OwnPhotos

Final Presentation

2020.02.19

About Project

What is Ownphotos?

- Self-hosted photo management solution.
- Replacement of Google photo and I-photos.

About Project

- For what?
    - Manage your photos.
    - Share your photos with your friends.

- For who?
    - People who use smartphones.
    - Especially, who want to manage their photos.
    - People who care about their privacy.

About Project

- Main Features
    - Not sending your photo to Server !
    - Process on client side
    - Classify by Month/Year
    - Support various file formats such as JPG, PNG, HEIC
    - Easier user interface to browse photos.

Own Photos – Frontend

- Tasks
    - Setup development environment
    - Design API to get browse by month/year
    - Create interface mockups to support browsing by month/year
    - Implement feature to browse month/year

Own Photos – Frontend

- Design API to browse by month/year

user

```
getMonth(e){
    this.setState({
        month_txt : e.target.value
    });
}
getDay(e){
    this.setState({
        day_txt : e.target.value
    });
}

const month_txt = this.state.month_txt;
const day_txt = this.state.day_txt;
 return (
  <div>
  <input
    value={month_txt}
    onChange={this.getMonth} />
  <input
    value={day_txt}
    onChange={this.getDay} />
   {renderable}
  </div>
```
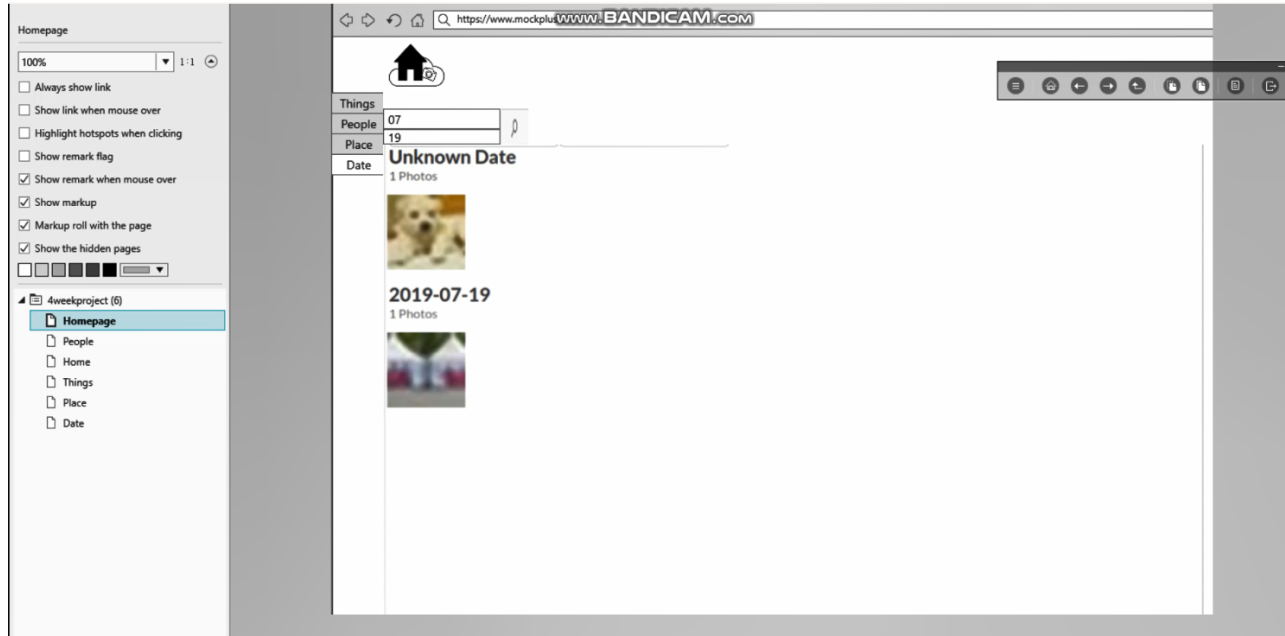
ownphoto

```
this.props.photos.map(function(photo){
  if (photo.exif_timestamp != null) {
    var date = photo.exif_timestamp.split('T')[0]
  var month = date.substring(5,7)
  var day = date.substring(8,10)

  if(! photosGroupedByDate.hasOwnProperty(date)){
    photosGroupedByDate[date] = []
  }
  if(curMonth == 13 && curDay == 32){
      photosGroupedByDate[date].push(photo)
  } else if(curMonth == month && curDay == 32){
      photosGroupedByDate[date].push(photo)
  } else if(curMonth == 13 && curDay == day){
      photosGroupedByDate[date].push(photo)
  } else if(curMonth == month && curDay == day){
      photosGroupedByDate[date].push(photo)
  }

  }
  else {
    photosGroupedByDate['Unknown Date'].push(photo)
  }
})
return photosGroupedByDate
```
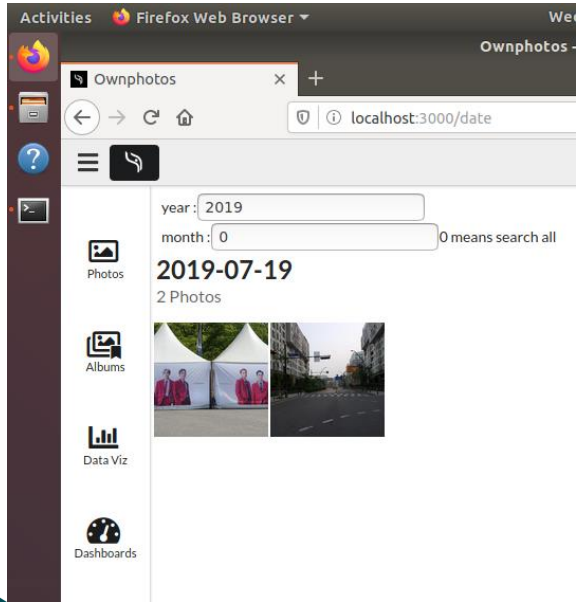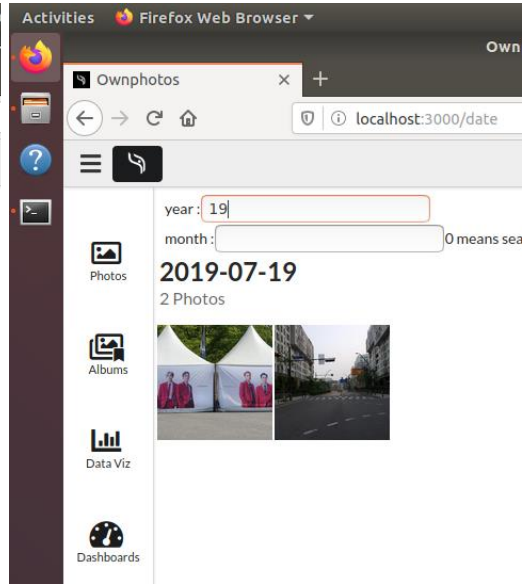
Own Photos – Frontend

- Create interface mockups

Own Photos – Frontend

- Implement search function

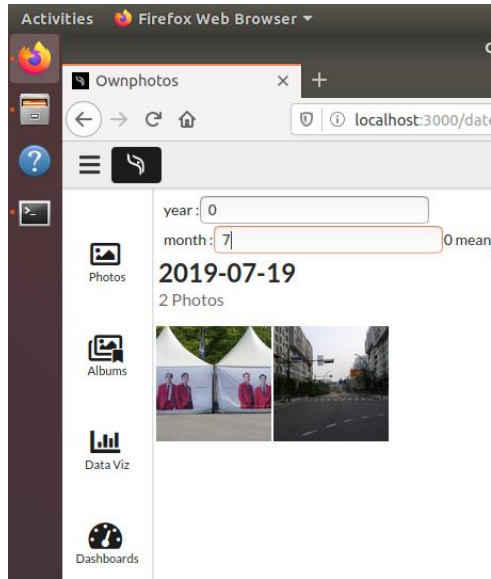Year : 2019, month : 0   Year : 19, month : null

Search Year
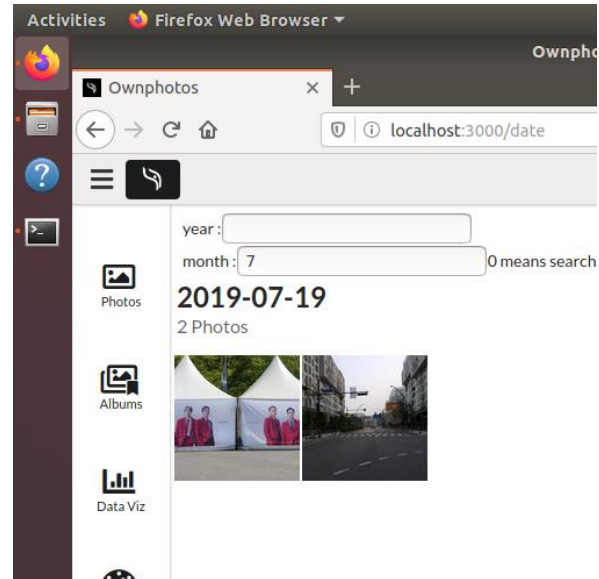
Own Photos – Frontend

- Implement search function

Year : 0, month : 7   Year : null, month : 7
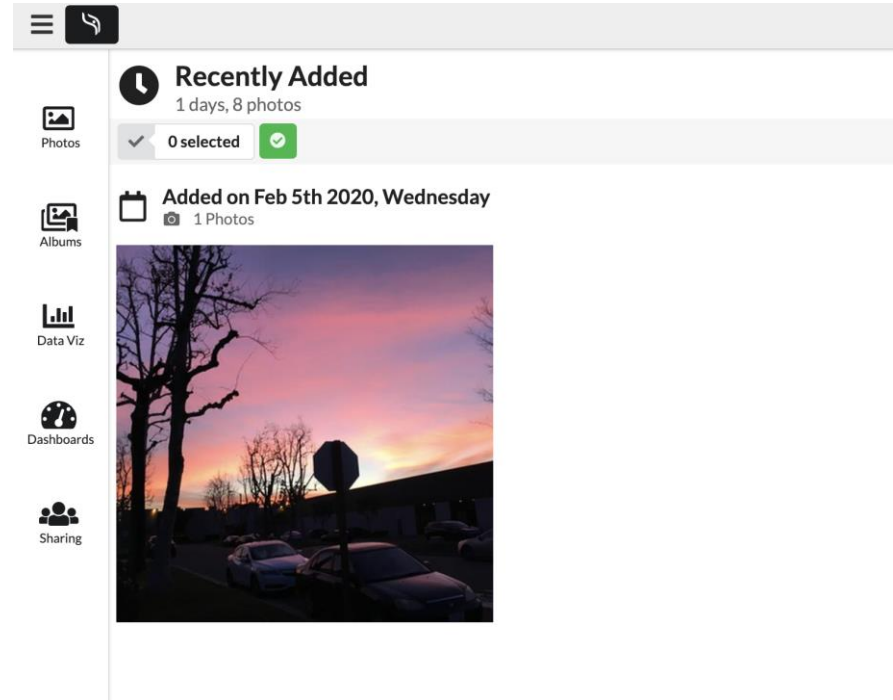
Search Month

Own Photos – Backend

- Tasks
    - Setup environment
    - Add support for HEIC files
    - Add support for Png files

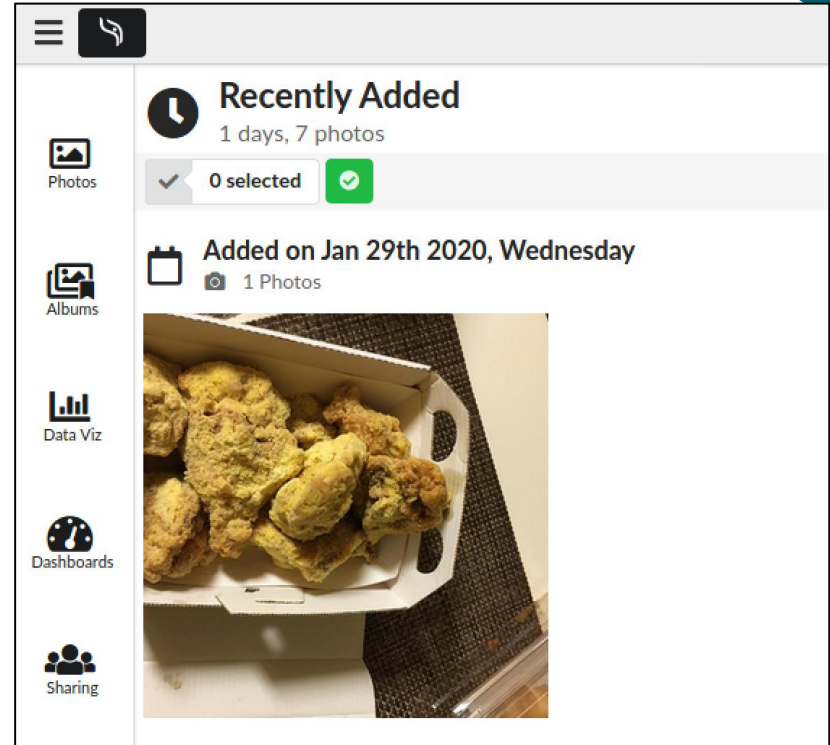# Own Photos - Backend

- Support for HEIC files

Own Photos – Backend

- Support for Png files


```
alkene@ubuntu:~/Pictures/Alkene$ ls -al
total 33796
drwxrwxrwx 2 alkene alkene    4096 Jan 30 12:29
drwxr-xr-x 3 alkene alkene    4096 Jan  7 15:16 ..
-rwxrwxrwx 1 alkene alkene 1980195 Jan  7 15:10 IMG_4613.JPG
-rw-r--r-- 1 root   root    787583 Jan 29 16:43 box.jpg
-rwxrwxrwx 1 alkene alkene 15053661 Jan 28 14:52 box.png
-rw-r--r-- 1 alkene alkene 167421 38 Jan 29 16:40 chicken.png
-rwxrwxrwx 1 alkene alkene  1922        11:18 Maxresdefault.jpg
-rw-r--r-- 1 root   root       703 Jan 28 15:05 test.py
alkene@ubuntu:~/Pictures/Alkene$ _
```

Own Photos – Mobile
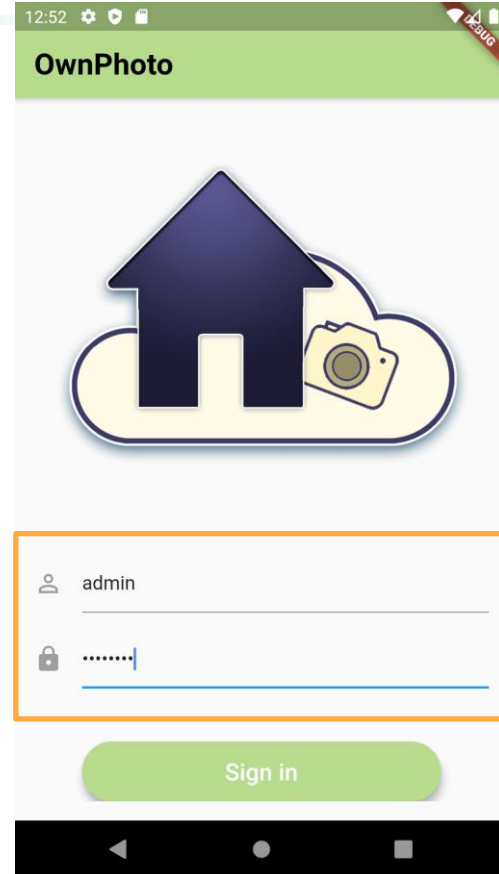
- Tasks
    - Setup flutter development environment
    - Research and map out Ownphoto API
    - Create mockups for app layout
    - Create login/list photo/see photo page
    - Create search bar
    - Add tabs for filtering by people, things

Own Photos - Mobile

- Support to sign in.

Own Photos – Frontend & Backend

DEMO

Own Photos – Mobile

DEMO

Thank you for listening