

# PROJECT PLAN

## 1. Personal information

- Title: Picture Annotator
- Creator's name: Thang Phan
- Student number: 1008713
- Study program: Economics
- Study year and date: 2021 – 2024

## 2. General description and difficulty level

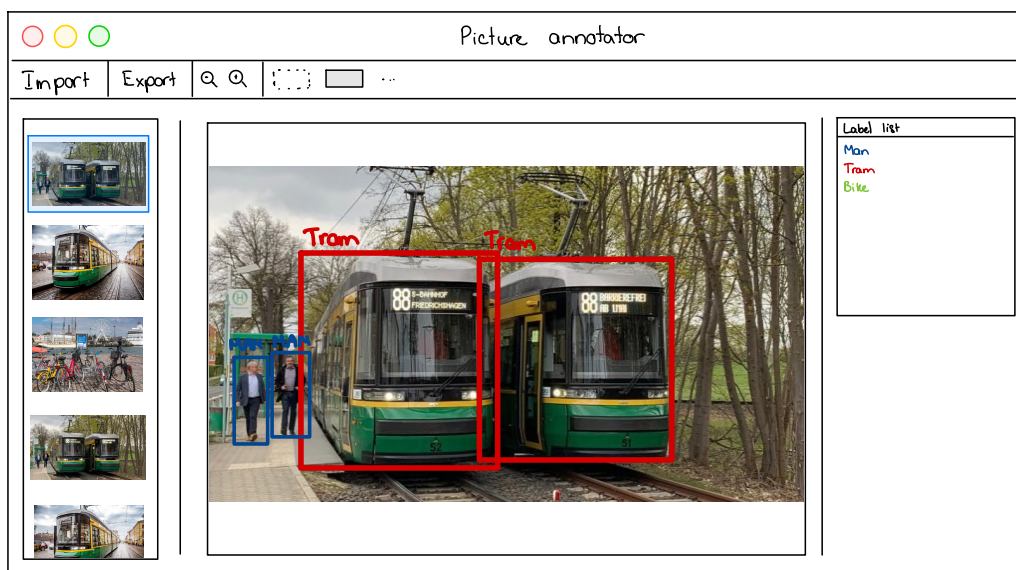
### 2.1. General description

Modern machine vision implementations are based on so-called convolution networks. These networks are created by inserting large quantities of pictures that have been annotated accordingly. Thus, the objective of this project is to create a software that can annotate pictures, which can be used for training.

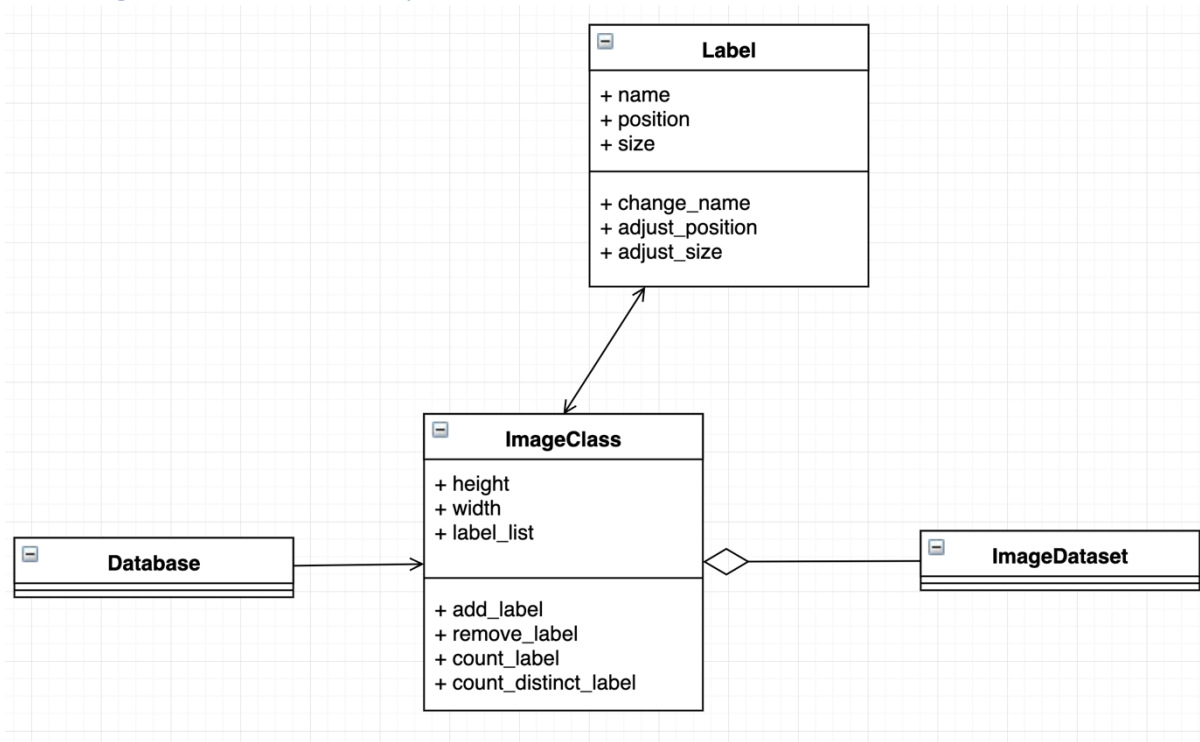
### 2.2. Difficulty level – Hard

- Drawn squares can be classified as different values.
- The software should be able to save given annotations and browse existing ones.
- There has to be a way to import and export pictures and their annotations from the software.
- Overall testing of the software. The path of the box travels through the program has to be tested.
- Unittests for at least part of the program.
- Pytorch integration for the software. From the software a Pytorch Dataset class can be created that can be used in other softwares.

## 3. Use case description and draft of the user interface



## 4. Program's structure plan



## 5. Data structures

Python's built-in data structures (list, dictionary, etc.) and possibly json.

## 6. Files and file formats

Since our program requires pictures as inputs, it should be capable of handling picture file formats such as JPEG, PNG, etc. In addition, the program might have a size limit of picture files.

The data is presented in pixel through color channels.

## 7. Algorithms

- Sorting and filtering algorithms

## 8. Testing plan

- UI test (test the functional buttons of the graphical interface such as draw boxes and labels them)
- Test import and export functionalities.
- Test database functionality.
- Unittest for at least part of the program.

## 9. Libraries and other tools

- PyQt5: Create graphical user interface
- Pytorch and torchvision: Create dataset
- Json: Store classified label
- Pillow: Process images

- PostgreSQL: Database

## 10. Schedule

- 24.02: Upload and return the project plan.
- 03.03: Project guidance meeting.
- 24.03: First checkpoint. Design the database, define the classes.
- 14.04: Second checkpoint. Create UI and add unittests.
- 11.05: Add documentation and submit the final project.
- 19.05: Present the project demos.

## 11. Literature references and links

- Course material: <https://plus.cs.aalto.fi/y2/2023/>
- PyQt5: <https://www.riverbankcomputing.com/software/pyqt/>
- Pytorch: <https://pytorch.org/>
- torchvision: <https://github.com/pytorch/vision>
- pillow: <https://pillow.readthedocs.io/en/stable/>
- PostgreSQL: <https://www.postgresql.org/>

## 12. Attachments