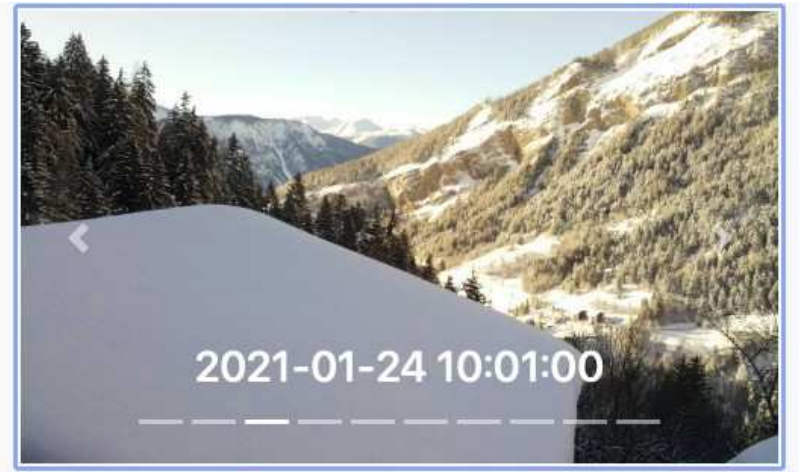


# Don't give up



# «Dorben»

- No power supply
- But a small solar panel
- Every hour a picture
- NFA
  - Energy efficient
  - Hosting backend at existing Hoster



# Implementation

- Hosted just supports php
- Decision for «Lumen» as Framework
- Followed «cookbook» to realize rest api
- Just Integration tests
- No real «Unit»-Tests

# Result

- Long methods
- Strong coupling to «Lumen»
- No tests without database
- Not motivated for more features

```
$query = Picture::query();
// Log::debug("Camera: $cameravid");
if ($cameravid != null) {
    $query = $query->where('camera_id', '=', $cameravid);
}
if ($request->hasAny([
    self::QUERY_PARAMETER_DAY,
    self::QUERY_PARAMETER_WEEK,
    self::QUERY_PARAMETER_MONTH,
    self::QUERY_PARAMETER_YEAR
])) {
    if ($request->has(self::QUERY_PARAMETER_DAY)) {
        $day = new \DateTime($request->input(self::QUERY_PARAMETER_DAY));
        $from = $day;
        $to = $day;
    } else if ($request->has(self::QUERY_PARAMETER_WEEK)) {
        $day = new \DateTime($request->input(self::QUERY_PARAMETER_WEEK));
        $from = (clone $day)->modify('monday this week');
        $to = (clone $day)->modify('sunday this week');
    } else if ($request->has(self::QUERY_PARAMETER_MONTH)) {
        $day = new \DateTime($request->input(self::QUERY_PARAMETER_MONTH));
        $from = (clone $day)->modify('first day of this month');
        $to = (clone $day)->modify('last day of this month');
    } else if ($request->has(self::QUERY_PARAMETER_YEAR)) {
        $day = new \DateTime($request->input(self::QUERY_PARAMETER_YEAR));
        $from = (clone $day)->modify('first day of january');
        $to = (clone $day)->modify('last day of december');
    }
    $query = $query->where('taken_at', '>=', $from->format('Y-m-d') . ' 00:00:00')
        ->where('taken_at', '<=', $to->format('Y-m-d') . ' 23:59:59');
}
if ($request->has(self::QUERY_PARAMETER_FROM)) {
    $from = new \DateTime($request->input(self::QUERY_PARAMETER_FROM));
    $query = $query->where('taken_at', '>=', $from);
}
if ($request->has(self::QUERY_PARAMETER_TO)) {
    $to = new \DateTime($request->input(self::QUERY_PARAMETER_TO));
    $query = $query->where('taken_at', '<=', $to);
}
$query = $query->orderBy('taken_at', 'ASC');
```

# After Refactoring

- (Nearly) independent implementation
- Simple, readable Unit-Tests
- Readable code
- Possibility to reuse code
- Found fun again for more features

```
private const QUERY_MAPPING = array(  
    self::QUERY_PARAMETER_DAY => 'byDay',  
    self::QUERY_PARAMETER_WEEK => 'byWeek',  
    self::QUERY_PARAMETER_MONTH => 'byMonth',  
    self::QUERY_PARAMETER_YEAR => 'byYear',  
    self::QUERY_PARAMETER_FROM => 'from',  
    self::QUERY_PARAMETER_TO => 'to',  
);
```

```
$queryBuilder = PicturesQueryBuilder::create();  
$queryBuilder->byCamera($cameraId);  
foreach (self::QUERY_MAPPING as $parameter => $method) {  
    $this->createQuery($request, $parameter, $queryBuilder, $method);  
}  
$query = $queryBuilder->build(Picture::query());
```

# Lessons learned

- Study the programming language helps
- Be careful when using a «cookbook»
- It's ok to have several attempts

# Lessons learned

- TPP helped me a lot (i.e. rule 7: scalar → array)
- Implementing on a framework may be easy, but there are costs
- I need to practice more

Don't just stay on the treadmill...



Downloaded from  
<https://www.sportxx.ch/de/p/463026200000/tunturi-t80-treadmill-endurance>  
at 2021-01-25 07:15

... go outside

