Vilniaus
universitetas

GraffLib

Faculty of Mathematics and
Informatics
Problem-based Project

Gintautas Švedas

Vilniaus
universitetas

CONTENT

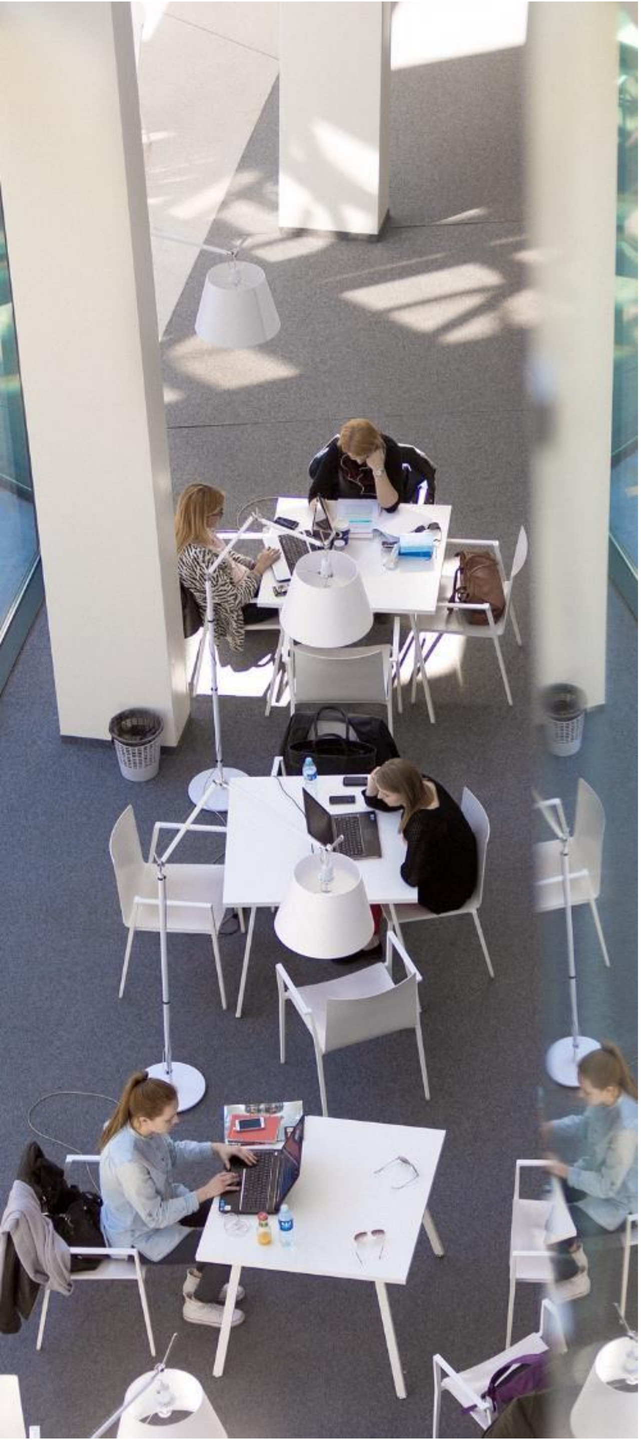
1. What is GraffLib
2. Motivation
3. Technical corner
    a. Used technologies
    b. How does it work
    c. Back-end analysis
    d. Database analysis
    e. Front-end analysis
4. Demonstration

GraffLib problem statement

- Graffiti Library. Graffiti **temporal gallery** and search for **similar graffiti**. **How many graffiti spots** are there in the town? **How do they change**?

What is GraffLib?

- This is a project about graffiti and everything it entails, beauty, horror, cities and concrete.
- This name is combined from two words GraffLib = Graffiti + Library.



Vilniaus universitetas

Motivation

Technical Corner:

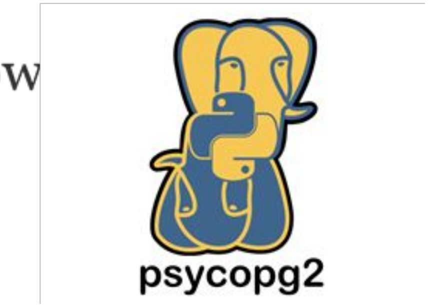
Used technologies

Used technologies - Front-end & Back-end

Front-end

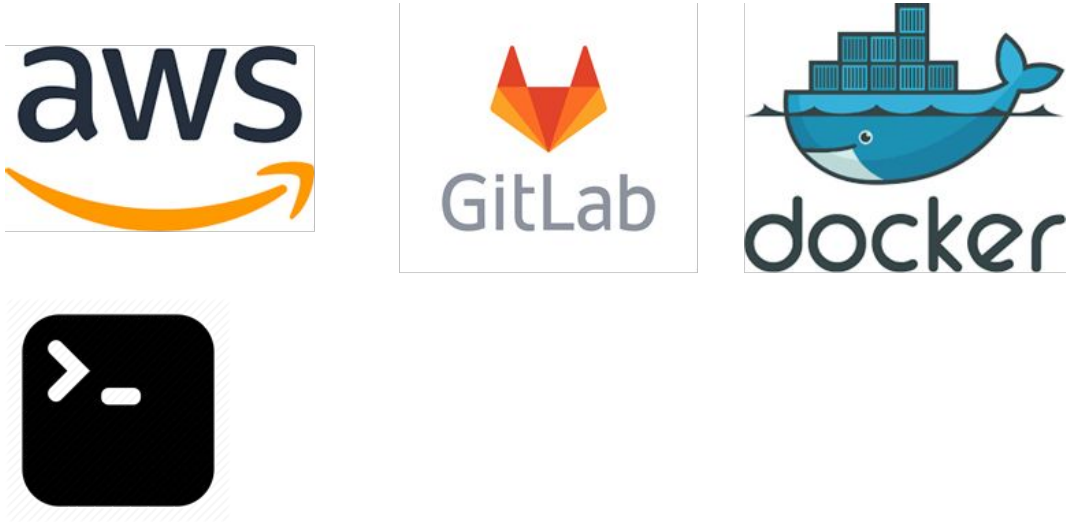
Back-end

40+ Python packages



Used technologies - DevOps & Tools

DevOps



Tools

Technical Corner:

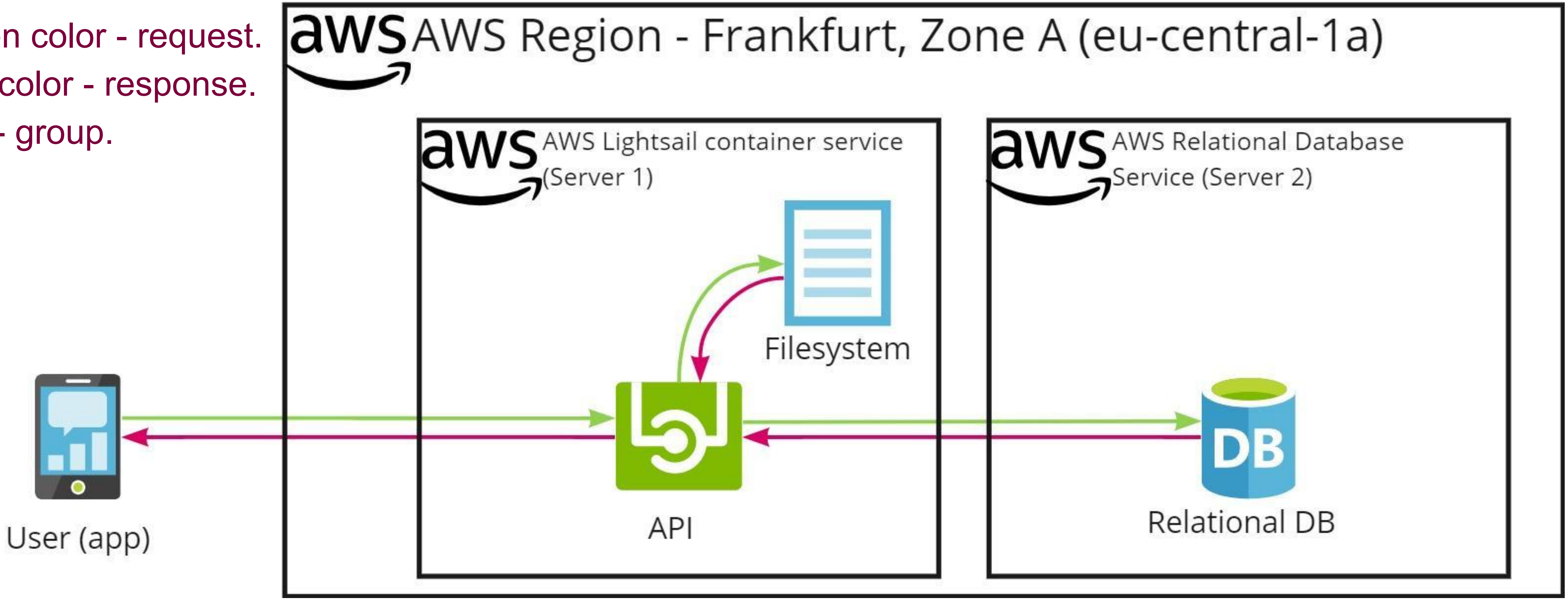
How does it work?

How does it work?

Green color - request.

Red color - response.

Box - group.


AWS Region - Frankfurt, Zone A (eu-central-1a)
AWS Lightsail container service (Server 1)
AWS Relational Database Service (Server 2)
Filesystem
API
DB
Relational DB
User (app)

Why Amazon Web Services (AWS)?

Open Nebula

Failures with setting the project up with Windows VMs.

Power

AWS provides more powerful machines.

Security

More secure.

Technical Corner:

Back-end analysis

Back-end analysis - Endpoints

20

Different endpoints are implemented.

5

Are fully used by front-end

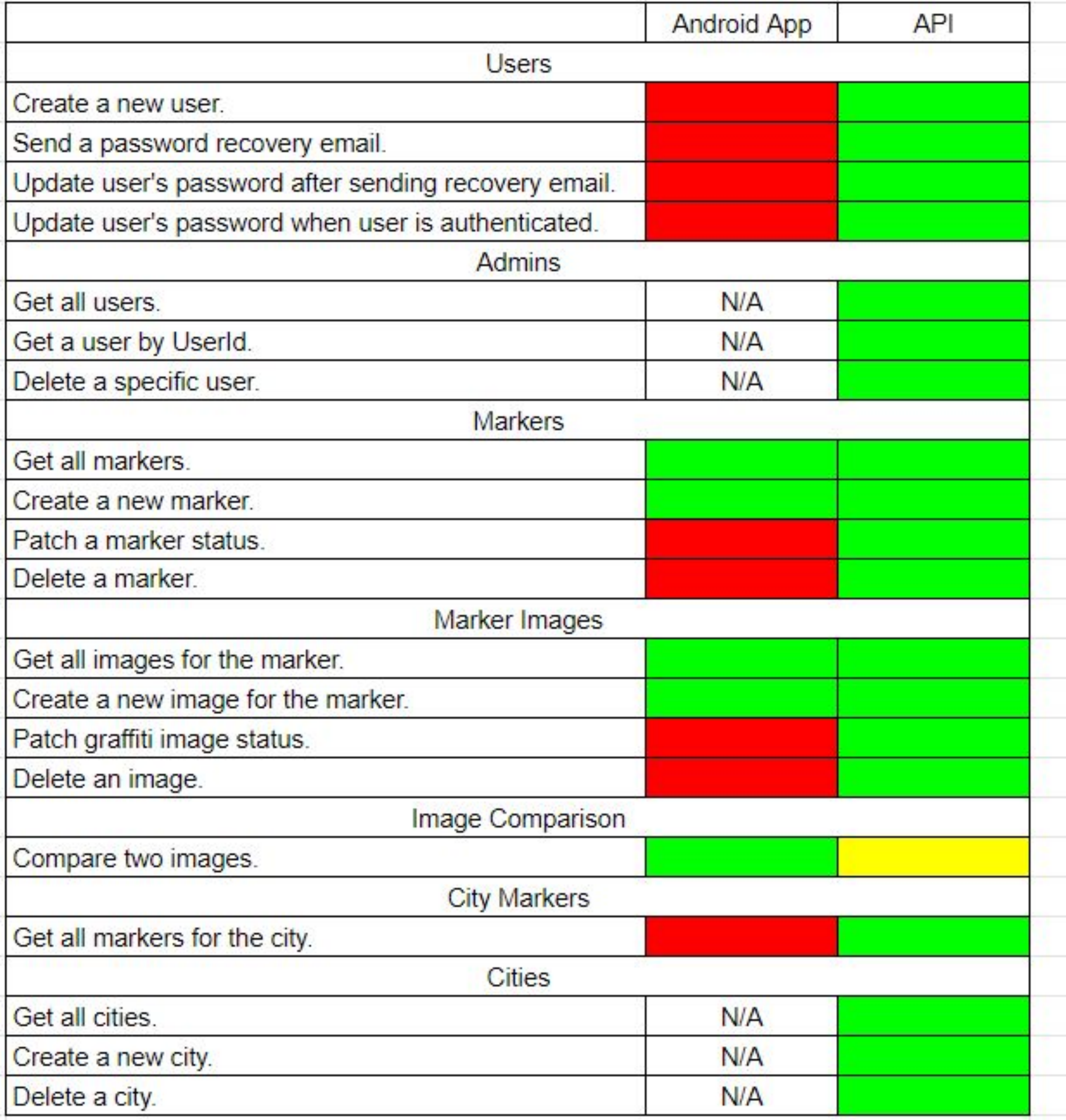
7

Different API domains.

Green color - implemented.

Red - not implemented.

Yellow - partial implementation.



	Android App	API
Users		
Create a new user.	🟥	🟩
Send a password recovery email.	🟥	🟩
Update user's password after sending recovery email.	🟥	🟩
Update user's password when user is authenticated.	🟥	🟩
Admins		
Get all users.	N/A	🟩
Get a user by UserId.	N/A	🟩
Delete a specific user.	N/A	🟩
Markers		
Get all markers.	🟩	🟩
Create a new marker.	🟩	🟩
Patch a marker status.	🟥	🟩
Delete a marker.	🟥	🟩
Marker Images		
Get all images for the marker.	🟩	🟩
Create a new image for the marker.	🟩	🟩
Patch graffiti image status.	🟥	🟩
Delete an image.	🟥	🟩
Image Comparison		
Compare two images.	🟩	🟨
City Markers		
Get all markers for the city.	🟥	🟩
Cities		
Get all cities.	N/A	🟩
Create a new city.	N/A	🟩
Delete a city.	N/A	🟩

Back-end analysis - API information

3.7

Python version.

40+

Python packages.

3

External APIs.

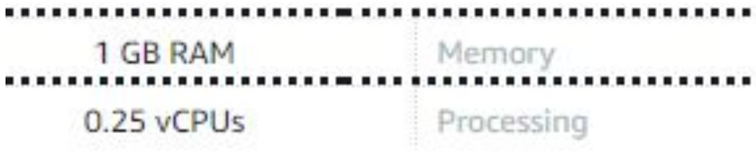
Swagger

For API documentation.

MVC

Architectural pattern.

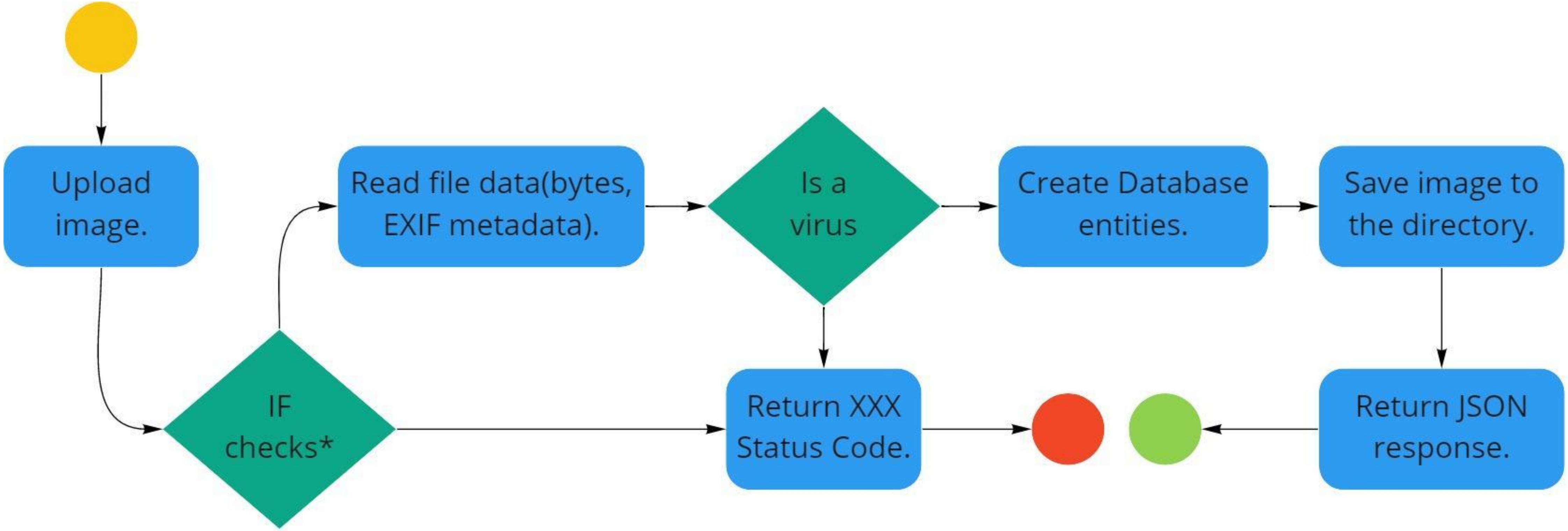
Server spec



1 GB RAM	Memory
0.25 vCPUs	Processing

Back-end analysis - Image upload

Vilniaus universitetas



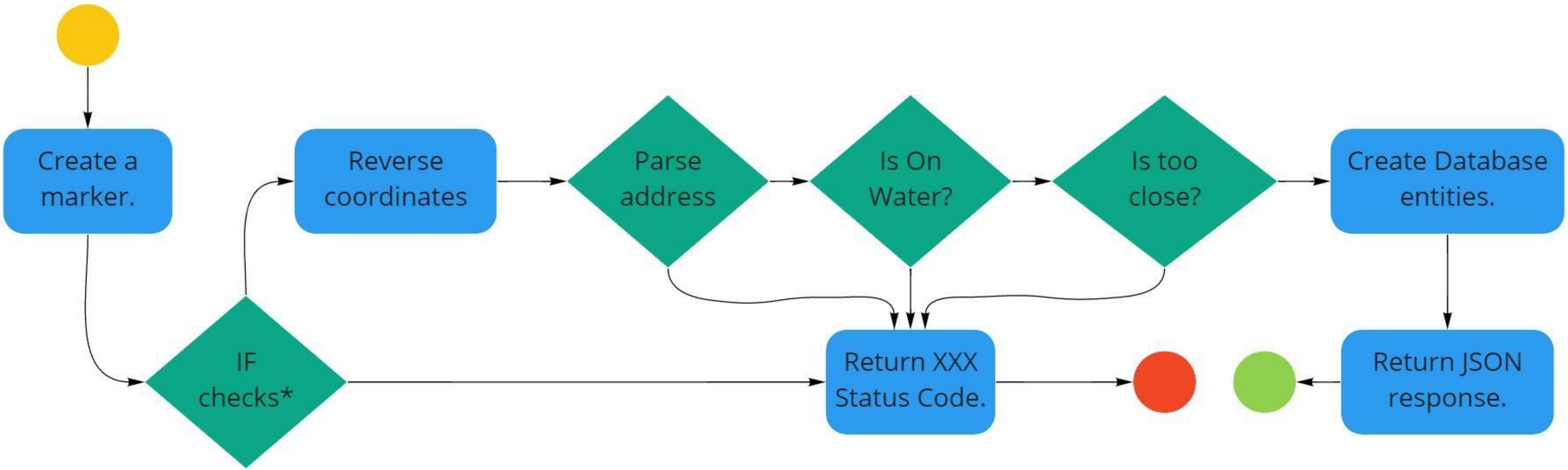
Upload image.
IF checks*
Read file data(bytes, EXIF metadata).
Is a virus
Return XXX Status Code.
Create Database entities.
Save image to the directory.
Return JSON response.

* - If image has metadata, HTTPS request correct, image size and etc).

Yellow - entry point.
Red - failure exit point.
Green - success exit point.

Back-end analysis - Create marker

Vilniaus universitetas



* - If user exists, if city exists, etc.

Yellow - entry point.
Red - failure exit point.
Green - success exit point.

Back-end analysis - Security

HTTPS

Provided by Amazon.

Hashing

Passwords via BCRYPT.

URL safe tokens

For markers and images.

Field validation

Validation for models, requests, responses.

SQL Injection

SQL Alchemy prevents SQL injection.

Technical Corner:

Database analysis

Database analysis - DB information

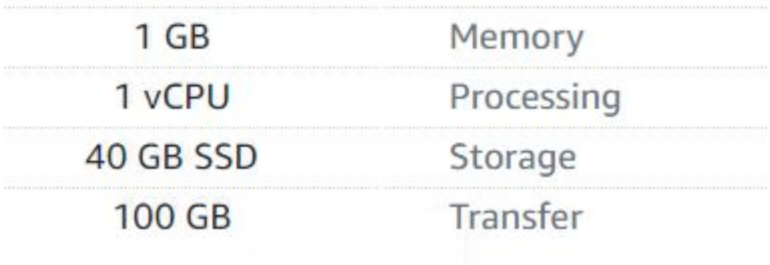
10.18

PostgreSQL version.

PostGIS

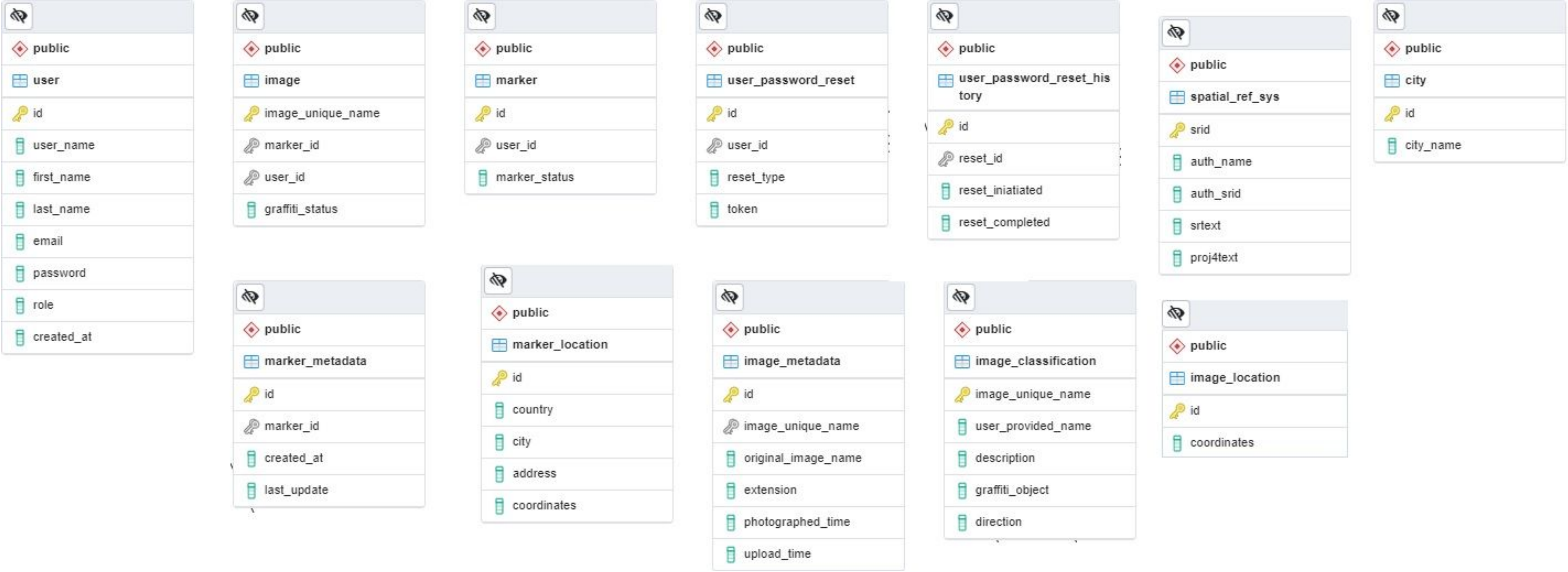
Extension for working with GIS data.

Database server



1 GB	Memory
1 vCPU	Processing
40 GB SSD	Storage
100 GB	Transfer

Database analysis - DB schema



public
user
id
user_name
first_name
last_name
email
password
role
created_at

public
image
image_unique_name
marker_id
user_id
graffiti_status

public
marker
id
user_id
marker_status

public
user_password_reset
id
user_id
reset_type
token

public
user_password_reset_history
id
reset_id
reset_iniatiated
reset_completed

public
spatial_ref_sys
srid
auth_name
auth_srid
srtext
proj4text

public
city
id
city_name

public
marker_metadata
id
marker_id
created_at
last_update

public
marker_location
id
country
city
address
coordinates

public
image_metadata
id
image_unique_name
original_image_name
extension
photographed_time
upload_time

public
image_classification
image_unique_name
user_provided_name
description
graffiti_object
direction

public
image_location
id
coordinates

Technical Corner:

Front-end analysis

Front-end analysis - App information

15

Java version.

Android 8.1

Min SDK.

Android 12

Target SDK.

Camera

Is required to have for the mobile phone.

Required permissions

- INTERNET
- ACCESS_COARSE_LOCATION
- ACCESS_FINE_LOCATION
- READ_EXTERNAL_STORAGE
- ACCESS_MEDIA_LOCATION

Front-end analysis - Activities

13

Activities exist.

9

Are fully implemented with back-end.

16

External packages are being used.

Multithreading

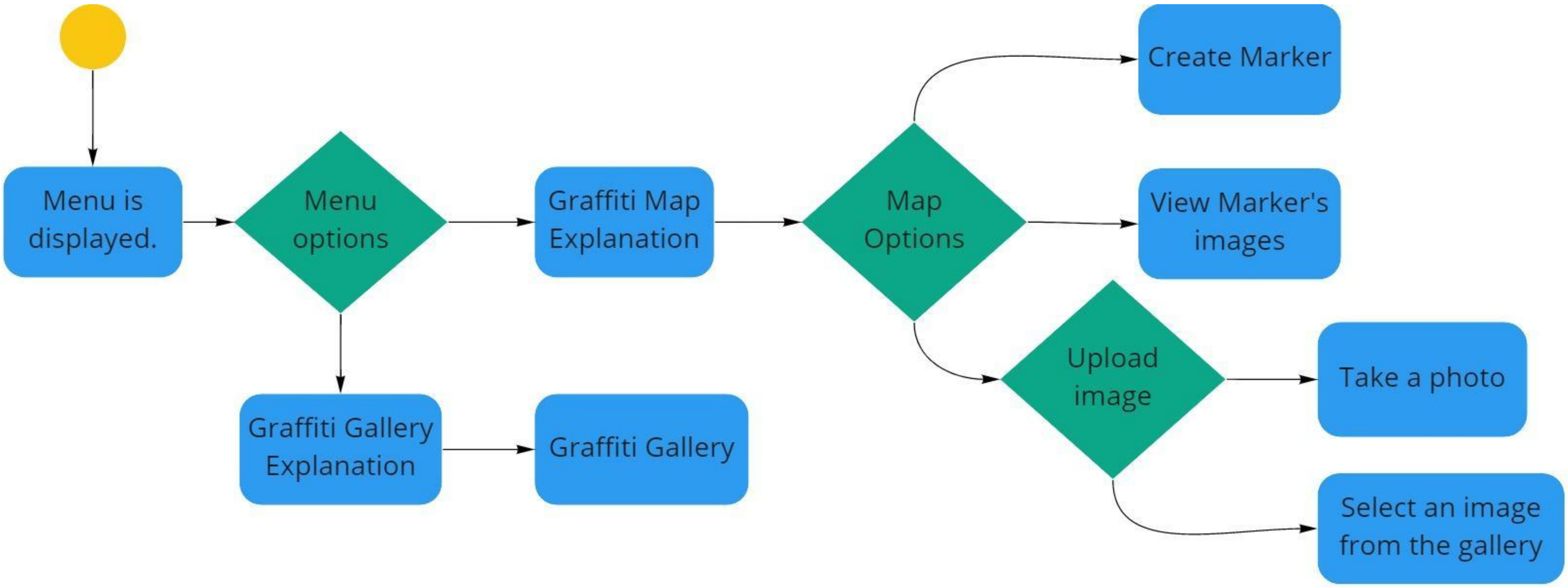
HTTPS requests and responses are performed on another thread.

Messages

User is performed about the status of the operations.

Front-end analysis - App flow

Vilniaus universitetas



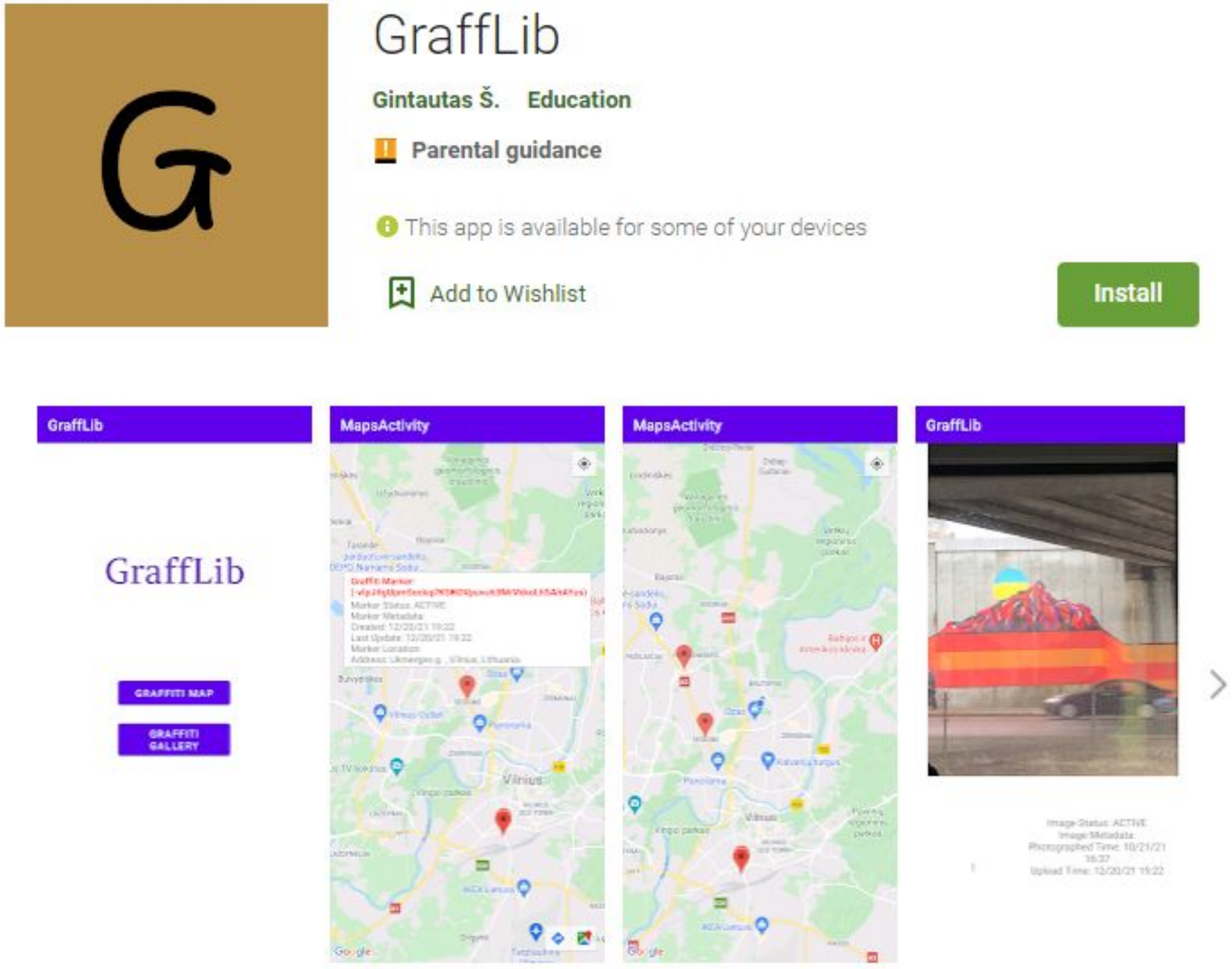
* - If user exists, if city exists, etc.

Yellow - entry point.

Front-end analysis - Google Play

GraffLib

Find it on
Google Play.



GraffLib

Gintautas Š. Education

Parental guidance

This app is available for some of your devices

Graffiti Marker:
[illegible]
Marker Status: ACTIVE
Marker Metadata:
Created: 12/20/21 19:22
Last Update: 12/20/21 19:22
Marker Location:
Address: Ukmergės g. , Vilnius, Lithuania

Image Status: ACTIVE
Image Metadata:
Photographed Time: 10/21/21 16:37
Upload Time: 12/20/21 19:22

Demonstration

Thank you for your time.