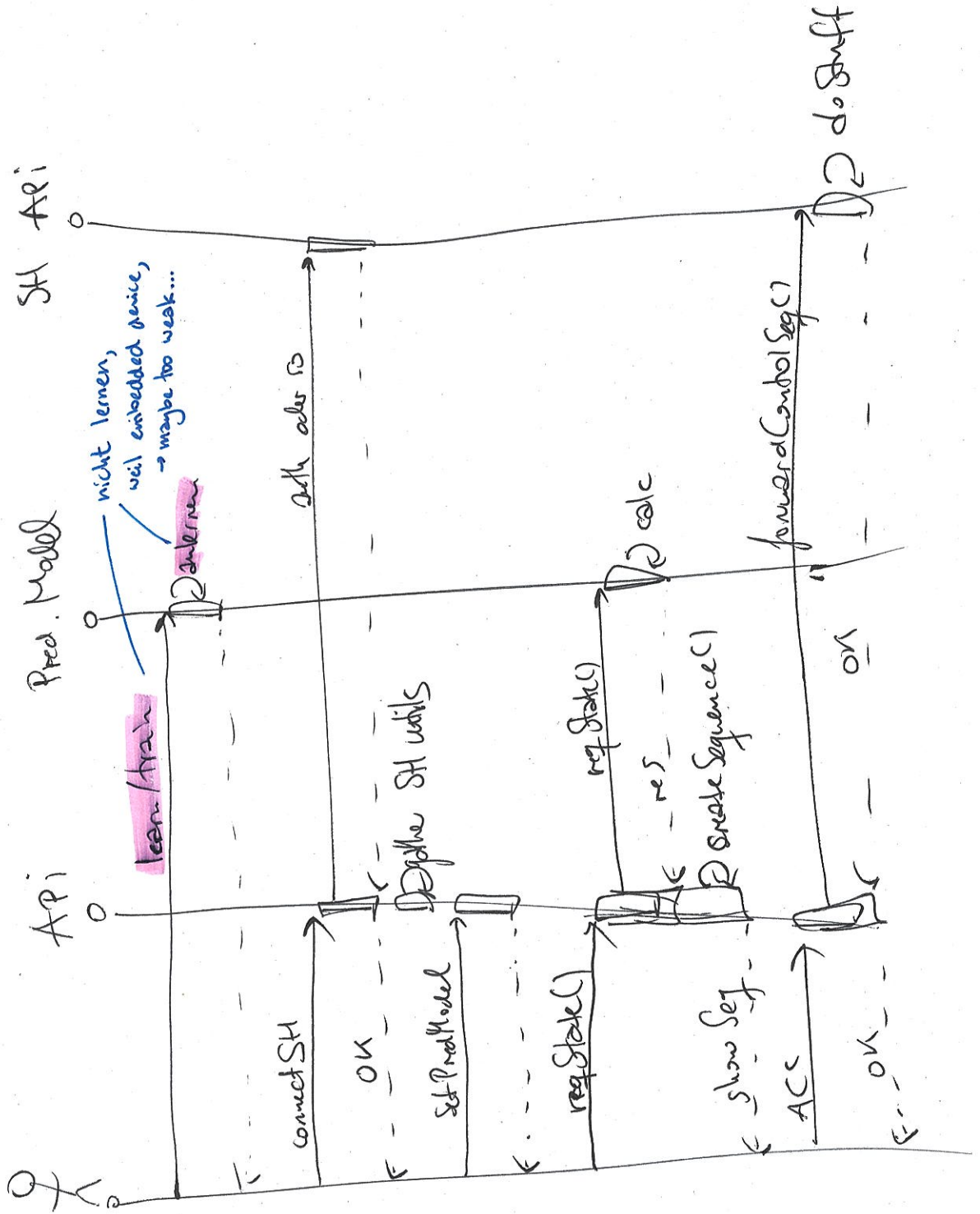
SETUP

1. Connect FW to SH-API (credentials hardcoded or via REST call?)

2. Fetch, gather services + devices + env. properties.

3. Set prediction model(s) to environment property.

4. Do fancy shit!

    4.1. "Common" command to control devices + services.

    4.2. "Special"/"Predictive" command to control env. property.

    4.3. Get SH device, service, etc. states.

API   Pred. Model   SH API

? K

connect SH

OK

Set Pred Model

learn/train

Inference

nicht lernen,
weil embedded device,
→ maybe too weak...

auth als S

Update SH units

req State()

req State()

res

calc

create Sequence()

show Seq

forward Control Seq()
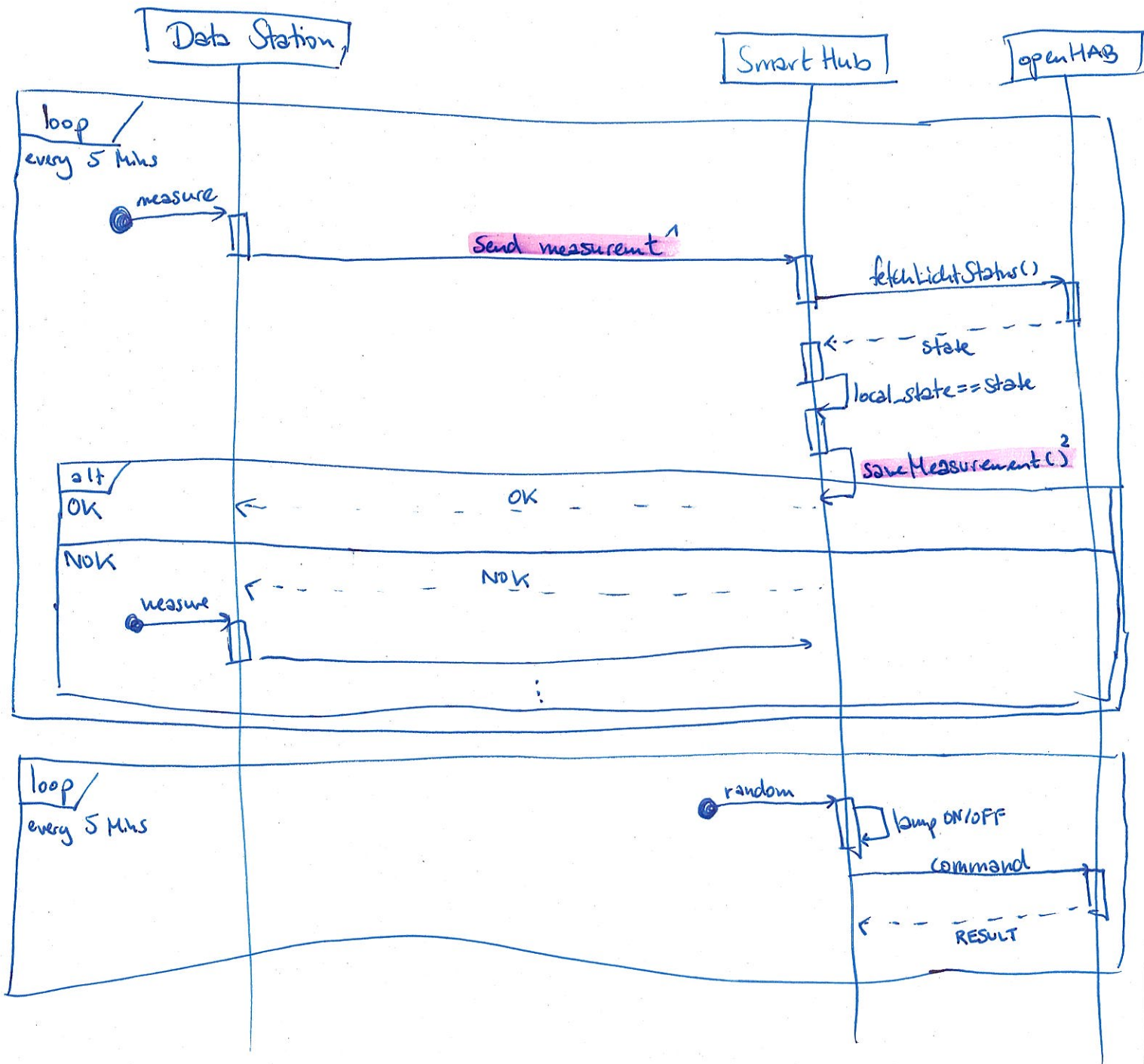
ACC

OK

OK

do Stuff

PoC

alle 5 Minuten:

Messstation:
1. Lichtint. messen
2. An SmartHub senden

P:/SmartHub:
1. zufällig Licht an/aus
2. Licht ggf. steuern

1 measurement:

```
{                                              {
  timestamp: YYYY-MM-DD HH:mm:ss,       OR      timestamp: <miliseconds>,
  value   : <INT>                               value : <INT>
}                                              }
```

2 Save Measurement:

```
{
  timestamp: <see above>,
  value    : <INT>,
  ~~light~~ devices: [
                {<device_type>
                 ~~plug~~: {
                            <name>
                     name: "~~plug~~",
                            <value>
                     value: "~~ONS~~"
                 },
              ]
}
```

EX:

```
{ timestamp: "2023-06-13 12:00:11",
  value     : "2083",
  devices  : [ "plug" : {"name": "Shelly PlugS",
                         "value": "ON"
               },
             ]
}
```

~~time.csv~~ time → append to measurements.json
        → Security back-up / copy every 30/60 Minutes