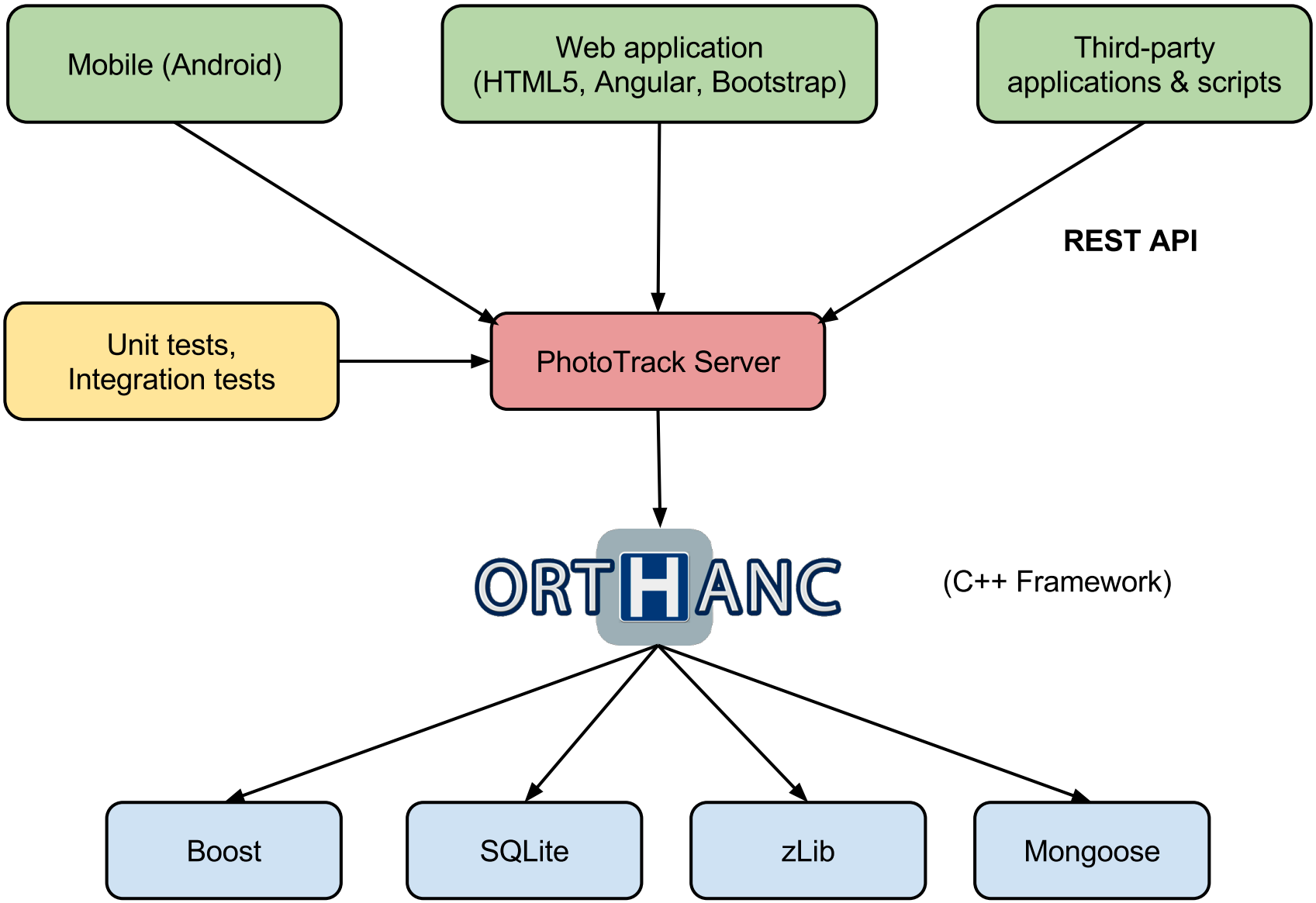
PhotoTrack

Grégory Art
Jean-François Colson
Benjamin Golinvaux
Sébastien Jodogne

August 3rd, 2014

Architecture



Going Native

Orthanc Project

- Open-source server for medical imaging (DICOM).
- Originates from University Hospital of Liège.
- A spin-off project is ongoing (incubation by WSL).

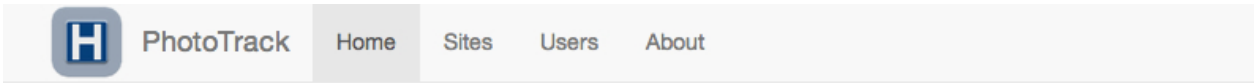
Why C++?

- Lightweight (deploy many separate PhotoTrack servers).
- Fully standalone (no framework, no DBMS).
- Cross-platform (Windows, Linux, OS X, Raspberry Pi).

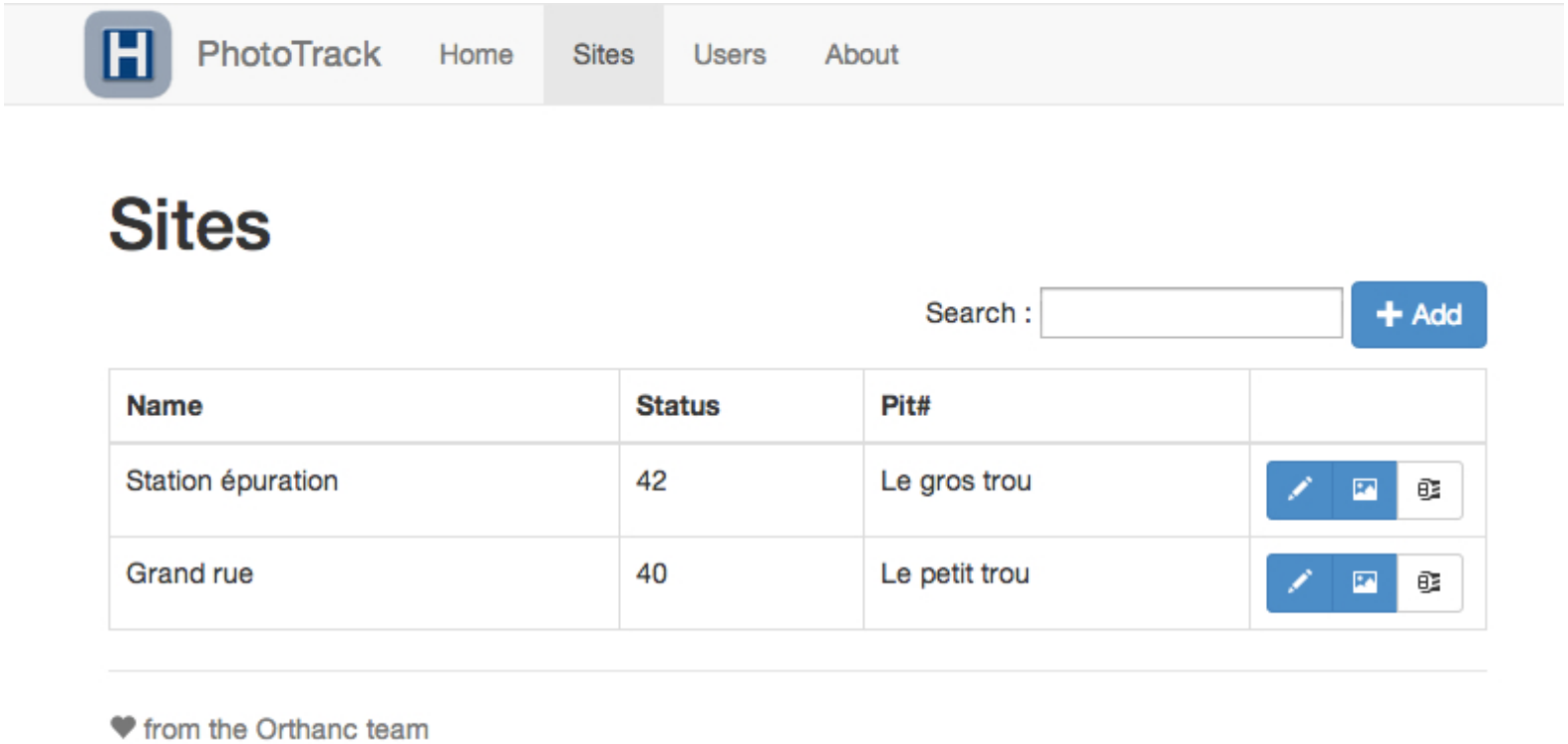
REST/JSON API in a Nutshell

	GET	POST	DELETE	PUT	
/	y	n	n	n	
/changes	y	n	n	n	"last", "limit" and "since" arguments
/photos	y	y	n	n	POST = create photo
/photos/{uuid}	y	n	y	y	
/photos/{uuid}/image	y	n	n	y	PUT = change image (MIME type comes from HTTP content-type)
/photos/{uuid}/latitude	y	n	n	n	Error 400 if no gps
/photos/{uuid}/longitude	y	n	n	n	Error 400 if no gps
/photos/{uuid}/seconds-since-epoch	y	n	n	n	
/photos/{uuid}/tag	y	n	n	n	
/sessions	t	y	n	n	POST = login
/sessions/{uuid}	y	n	y	n	
/sessions/{uuid}/username	y	n	n	n	
/sites	y	y	n	n	POST = create site
/sites/{uuid}	y	n	y	y	PUT = update, GET = JSON
/sites/{uuid}/address	y	n	n	n	
/sites/{uuid}/archive	y	n	n	n	Download ZIP file
/sites/{uuid}/latitude	y	n	n	n	Error 400 if no gps
/sites/{uuid}/longitude	y	n	n	n	Error 400 if no gps
/sites/{uuid}/name	y	n	n	n	
/sites/{uuid}/photos	y	n	n	n	
/sites/{uuid}/pit-number	y	n	n	n	
/sites/{uuid}/seconds-since-epoch	y	n	n	n	
/sites/{uuid}/status	y	n	n	n	
/users	y	y	n	n	POST = create user
/users/{uuid}	y	n	y	y	
/users/{uuid}/email	y	n	n	n	
/users/{uuid}/full-name	y	n	n	n	
/users/{uuid}/is-admin	y	n	n	n	
/users/{uuid}/is-supervisor	y	n	n	n	
/users/{uuid}/organization	y	n	n	n	
/users/{uuid}/password	t	n	n	n	
/users/{uuid}/username	y	n	n	n	

Web Application (1/4)



Web Application (2/4)



Web Application (3/4)

PhotoTrack Home Sites Users About

Photos

< Back to sites list

Download Archive

Après chantier

Jan 12, 1970

Avant chantier

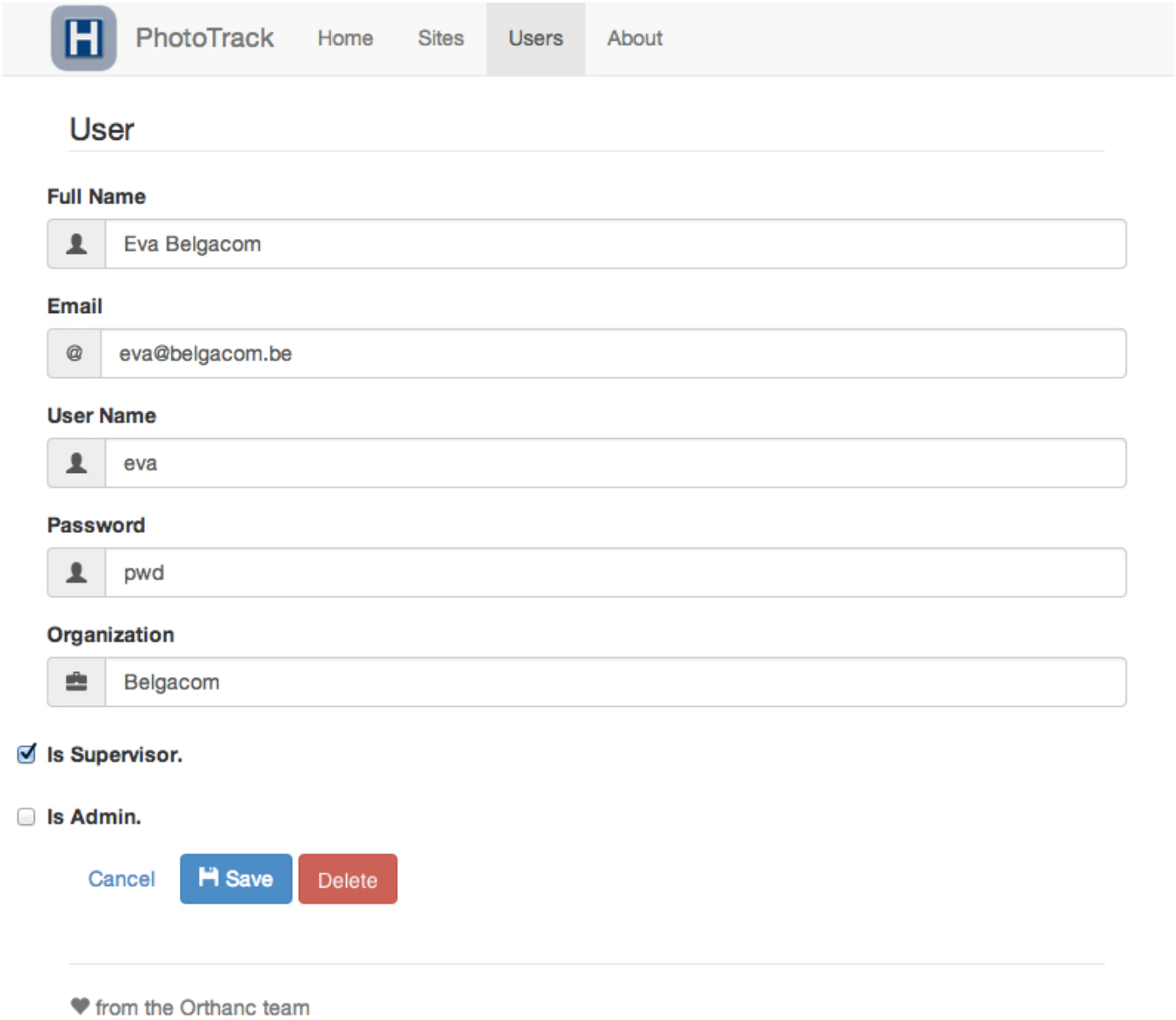
Jan 12, 1970

Current

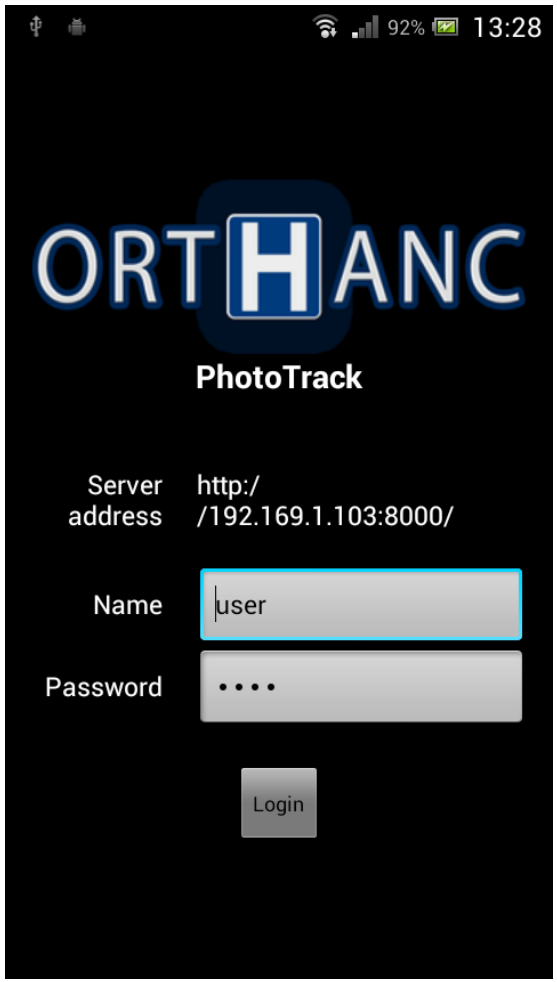
Jan 17, 1970

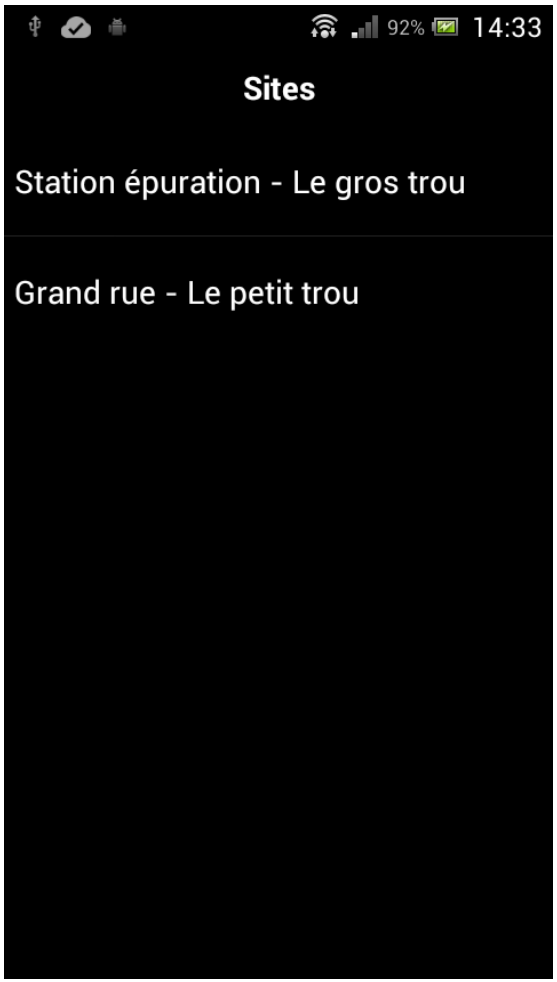
♥ from the Orthanc team

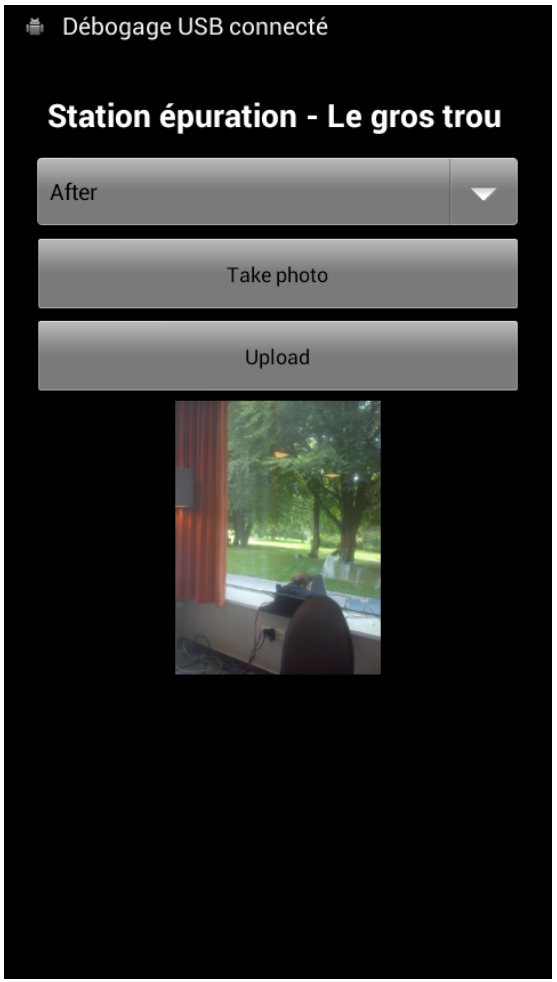
Web Application (4/4)



Mobile Application (Android)







Third-Party Scripts — “To DropBox”

```
PHOTO_TRACK_SERVER = 'http://localhost:8000'
TARGET_FOLDER = 'Dropbox'

# Polling loop waiting for the arrival of new images
current = 0
while True:
    r = DoGetJson(PHOTO_TRACK_SERVER + '/changes', { 'since' : current })

    for change in r['Changes']:
        # Retrieve the photo and the site of this change
        photoUri = PHOTO_TRACK_SERVER + '/photos/' + change['PhotoUuid']
        photo = DoGetJson(photoUri)
        site = DoGetJson(PHOTO_TRACK_SERVER + '/sites/' + photo['SiteUuid'])

        # Choose the destination path
        folder = os.path.join(TARGET_FOLDER, site['PitNumber'])
        filename = photo['Time'] + ' - ' + photo['Tag'] + '.jpg'

        # Create the destination folder
        os.makedirs(folder)

        # Retrieve and write the image
        with open(os.path.join(folder, filename), 'wb') as f:
            f.write(DoGet(photoUri + '/image'))

    current = r['Last']
    time.sleep(1)
```

Conclusions

Summary

- Approach: REST server with an open API.
- Server-side: Cross-platform, lightweight, C++.
- Client-side: Compatible with AJAX, Python, C#, Java...
- Delivrables: Web app, Android application, Python samples.
- Quality: Unit tests, integration tests, CIS, valgrind.

Perspectives

- Open-sourcing (licensing, documentation, public release).
- User management.
- Enhance robustness.
- Geocoding in mobile application.
- Connect to multiple PhotoTrack servers.