FH University of Applied Sciences TECHNIKUM WIEN

Documentation & Project Diary

Innovation Lab 2
Year 2022

Project: **Visitor-Counter (the Observer)**

Team: **Group 29**

1. General Information

Project name: Visitor-Counter (the Observer)

Supervisor: Lukas Rohatsch, MSc

Innovation Lab 2, 2022

Projectteam:

Varga Lukas,	if20b167@technikum-wien.at,	project manager
Düx Stefan,	if20b245@technikum-wien.at,	
Görög Jessica Isabella,	if20b094@technikum-wien.at,	
Grabner Dominic,	if20b219@technikum-wien.at,	
Tscheppen Rebekka,	if20b164@technikum-wien.at,	

Management Summary of the Project

This project is about setting up a camera module at the entry of the Presentation Lab (located: Bx.yz) to count visitors and deliver data categorized to different data packages (total, daily, current and so forth). To deliver this project, hardware is necessary to be set up correctly (central processing unit, camera, body, holding), combined with efficient software (control the hardware) and a webpage to enable access of the data. Furthermore, this project shall serve as a preparation for more advanced projects like analyzing traffic data, rate of flow and similar purposes.

Framework Conditions and Project Environment

Programming languages:
The project assignment demands to use python for processing the data. As a commonly used programming language, currently no other programming language is considered for further use regarding data processing. For the webpage the team will use the framework Angular.

Usability:
There are 2 access points for the client and / or users. The webpage is enabled for open access which includes all users with access to the internet. Therefore, the usability aims for beginner-level access. The local access directly connected to the visitor counter will be especially designed for the client.

Interfaces:
Data transfer by JSON.
wait for details regarding webspace

Standards:
It is necessary to respect the data protection rules as we enable open access to the collected data by our built webpage. Therefore, a livestream of the set up camera will only be accessible at the local console, where the module is located. The webpage will only display the processed data without any pictures or livestreams.
Using own made videos or demo videos to test or logic programming.

Deadlines:
Prototype for end of Semester #4 (demo-version for project)

currently no further details identified

Semester-Roadmap

1. semester	Focus: Raspberry Pi **1. sprint** install Raspberry Pi OS and configure packages, install camera module **2. sprint** ensure requirements (python and required packages) **3. sprint** expand python libraries **4. sprint** implement system for visitor counter **5. sprint** final capture of visitor **6. sprint** prepare presentation
2. semester	Focus: Raspberry Pi / Interface **7. sprint** test phase of last streams / buffer for bug fixes **8. sprint** camera stream over HDMI **9. sprint** test video capture of visitors on site **10. sprint** install and configure mgtt **11. sprint** provide and test interface **12. sprint** bug fixing **13. sprint** prepare presentation
3. semester	Focus: Interface/Website **14. sprint** plan website and install webserver **15. sprint** implement website **16. sprint** finalize website – Raspberry Pi – connection **17. sprint** bug fixes **18. sprint** bug fixes **19. sprint** prepare presentation / end project

Collaboration & Tooling

Communication: Discord & Zoom
FH Technikum Wien - Moodle

Documentation: Jira Software
https://visito-counter29.atlassian.net/jira/software/projects/TO29/boards/1

Timetable: Google Sheets
https://docs.google.com/spreadsheets/d/1ep5FswrHorSOHT6Ck-h5r6ZSO9rwkWMd2KY0pQrVxrQ/edit#gid=1398090908

Tracking Work: Jira Software
https://visito-counter29.atlassian.net/jira/software/projects/TO29/boards/1

Source Code Management: GitHub
GitHub - unravelet/theObserver: Innovation Lab Project - Visitor Counter

Screen Prototyping: Mockplus
No project currently existing

Modelling: Modelio
No project currently existing

Remarks

--

2. Brief Description of the Project

The **main object** for the project is a visitor counter which will be located at the “Presentation Lab” room to count the number of visitors. For the counting a camera module will be used in combination with a software to detect and count people from a livestream. This functionality can be used as an example to display or even regulate a maximum number of people inside for health (Covid) or security reasons (events).

As the **biggest challenge**, the project group will face the needs of learning the necessary skills (logic programming with python) and mechanical capabilities (working with hardware like Raspberry Pi, camera module and 3D printer).

The **usability object** will be the easy access to the data and controlling the interfaces at beginner level (local livestream or processed data via webpage).

The **non-objective** of the project is involving face recognition and tracking.

As for the **scope**, the project covers the functionality to process data from a livestream to calculate the number of people (staying inside a room or entering/leaving) and to display it on a local console + webpage with free access. For implementing the software, we need to know about the specifications necessary for the Raspberry Pi and will orientate on similar projects / products. Additionally, we need to learn about video-recognition software to get the necessary skills.

ORDER model questions:

Opportunity - Development of visitor counter for room “Presentation Lab” of campus FH Technikum Wien featuring people recognition to count visitors and displaying data via webpage and local livestream.

Resources - Hardware / Webspace / Software (Libraries)

- Raspberry Pi + camera module (sponsored)
- display for local livestream (sponsored)
- host + domain (sponsored)
- python libraries

Decision Process - guaranteed anonymity

Exact Solution - High complexity implies risk of failing the project.

Relationship - Project is happening.

3. Specification of the Solution

Sprint 1

Basic description of sprint #1:

For the first sprint, all group members will check the hardware for existing problems or damage and functionality. Fitting software / operating system will be chosen and installed. An official name for the project is going to be set by brainstorming to use it for the GitHub repository and the webpage domain later on. The trello board will be filled up with tasks and an IDE chosen.

Time:

start (sprint #1): 12.10.2021, 17:40:00
deadline (protocol): 10.11.2021, 23:59:00
deadline (sprint #1): 11.11.2021, 16:09:59

Content:

bug fixes [#1reqID]

- no bugs at start

software [#1reqID]

- #1 check installed OS and think about need of alternative OS
- #2 install packages (SSH)
- #3 test camera module by live recording
- #4 choose and set up IDE

hardware [#1reqID]

- #5 check quality of all hardware components
- #6 test functionality of all hardware components

documentation [#1reqID4_]

- #7 fill trello board with tasks (backlog)
- #8 specify name of project in preparation for GitHub repository + web domain
- #9 add “3D print” as task in semester roadmap
- #10 generate 3D model of PresentationLab for overview
- #11 add pictures to notes
- #12 plan switch to Jira Collaboration Tool

Assignment:

- Dominic, Jessi + Lukas => documentation (total)
- Stefan + Rebekka => software (total)
- whole team=>hardware (total)

Sprint 2

Basic description of sprint #2:

For the second sprint the group will ensure requirements, try out to use Jira collaboration tool (past sprint has shown potential of difficult information handling by Google Docs and Trello for a project volume of 1,5 years. A repository will be generated + domain name for webpage defined in documentation and the 3D case print for the Raspberry Pi (camera module inclusively) planned.

Time:

start (sprint #2): 11.11.2021, 17:40:01
deadline (protocol): 24.11.2021, 23:59:00
deadline (sprint #2): 25.11.2021, 16:09:59

Content:

bug fixes [#2reqID1_]

- #1 swap PoE because of malfunctions

software [#2reqID2_]

- #2 install chosen IDE
- #3 search for suitable packages / libraries
- #4 install new packages
- #5 test new packages

hardware [#2reqID3_]

documentation [#2reqID4_]

- #6 update Trello tasks
- #7 create GitHub repository
- #8 upload all documentation and other elements to repository
- #9 determine constant date for weekly sprint meeting
- #10 embed Jira
- #11 3D model case Raspberry Pi planning (+ PoE, + camera module)
- #12 create Milestones for Trello

Assignment:

- Rebekka (GitHub + Trello tasks, Milestones)
- Dominic + Lukas (3D Planung + Jira)
- Jessica + Stefan (Packages + Libraries)
- Team (IDE, Libraries)

Sprint 3

Basic description of sprint #3:

For the third sprint the group will expand the python libraries, proceed with the planning and realizing of the 3D model case. Furthermore, the change in documentation from Google Docs and Trello to Jira will be realized and every team member uses the sprint to get in touch with the different collaboration tool.

Time:

start (sprint #3): 25.11.2021, 17:40:01
deadline (protocol): 08.12.2021, 23:59:00
deadline (sprint #3): 09.12.2021, 16:09:59

Content:

bug fixes [#3reqID1_]

- #01 clarify status of replacement for PoE module

software [#3reqID2_]

- #02 adapt missing libraries / packages
- #03 test adapted libraries / packages
- #04 tutorium of first steps for rest of team members
- #05 test and use new packages (no hardware nec. for testing with dummy data
- #06 generate test video for dummy data in PresentationLab

hardware [#3reqID3_]

- #07 print 3D hardware case

documentation [#3reqID4_]

- #08 update Jira tasks
- #09 connect GitHub repository
- #10 continue communication with ProjectKitchen
- #11 start plan and execute printing process for 3D case

Assignment:

- Jessica + Stefan (02, 03, 04)
- Dominic + Rebekka + Lukas (07, 10, 11)
- Lukas (01)
- Rebekka (08)
- Alle (04, 05, 06, 09)

Sprint 4

Basic description of sprint #4:

For the fourth sprint the group will implement the system for visitor counter, decide and plan next steps for 3D print. First thoughts about final phase of semester will be topic in discussions and planning on what to do for the upcoming sprints regarding the end of semester #3.

Time:

start (sprint #4): 09.12.2021, 16:45:01
deadline (protocol): 15.12.2021, 23:59:00
deadline (sprint #4): 16.12.2021, 16:09:59

Content:

bug fixes [#4reqID1_]

- #01 library PIL could not be installed

software [#4reqID2_]

- #02 use demo videos and generate data from it
- #03 test / use new packages in team

hardware [#4reqID3_]

documentation [#4reqID4_]

- #04 update Jira tasks
- #05 update GitHub repository
- #06 discuss next steps with supervisor for 3D print
- #07 answer ProjectKitchen about next steps for 3D print
- #08 discussions about what to do in preparation for end of semester #3

Assignment:

- Team (02, 03, 08)
- Rebekka, Lukas, Dominic (06, 07)
- Rebekka (04)
- Jessica, Stefan (01)

Sprint 5

Basic description of sprint #5:

For the fifth sprint the group will take time to focus on the main functionality of recognizing people from pre-defined videos. Additional software to adapt from computervision.zone will be used and python project added to the repository. An additional version to use person recognition from livestream will be tried to implement. Furthermore a prototype will be generated to use for the final presentation of the semester.

Time:

start (sprint #5): 16.12.2021, 16:45:01
deadline (protocol): 12.01.2022, 23:59:00
deadline (sprint #5): 13.01.2022, 16:09:59

Content:

bug fixes [#5reqID1_]

- #01 fix problem with current person recognition in terms of speed
- #02 fix problem with current person recognition in terms of recognition

software [#5reqID2_]

- #03 adapt software for better recognition
- #04 adapt software for useful display of output (text in video)
- #05 improve software by adding from computervision.zone (gestic recognition etc)

hardware [#5reqID3_]

documentation [#5reqID4_]

- #06 update Jira tasks
- #07 update GitHub repository
- #08 add python project to repository
- #09 create branches on repository to work with the team
- #10 remind ProjectKitchen of email communication to clarify current status of 3D print

Assignment:

- Team (03, 04, 05)
- Rebekka (06, 08, 09)
- Lukas und Dominic (10)
- Jessica und Stefan (01, 02)

Sprint 6

Basic description of sprint #6:

The sixth and final sprint for the semester will focus on documentation only. After the documents are summarized to one project diary, a video of the documentation and working python project will be generated and uploaded.

Time:

start (sprint #6): 13.01.2022, 16:00:01
deadline (protocol): 19.01.2022, 23:59:00
deadline (sprint #6): 20.01.2022, 15:14:59

Content:

bug fixes [#6reqID1_]

- #01 print correct number of people detected
- #02 adapt position of camera for full recognition of software

software [#6reqID2_]

hardware [#6reqID3_]

documentation [#6reqID4_]

- #03 update Jira tasks
- #04 update GitHub repository
- #05 answer ProjectKitchen for 3D print in presence
- #06 generate video for “final” presentation (including documentation + python code)
- #07 edit/sum up documentation for “final” hand-in

Assignment:

- Stefan, Jessica, Dominic (#04, #06)
- Rebekka, Lukas (#01, #02, #03, #05, #07)

Sprint 7

Basic description of sprint #7:

The 7th sprint indicates the start of the 4th semester and second semester of time working on the project. For this initial phase the group decided to start working on 2 mockups (prototype) for the user interface, adapt and research for better use of the software regarding scope of person recognition and counting daily or current number of visitors and contacting ProjectKitchen in regards of 3D print.

Time:

start (sprint #7): 18.03.2022, 13:35:01
deadline (protocol): 29.03.2022, 23:59:00
deadline (sprint #7): 01.04.2022, 12:04:59

Content:

bug fixes [#7reqID1_]

software [#7reqID2_]

- #01 browse other software to recognize a person within a room (out of scope/back)?
- #02 discuss if necessary to differ between number of daily visitors or current visitors
- #03 adapt software to recognize person with whole body or only torso/etc

hardware [#7reqID3_]

documentation [#7reqID4_]

- #04 update Jira tasks
- #05 update GitHub repository
- #06 write ProjectKitchen for 3D print in presence
- #07 write mail to Supervisor to confirm date and time of next meeting
- #08 choose fitting software to create mockups
- #09 create 2 mockups to display a user interface for Presentation Lab
- #10 discuss with supervisor of whom to fill in the Businessvalue of T-Shirt sizing at effort estimation sheet

Assignment:

- Stefan, Lukas und Jessica (#01, 02, 03)
- Dominic und Rebekka (#08, 09)
- Lukas (#04, 05, 06, 07)

Sprint 8

Basic description of sprint #8:

The 8th sprint sums up our work with finalizing the software for human detection while deciding for the final version and doing some bugfixes. One of the 2 created mockups will be chosen to get a final design (supervisor will give some input for further functionality) and prepared to be used for the implementation and presentation at the end of the current semester. The time effort estimation will be redone with the DAGoPERT sheet and other administration tasks like Jira, GitHub and mail communication on the regular basis. Finally the Raspberry Pi will get the software implemented and connected with the hardware (hdmi cable, camera module) and tested.

Time:

start (sprint #8): 01.04.2022, 12:50:01
deadline (protocol): 19.04.2022, 23:59:00
deadline (sprint #8): 21.04.2022, 12:04:59

Content:

bug fixes [#8reqID1_]

- #01 new versions of human detection need fixing in orientation of recognizing shapes

software [#8reqID2_]

- #02 decide for final version of software and present to supervisor at upcoming meeting
- #03 discuss further functionality of user interface for local use of raspberry pi

hardware [#8reqID3_]

- #04 implement final software of human detection @raspberry pi 4
- #05 test automatic start of software within start of system
- #06 connect hardware (camera + raspberry pi) and test current software
- #06_2 try NoIR module for Raspberry Pi 4
- #06_3 browse information for alternative camera modules for usage and mail supervisor if needed (necessary adaption)

documentation [#8reqID4_]

- #07 update Jira tasks
- #08 update GitHub repository
- #09 contact ProjectKitchen again for appointment of 3D print
- #10 redo time effort estimation via DAGoPERT
- #11 add Tshirt Sizing for time effort estimation in DAGoPERT sheet file
- #12 choose one of the created mockups for user interface @PresentationLab
- #13 finalize mockup for user interface (adapt & finish)
- #14 make appointment with supervisor for upcoming meeting

Assignment:

- Dominic, Rebekka (#12, #13)
- Stefan, Jessica, (#01)
- Stefan, Jessica, Lukas (#02, #03)
- Stefan, Jessica, Dominic, Rebekka, Lukas (#04, #05, #06, #10, #11)
- Lukas (#07, #08, #09, #14)

Sprint 9

Basic description of sprint #9:

In the 9th sprint the Raspberry Pi will get the software implemented and connected with the hardware (hdmi cable, camera module) and tested live. Project Kitchen appointment will finally be set and 3d case printed. Possible software problems will be fixed if hardware connection and/or live demo fails.

Time:

start (sprint #9): 21.04.2022, 13:45:01
deadline (protocol): 03.05.2022, 23:59:00
deadline (sprint #9): 03.05.2022, 23:59:00

Content:

bug fixes [#9reqID1_]

- #01 fix possible problems while connecting to hardware / testing live

software [#9reqID2_]

hardware [#9reqID3_]

- #02 implement final software of human detection @raspberry pi 4
- #03 test automatic start of software within start of system (linux command anlegen um als property zu verwenden)
- #04 connect hardware (camera + raspberry pi) and test current software
- #05 test setup live

documentation [#9reqID4_]

- #06 update Jira tasks
- #07 update GitHub repository
- #08 contact ProjectKitchen again for appointment of 3D print

Assignment:

Sprint 10

Basic description of sprint #10:

In the 10th sprint the Raspberry Pi will get adapted in the auto start and the mqtt installed/configured. The appointment for the ProjectKitchen will finally be fixed and the 3D case printed. Additionally, a live demo will happen at the location of the Presentation Lab.

Time:

start (sprint #10): 05.05.2022, 16:45:01
deadline (protocol): 17.05.2022, 23:59:00
deadline (sprint #10): 17.05.2022, 23:59:00

Content:

bug fixes [#10reqID1_]

- #01 fix possible problems while connecting to hardware / testing live
- #02 fix multithreading on pi
- #03 fix autostart problem

software [#10reqID2_]

- #04 install mqtt
- #05 configure mqtt
- #06 fine tuning auto start

hardware [#10reqID3_]

- #07 live demo at location

documentation [#10reqID4_]

- #08 update Jira tasks
- #09 update GitHub repository
- #10 meet at ProjectKitchen and print case
- #11 fill in DAGoPERT T-Shirt sizing for time effort estimation

Assignment:

- Team (#01, 02, 03, 04, 05, 06, 07, 09)
- Rebekka (#02, 08, 09)
- Lukas (#10)

Sprint 11

Basic description of sprint #11:

In the 11th sprint the Raspberry Pi will finally get its autostart and mutlithreading, because the effort was higher than anticipated. Additionally, the already configured mqtt software component will be added to the pi. The appointment for the ProjectKitchen will finally be fixed and the 3D case printed, after another delay. With the implemented interface a live demo with the group will happen.

Time:

start (sprint #11): 17.05.2022, 13:15:01
deadline (protocol): 31.05.2022, 23:59:00
deadline (sprint #11): 31.05.2022, 23:59:00

Content:

bug fixes [#11reqID1_]

- #01 fix multithreading on pi
- #02 fix autostart problem

software [#11reqID2_]

- #03 get mqtt software component to raspberry pi
- #04 fine tuning auto start
- #05 install interface by blueprint
- #06 test interface functionality

hardware [#11reqID3_]

- #07 live demo with project group

documentation [#11reqID4_]

- #08 update Jira tasks
- #09 update GitHub repository
- #10 meet at ProjectKitchen and print case

Assignment:

- Rebekka, Stefan (#01)
- Jessica, Lukas, Dominic (#05, 06)
- Lukas, Jessica (#02, 03, 04)
- Rebekka (#08, 09)
- Lukas (#10)
- Group (#07)

Sprint 12

Basic description of sprint #12:

In the 12th sprint all the missed or not fulfilled tasks from past sprints will be fixed and finally implemented. The appointment for the ProjectKitchen will finally be fixed and the 3D case printed, after another delay. With the implemented interface a live demo with the group will happen.

Time:

start (sprint #12): 01.06.2022, 19:45:01
deadline (protocol): 14.06.2022, 23:59:00
deadline (sprint #12): 14.06.2022, 23:59:00

Content:

bug fixes [#12reqID1_]

- #01 fix multithreading on pi
- #02 fix autostart problem
- #03 implement interface

software [#12reqID2_]

- #04 fine tuning auto start
- #05 test interface functionality

hardware [#12reqID3_]

- #06 live demo with project group

documentation [#12reqID4_]

- #07 update Jira tasks
- #08 update GitHub repository
- #09 appointment ProjectKitchen

Assignment:

Sprint 13

Basic description of sprint #13:

In the 13th sprint it is all about completing the missing parts, creating a video for the necessary content to upload and the regarding documentation too.

Time:

start (sprint #13): 15.06.2022, 16:45:01
deadline (protocol): 28.06.2022, 23:59:00
deadline (sprint #13): 28.06.2022, 23:59:00

Content:

bug fixes [#13reqID1_]

- #01 implement interface

software [#13reqID2_]

- #02 test interface functionality

hardware [#13reqID3_]

documentation [#13reqID4_]

- #03 update Jira tasks
- #04 update GitHub repository
- #05 3D print
- #06 generate video for semester upload
- #07 prepare / complete documentation for semester upload

Assignment:

- Rebekka (#03, #07)
- Jessica, Stefan, Lukas, Dominic (#06)
- Group (#04)

Backlog - (Sprints 14-20)

Epic	
User Story	Sub Task
TO29-36 - 1.General Administrative	
TO29-340 - Paper	TO29-345 - Identify structure of documentation
	TO29-346 - Install / configure LaTeX
	TO29-349 - literature research
	TO29-350 - prepare citations
	TO29-351 - add citations
	TO29-358 - write paper
	TO29-359 - refactor paper
TO29-341 - Scientific Methods	TO29-353 - develop research question
	TO29-354 - compare quantity / quality / mixed methods approach
	TO29-355 - create questionnaire
	TO29-356 - create checklist for interview
	TO29-357 - scientific research interview
TO29-343 - Marketing Jingle Video	TO29-360 - plan what scenes to video-capture
	TO29-361 - create video material
	TO29-362 - select eye-catching scenes for video from video resources
	TO29-363 - create audio material
	TO29-364 - select music
	TO29-365 - edit video
	TO29-366 - convert video file
TO29-344 - Project Diary	TO29-367 - add sprints
	TO29-368 - add effort estimation
	TO29-369 - update "Delivery"
TO29-352 - Final Presentation Powerpoint	TO29-372 - decide for content
	TO29-373 - edit slides (incl. picture / video resources)
	TO29-374 - edit design
	TO29-375 - final upload .pdf + .pptx

Epic	
User Story	Sub Task
TO29-37 - 2.Detection-Software	
TO29-311 - Detection software - find better working pre-defined solution	TO29-316 - Browse for new/more adequat solution
	TO29-317 - Implementation on Windows system
	TO29-318 - Testing of performance and functionality on Windows system
	TO29-331 - Install flask (python webserver to define routes -> sending camera stream to webpage)
	TO29-371 - create new demo video to test functionality
TO29-312 - Detection software - implement on Raspberry Pi 4	TO29-319 - Reset Raspberry Pi
	TO29-320 - Install all necessary packages / DB
	TO29-321 - Install source code synch application
	TO29-322 - Prepare directory for project
	TO29-323 - Execute and test (live demo) existing project on Raspberry Pi and test functionality
	TO29-324 - Setup autostart
TO29-38 - 3.Hardware	
TO29-314 - 3D print - Raspberry Pi case	TO29-325 - Request 3D print at ProjectKitchen (made appointment)
	TO29-326 - Send data to print
	TO29-327 - Wait for print
	TO29-328 - Pickup printed 3D case
	TO29-329 - Style printed 3D case (colouring)
	TO29-330 - Build Raspberry Pi hardware setup
TO29-40 - Website	
TO29-376 - Website	TO29-386 - research different frameworks
	TO29-377 - Choose Framework
	TO29-387 - setup framework
	TO29-378 - Design Interface
	TO29-388 - make interface mockup
	TO29-379 - Implement skeleton
	TO29-389 - connect database
	TO29-383 - plot statistics
	TO29-392 - implement views
	TO29-393 - admin mode for video

	TO29-394 - implementing functionality
	TO29-395 - testing
	TO29-396 - debugging
TO29-315 - Interface - local access point (hardware)	TO29-337 - Request Supervisor specifications (hardware / location)
	TO29-338 - setup autostart (displaying webpage)
TO29-41 - 5.Database	
TO29-313 - Statistical evaluation - store data in DB	TO29-332 - Request supervisor for database (Grafana + InfluxDB) access
	TO29-333 - Setup datatable columns
	TO29-334 - Implement connection in Python script
	TO29-335 - Synch new implementation to Raspberry Pi
	TO29-336 - Testing

4. Delivery

GitHub Repository to include all documentation, source code and other resources:
https://github.com/unravelet/theObserver.git

5. Our Project Diary

Notizen 07.10.2021

Folge Termine werden so wie das erste Treffen stattfinden (im Break-Out-Room warten)
Deutsch oder Englisch als Entscheidung für die Dokumentation steht frei zu wählen
Self Study C umsetzen bis zum nächsten Mal (kommenden Dienstag)
Presentation Lab B5.xx (Lifte links) => Ausstellungsraum für Projekte
Projekt Kitchen + 3D Drucker möglicher Zugang für das Projekt (=> Gehäuse für Raspberry Pi + Drucker anfertigen [Planung + Ausdruck])
Datenschutz => keine ganzen Bilder direkt über die Webpage abfangen => nur ausgewertete Daten
(2 verschiedene Anzeigen => lokal der Stream [nicht webpage sondern interne Lösung], über Webpage mit öffentlichem Zugang reine Datenauswertung)
=> webpage soll “schön” für Kundenzugang hergerichtet werden
Dokumentation kann von IT-Projekt aus Semester #2 verwendet werden (timetable + project schedule etc)
bei Sprint Zusammenfassung ähnlich wie Semester #2 + Fehleranalyse (grob) in schriftlicher Form
Montag, 11.10.2021, 14:30 5ter Stock A Gebäude rechts nach den Liften, Durchgang zum kleinen Meeting Raum (nicht Studiengangsassistenz) => Hardware Ausgabe (1 Set für Gruppe)
Exact Solution => Zeitabschätzung wie gewohnt
Namens-Gebung überlegen (Github Rep, Webpage => name.w3.cs.technikum-wien.at)
Kamera soll am Eingang positioniert werden (Vektoren verfolgen für +/-) oder permanent den ganzen Raum filmen (auf Eingang oder in den Raum ausgerichtet?)
Links:
Search Thingiverse - Thingiverse
zur Verwendung in ProjectKitchen: PrusaSlicer (Aufteilung von Masse und Materialnutzung) => .gcode (File Extension) => auf die SD Karte (beim Drucker vorhanden) spielen und für den 3D Drucker verwenden
office@htw.wien + project kitchen adresse wegen Schritte für 3D Druck ansprechen (Hochschulvertretung FH) => office@htw.wien ins CC setzen, project kitchen priorisieren

Notizen 08.10.2021

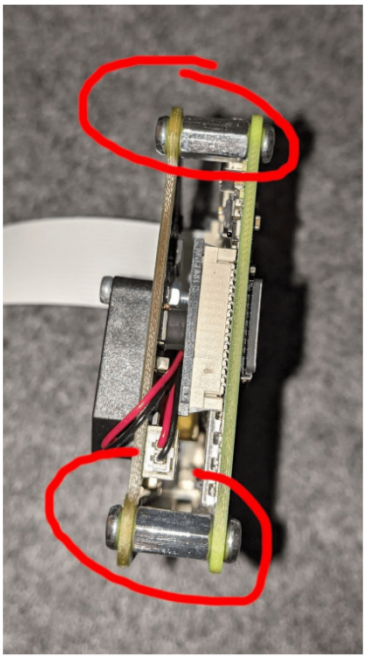
- Entscheidung für Positionierung => sollte in den Raum blicken für ständige Zählung durch geplanten 2ten Eingang
- Management-Survey Text ausbessern für die Positionierung der Kamera
- Donnerstag, 14.10.2021, B5.xx Presentation Lab Begehung 10:30/35
- Ende Semester 4 muss Live Demo + Video (5 min) gemacht werden mit Aussicht über noch ausstehende Aufgaben
- Ende Semester 5 Werbevideo mit voll funktionsfähigem Produkt anbieten
- Grafana / InfluxDB / noteRed => zur Verwendung für Webpage (googlen)

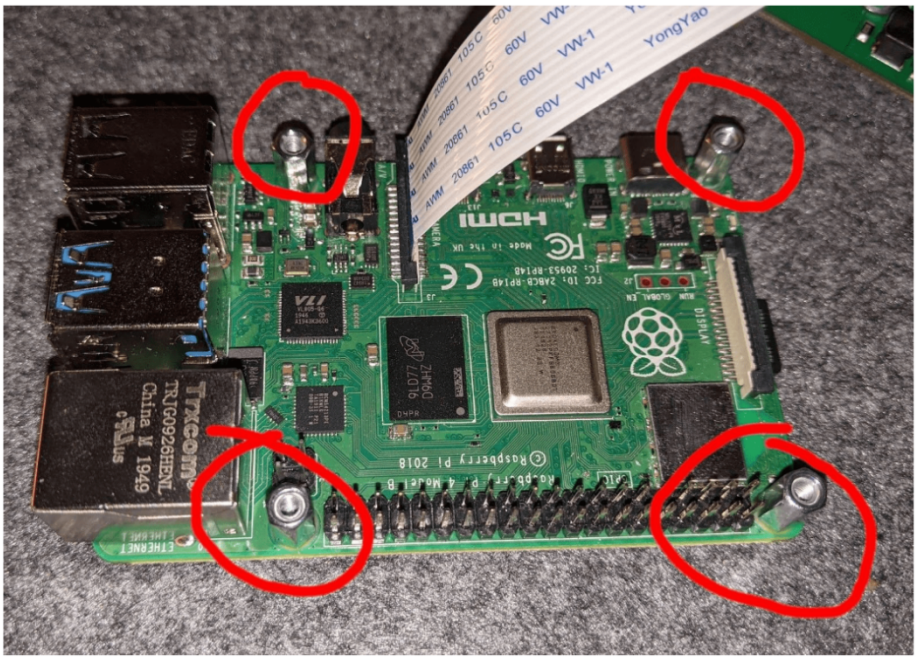
Notizen 14.10.2021

- Begehung von Presentation Lab (Raum für die Durchführung des Visitor Counter)
- *es wurde bestätigt, dass wir als Entwicklungsteam KEINE Zugangsrechte zum Raum*
- *bekommen => nur durch das Beisein von Supervisor möglich*
- Maße des Raums genommen (9x20 Meter)

Notizen 03.11.2021

- gemeinsames Meeting (Hardware wurde kontrolliert auf Funktionalität + Beschaffenheit)
- erstmalige Nutzung von Raspberry Pi 4 durch Stromkabel ergab volle Funktionsfähigkeit
- erstmalige Nutzung durch PoE (Power over Ethernet) musste abgebrochen werden aufgrund von Brandgefahr => genauere Verwendung wird in Dokumentation nachgelesen und bei erneuter Brandgefahr wird Supervisor wegen fehlerhafter Funktionalität kontaktiert
- restliche Aufgaben von Sprint #1 wurden verteilt (siehe Abschnitt Sprint#1)
- erste Ausgabe des Livestreams des Pis über HDMI mittels den Befehl raspivid -t 10000





weitere Erkenntnisse im Laufe des Abends (Dominic):

- nach ausführlicher Recherche und genauer Installation (mittels beigefügten Spacers, siehe Foto) funktioniert der PoE Hat weiterhin nicht, vermutlich war dieser bereits bei Auslieferung defekt

- weitere Überlegung bevor die Semester Roadmap adaptiert wird und die Trello Tasks gepflegt werden:

Sollten wir, da sich dieses Projekt über mehrere Semester streckt, eine Art Trello Guideline erstellen, wie die Tasks bei der Erstellung auszusehen hat sowie Handhabung der Tasks?

Erstellung der Tasks

z.B. Tags wie “Sprint 1.2”, erstes 1 für erstes Projektsemester, 2 für den zweiten Sprint des Semesters, dadurch könnten wir dynamische Listen erstellen, Task wie “Bug”, “Feature” etc.

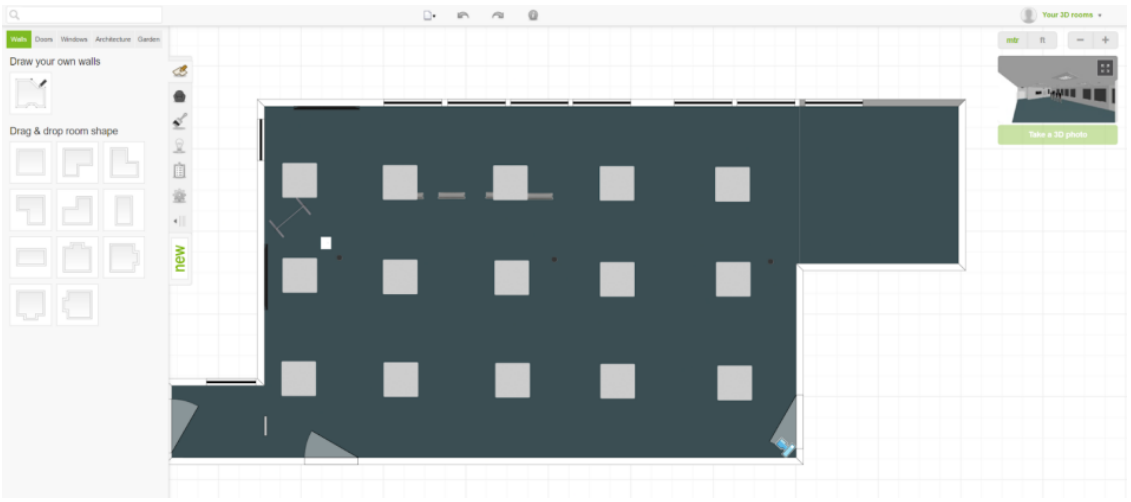
Handhabung der Taks

z.B. Gegencheck eines anderen Projektmitglieds bevor der Task unter Done abgelegt wird

- zusätzlich sollten wir das Case für den Raspberry Pi (Achtung auf die Kompatibilität des Kamera Moduls sowie dem PoE hat) bereits in den ersten zwei Sprints mit einplanen, da Transport sowie Lagerung (Staub) schädlich für den Raspberry Pi sein könnte

Notizen 09.11.2021

Erstellung eines 3D Modells vom “PresentationLab” zur späteren Orientierung und weiteren Handhabung.



Notizen 10.11.2021

Namensvorschläge (Domain / GitHub Repository)

GitHub	Domain
visitorCounter	www.visitorCounter.at
studentStalker	www.studentStalker.at
mysticEye	www.mysticEye.at
theObserver	**theObserver.w3.cs.technikum-wien.at**
secretEye	www.secretEye.at
studentCounter	www.studentCounter.at
TW-studentCounter	www.TW-studentCounter.at

Notizen 24.11.2021

Check filaments for 3D print:

- ABS - **Acrylnitril-Butadien-Styrol** (2nd most popular filament; Terpolymer => Polymer consisting of 3 different Monomers; popular in the 1950's; easy handling; tough & heat resistent)
- ASA - **Acryl-Styrol-Acrylnitrit** (high quality, but less usable & tough in comparison to ABS)
- PLA - **Polylactid Acid** (renewable resources (corn), biodegradable, easy to use for printing, might melt by heat)
- PETG - **Polyethylenterephthalat** (recyclable, not biodegradable (contains oil), heat resistent)

Notizen 25.11.2021

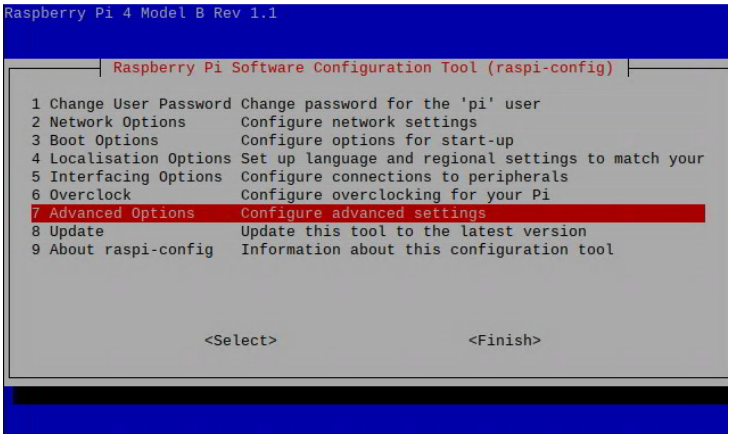
Meeting with Supervisor:
https://towardsdatascience.com/how-to-detect-objects-in-real-time-using-opencv-and-python-c1ba0c2c69c0
(link for further information of using recognizing software)

Infill
75-80% would be optimal for a resistent use of raspberry pi case
Filament - ABS
we are allowed to do the printing by our own

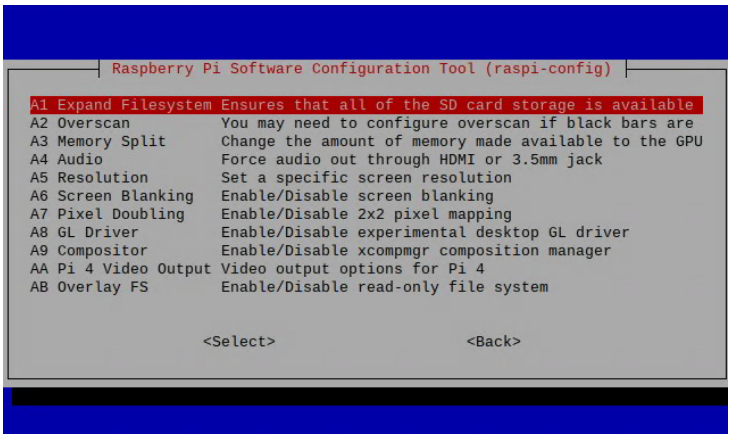
Notizen 06.12.2021

Configure OpenCV on Raspberry Pi

First make sure your Pi is using the full SD card
$sudo raspi-config
-choose “**Advanced Options**”

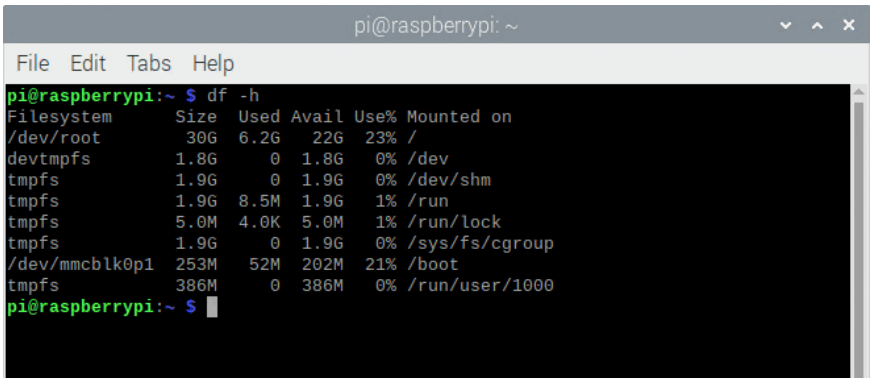


-choose **"Expand Filesystem"**



Reboot the machine
$sudo reboot

Check space
$df -h



Update the system
$sudo apt-get update && sudo apt-get upgrade

Get Pip and **Install OpenCV**
$mkdir ~/src && cd ~/src
$wget https://bootstrap.pypa.io/get-pip.py
$ sudo python3 get-pip.py

There are two ways to install OpenCV. It is possible to install it for the whole system or use a Python Virtual Environment (which is more recommended)

1. Install OpenCV for the Whole System

#$pip install --upgrade pip setuptools wheel
sudo pip install opencv-contrib-python

2. Install OpenCV in a Python Virtual Environment

Install the **venv** module for Python virtual environments - this need to be done only once for all future projects
sudo apt-get install python3-venv

Create a new Python project in a virtual environment
cd ~ python3 -m venv test

This will create a new folder called **test** that you should change to
cd ~/test

List the folder contents - you should see a file called **pyvenv.cfg**
cat pyvenv.cfg

You should see a result like this (though your version my vary):

test= /usr/bin
include-system-site-packages = false
version = 3.7.3

*The **include-system-site-packages = false** line indicates that this program will not use any external packages installed elswhere on your system. The idea is to create a program in an isolated environment. That way it can be run on other systems without having to worry about external dependencies.*

Switch to the **test** folder and activate the virtual enviromnent
cd ~/hello source bin/activate

Your prompt should change to something like this (your user and hostname may be different)
(test) pi@pi4:~/test $

Install OpenCV
pip install --upgrade pip setuptools wheel
pip install opencv-python

Create a new Python file called main.py
touch main.py

Edit the file using the nano editor:
nano main.py

Paste some text there and save the file
import sys
import cv2
print("Hello World")

From within the virtual environment, run this
python main.py

You should get a response like this:
Hello World

You know everything worked fine if you dont have an **error message** that cv2 coudnt be found or something else related to import cv2.

In order to deactivate the virtual environment and return to the main operating system command line
deactivate

Remove a Python virtual environment
cd ~
rm -rf test

Sources
https://www.jeremymorgan.com/tutorials/raspberry-pi/how-to-install-opencv-raspberry-pi/

https://desertbot.io/blog/raspberry-pi-headless-python-venv-setup

Notizen 08.12.2021

discussion about current topics:

- fill in current sprint review protocoll (review#3)
  - don't forget meta data
  - are there tasks not done in the current sprint?

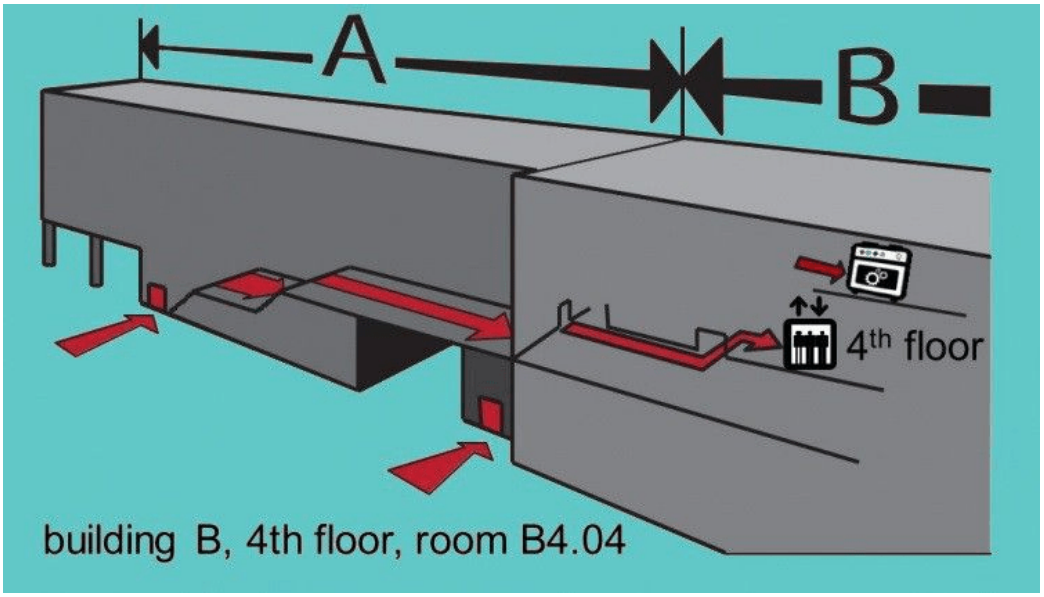
- requirements next sprint?
  - semester roadmap says "implement system for visitor counter" what is exactly meant?
  - common administration tasks
  - communication with ProjectKitchen

- next steps regarding 3D print
  - discuss with team if print will be done together in January or contact person does without any of us by himself in the next days due to construction inside ProjectKitchen

- next usage of opencv (regarding videos)
  - https://www.tensorscience.com/object-recognition/person-detection-in-video-streams-using-python-opencv-and-deep-learning Person detection in video streams using Python, OpenCV and deep learning
  - https://data-flair.training/blogs/python-project-real-time-human-detection-counting/
  - https://www.youtube.com/watch?v=oXlwWbU8l2o OpenCV Course - Full Tutorial with Python
  - https://www.computervision.zone/projects/ (Tipp Supervisor)

Notizen 10.12.2021

Where to find the ProjectKitchen



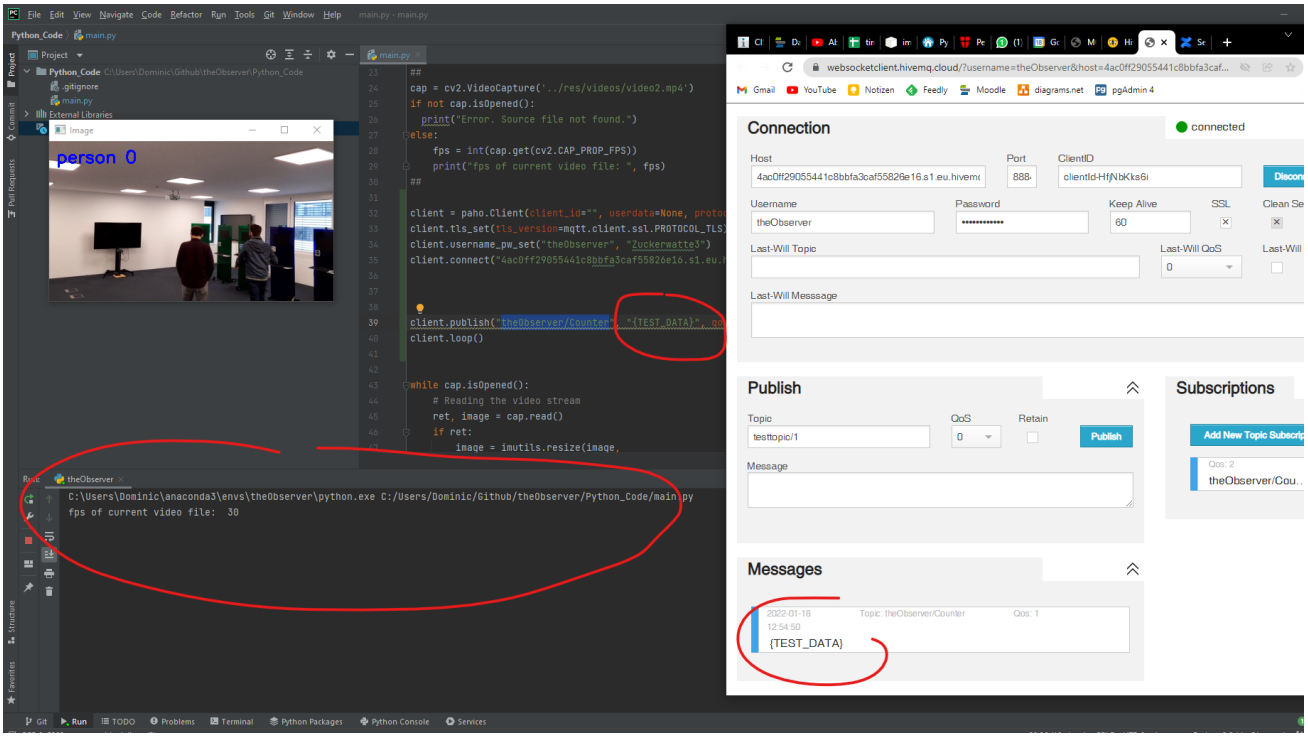
Notizen 18.01.2022

mqtt send and recieve counter number

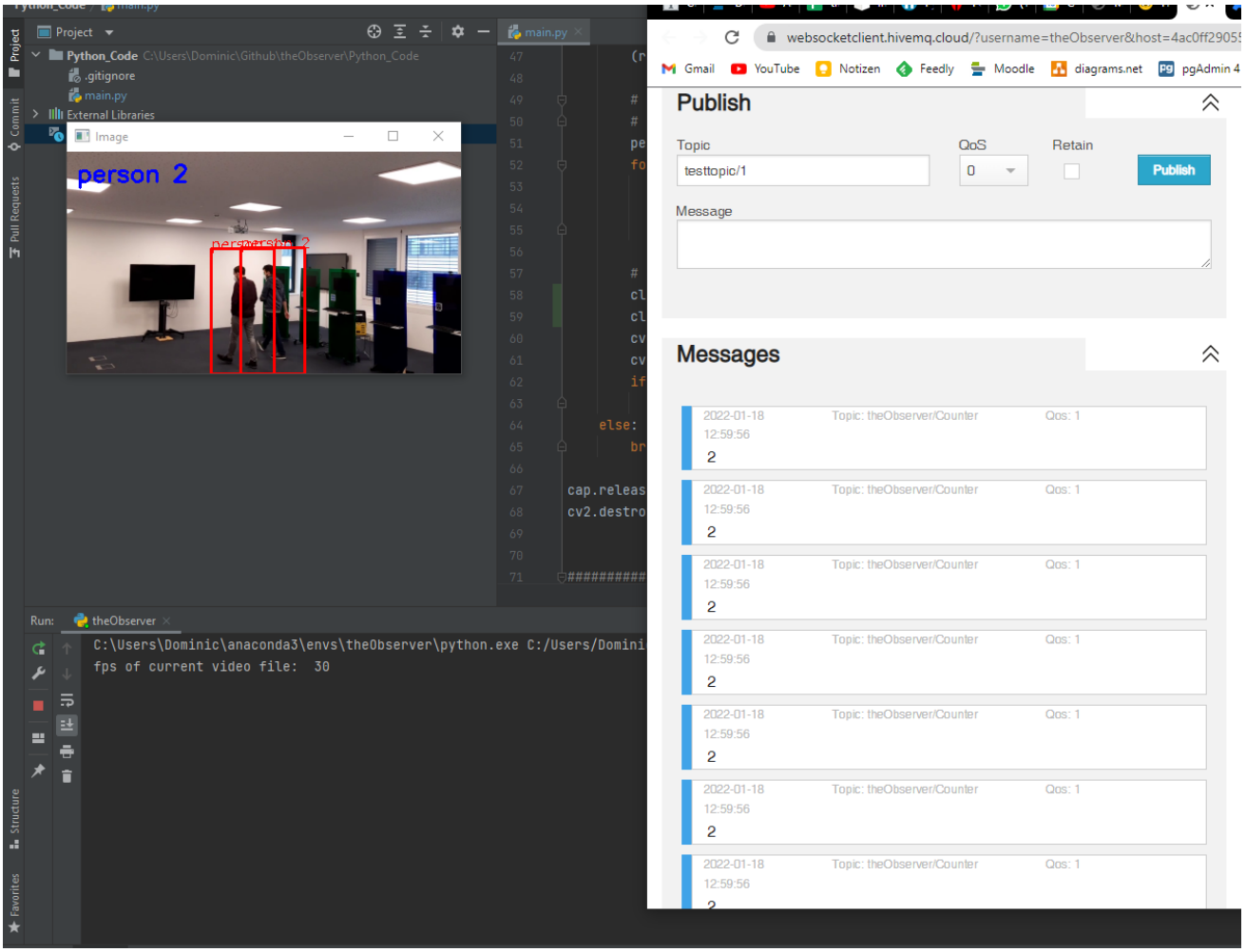
install dependencies for our python project

```
python.exe -m pip install opencv-python imutils paho-mqtt
```

then extend code with mqtt functionality and connect with another client and subscribe the topic and see the messages
Password for the Test Topic on HiveMQ is Zuckerwatte3
Client to test Topic:
https://websocketclient.hivemq.cloud/?username=theObserver&host=4ac0ff29055441c8bbfa3caf55826e16.s1.eu.hivemq.cloud&port=8884

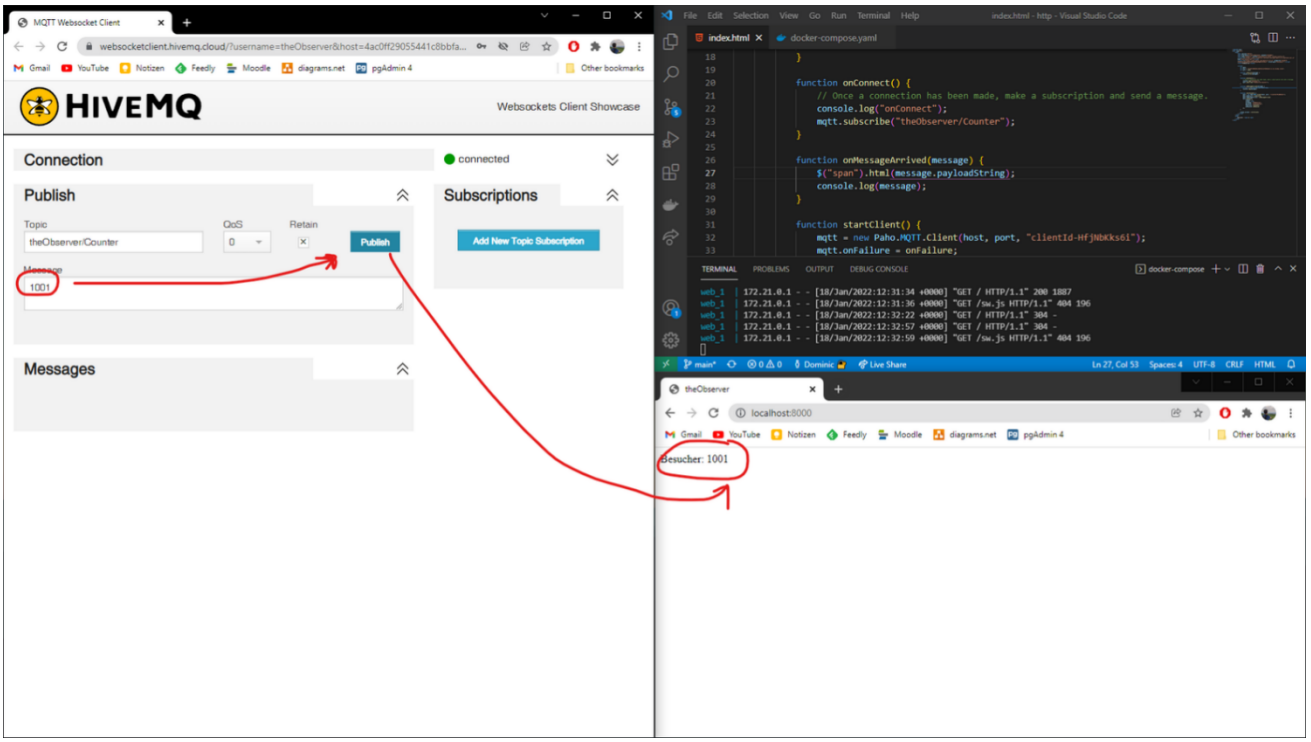


Source for python code:
https://console.hivemq.cloud/clients/python-paho?uuid=4ac0ff29055441c8bbfa3caf55826e16



With current code, the counter is send to the topic every frame, but we have to manage the time, otherwise the broker server of mqtt will overflow… maybe just when the number has changed

There is also a mqtt Library for Javascript, see here
https://console.hivemq.cloud/clients/mqtt-js?uuid=4ac0ff29055441c8bbfa3caf55826e16
So we don't really need PHP for our website
Maybe we should consider to set up our own Broker Server on the Raspberry Pi



Success test script for html/javascript in git repo in http folder is saved

Notizen 20.01.2022

für mqtt folgenden Link nutzen (weitere Informationen researchen):

https://mosquitto.org/

- öffentlichen Broker nutzen (da auch die Webpage öffentlich erreichbar werden soll)

Publisher und Subscriber laufen lassen => achte auf unterschiedliche UID

zuerst mit Broker über URL verbinden (client.connect(url, port, xy))

Notizen 25.03.2022

Notes of research regarding people detection/recognition of different scopes:
[explanation of detection vs tracking]

https://pyimagesearch.com/2018/08/13/opencv-people-counter/

[detect upper part of body and face]

https://www.youtube.com/watch?v=x70ALxNzKto

[detecting by head]

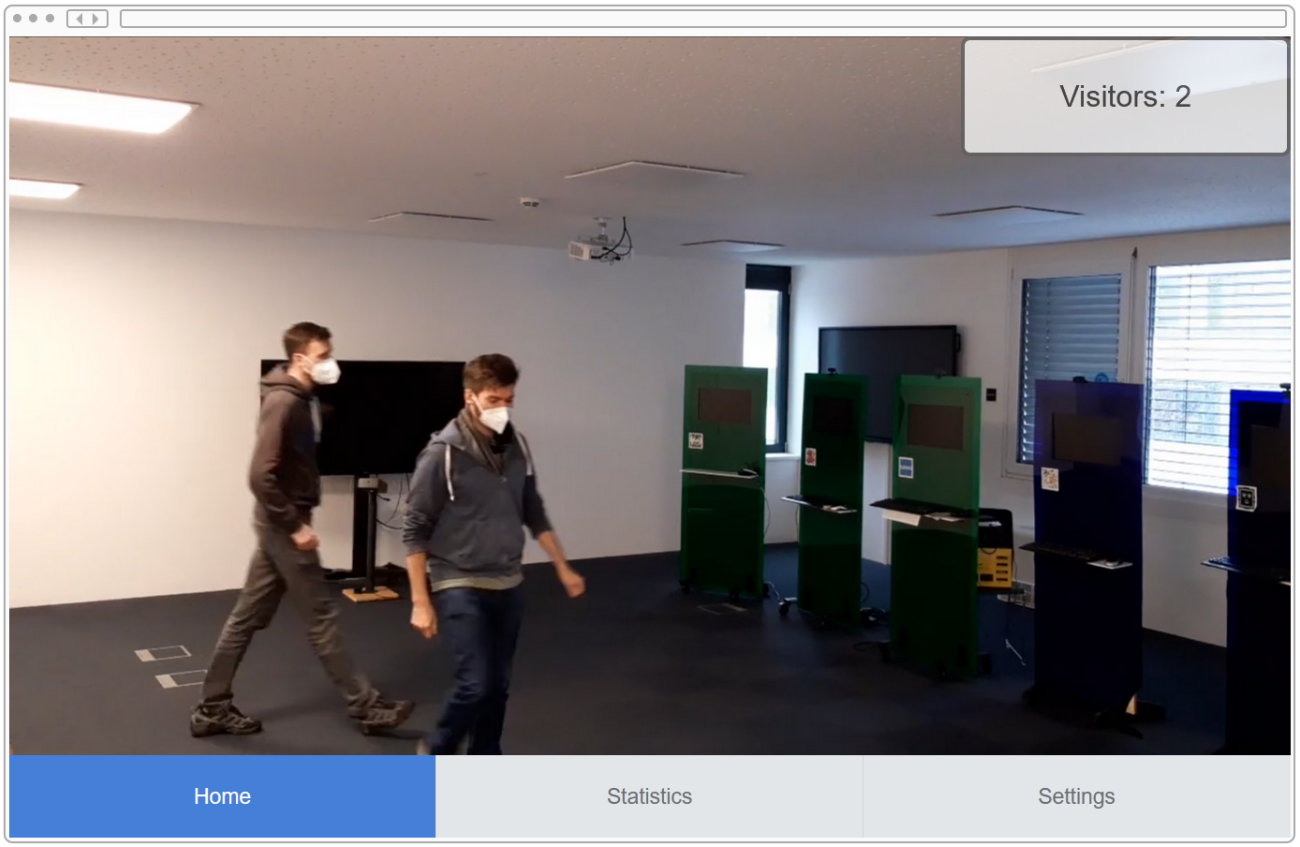
https://www.youtube.com/watch?v=d1bky80NXeQ

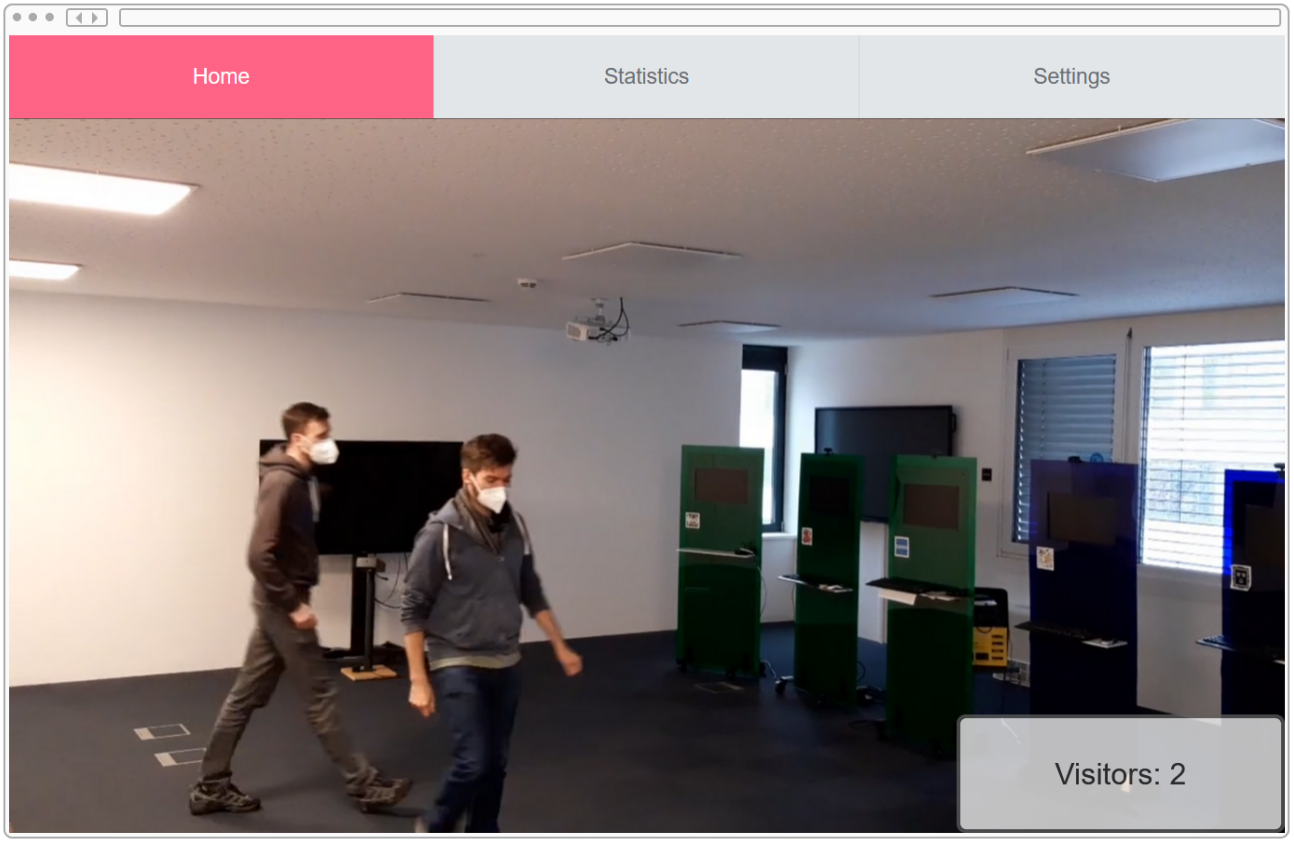
[count head]

https://mateoatwi.com/headcount/

Notizen 29.03.2022

theObserver UI Mockups



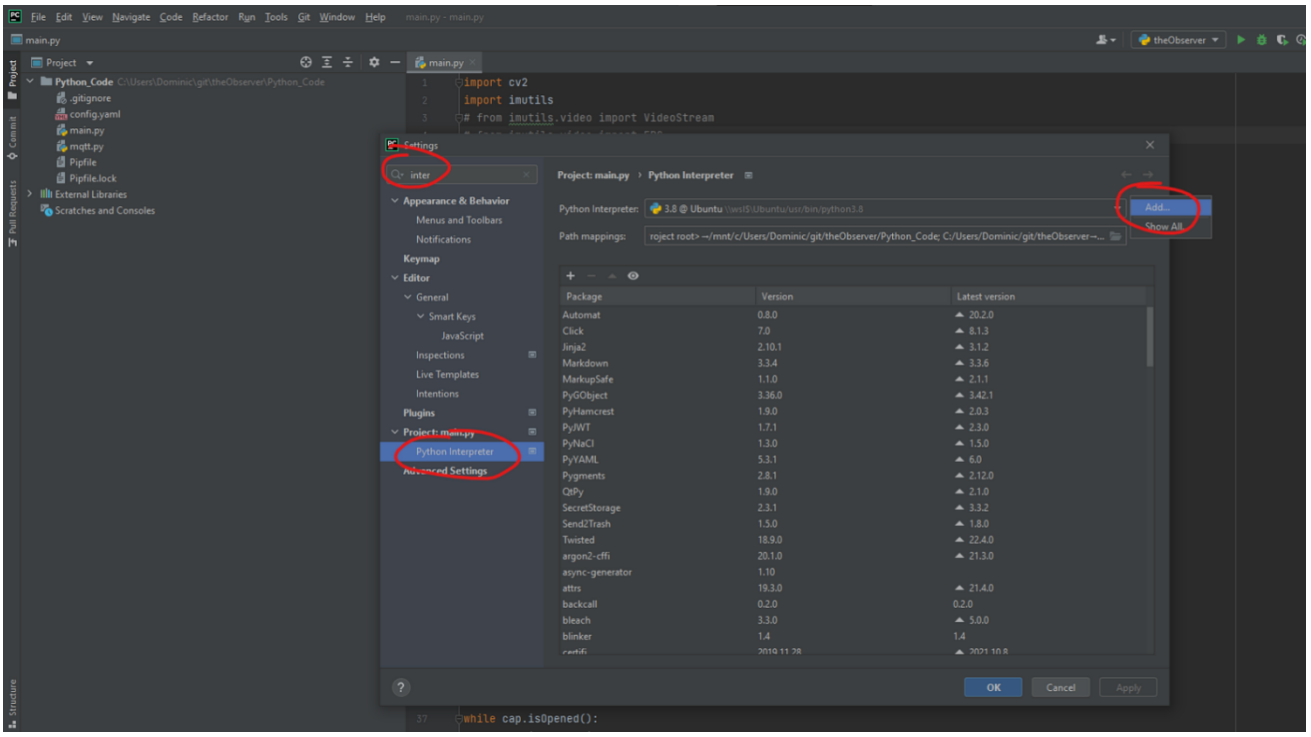


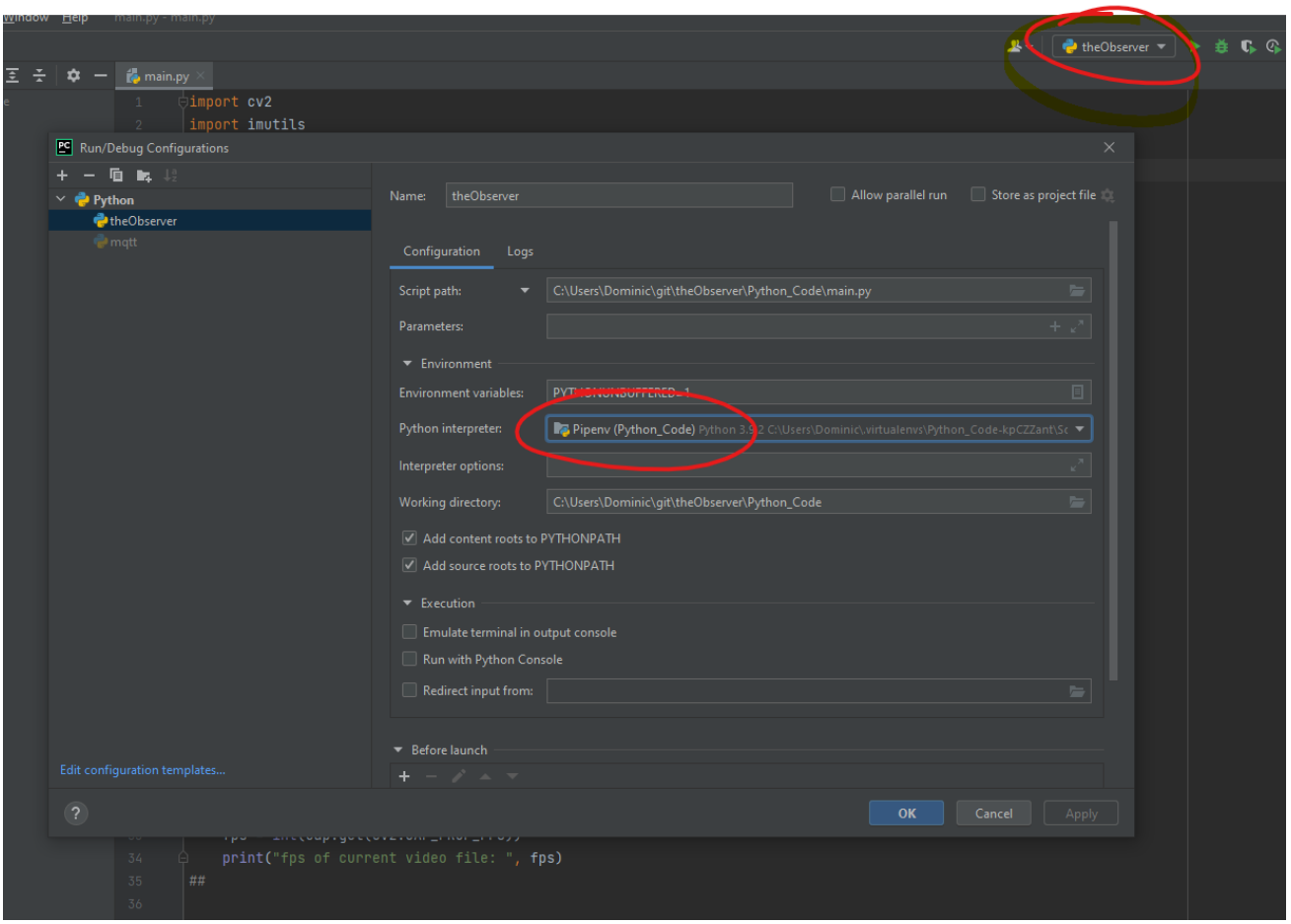
Notizen 05.05.2022

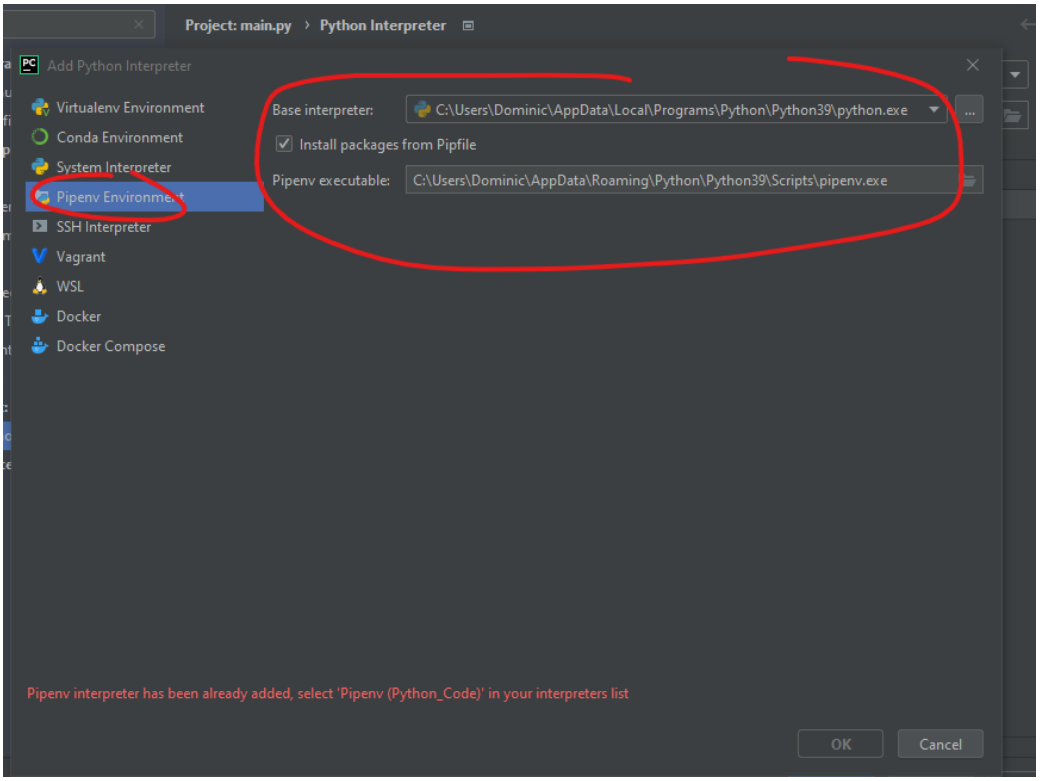
I removed the idea folder in the Python_Code folder, because it contains information about the interpreter, which is different on every our computers… so you have to copy it first, before you pull the newest changes, otherwise it will be deleted

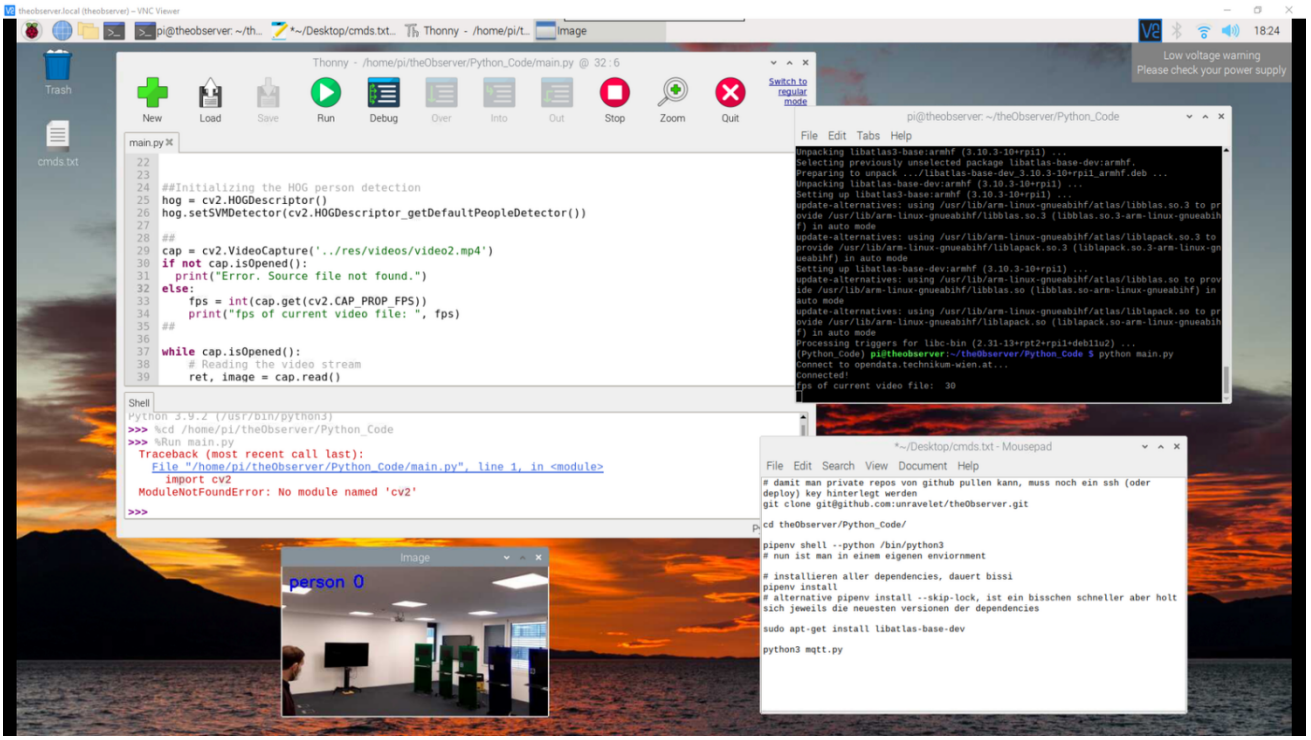
also I strongly recommend to use pipenv instead of normal pip and normal python usage

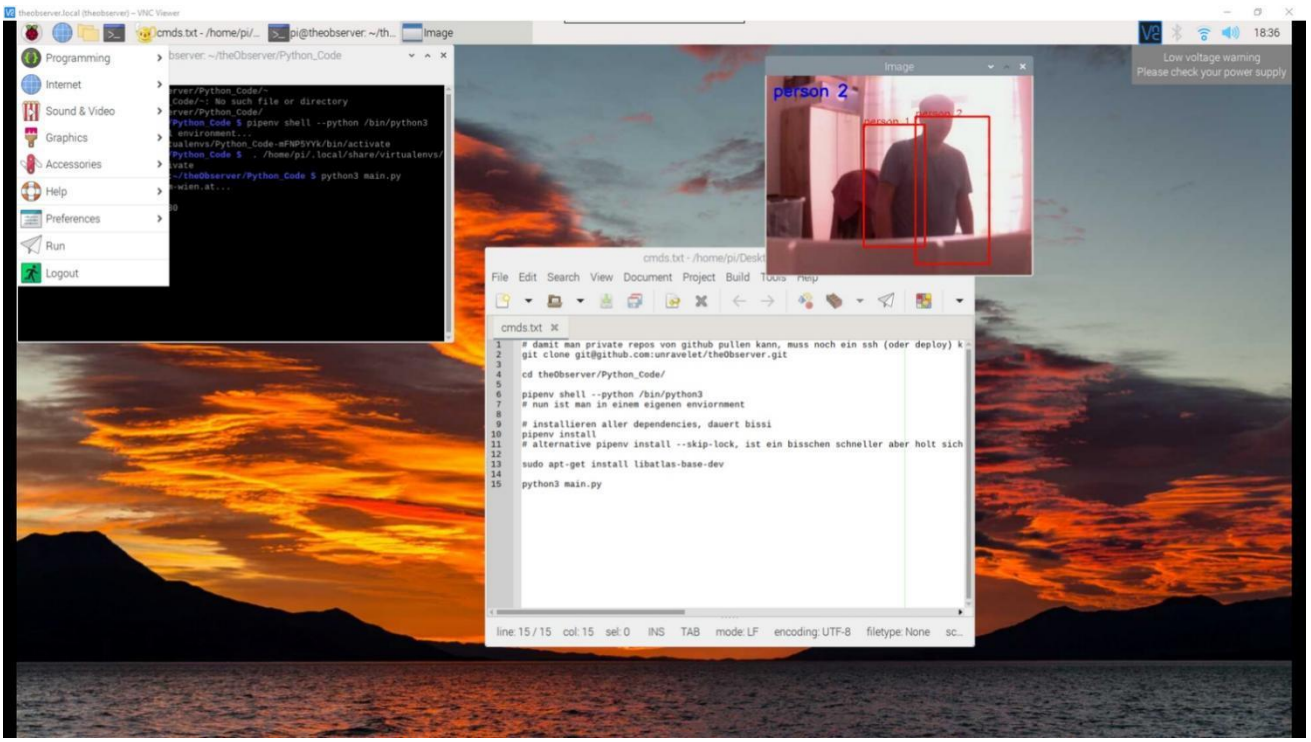
its better for hanelding dependencies in a collaboration environment like ours (greate guid: https://realpython.com/pipenv-guide/)




theObserver
main.py
import cv2
import imutils
Run/Debug Configurations
Python
theObserver
mqtt
Name: theObserver
Allow parallel run
Store as project file
Configuration
Logs
Script path: C:\Users\Dominic\git\theObserver\Python_Code\main.py
Parameters:
Environment
Environment variables:
Python interpreter: Pipenv (Python_Code) Python 3.8
Interpreter options:
Working directory: C:\Users\Dominic\git\theObserver\Python_Code
Add content roots to PYTHONPATH
Add source roots to PYTHONPATH
Execution
Emulate terminal in output console
Run with Python Console
Redirect input from:
Before launch
Edit configuration templates...
OK
Cancel
Apply
print("fps of current video file: ", fps)
##

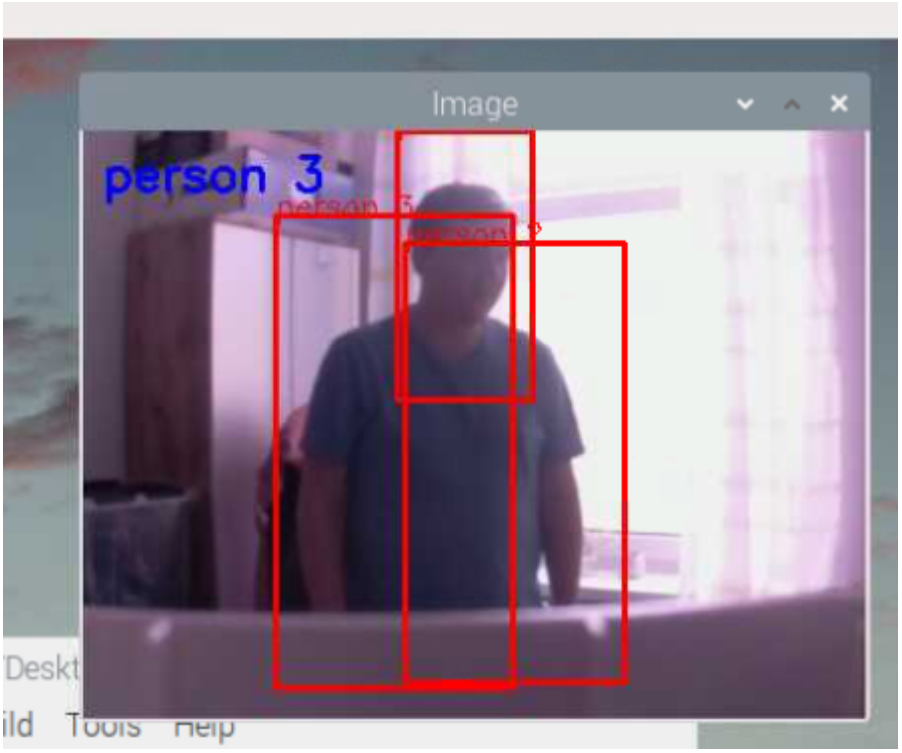

Project: main.py › Python Interpreter
Add Python Interpreter
Virtualenv Environment
Conda Environment
System Interpreter
Pipenv Environment
SSH Interpreter
Vagrant
WSL
Docker
Docker Compose
Base interpreter: C:\Users\Dominic\AppData\Local\Programs\Python\Python39\python.exe
Install packages from Pipfile
Pipenv executable: C:\Users\Dominic\AppData\Roaming\Python\Python39\Scripts\pipenv.exe
Pipenv interpreter has been already added, select 'Pipenv (Python_Code)' in your interpreters list
OK
Cancel






```
# damit man private repos von github clonen kann, muss man einen ssh key (oder noch besser, deploy key) hinterlegen

git clone git@github.com:unravelet/theObserver.git

cd theObserver/Python_Code/

pipenv shell --python /bin/python3
# nun ist man in einem eigenen enviornment

# installieren aller dependencies, dauert bissi
pipenv install
# alternativ pipenv install --skip-lock, ist ein bisschen schneller, holt aber sich jeweils die neuesten versionen der packages
```

```
sudo apt-get install libatlas-base-dev

python3 main.py

python3 mqtt.py
```



seems like our program thinks I’m big as 3 people

Notizen 11.05.2022

https://www.dexterindustries.com/howto/run-a-program-on-your-raspberry-pi-at-startup/#local

https://stackoverflow.com/questions/36466500/on-raspberry-pi-auto-start-terminal-after-login

https://forum-raspberrypi.de/forum/thread/3345-autostart-eines-python-script/

https://www.digikey.at/en/maker/blogs/2019/running-python-programs-on-startup-using-the-raspberry-pi

6. Effort estimation

Phase 2 (Semester #4)

Delphi Verfahren mit PERT - Unsicherheit ausdrückt

ID	Themes / Areas / Arbeitspakete Toplevel	Epics / User Stories / Arbeitspakete TopLevel	User Stories / Detail Level / Beschreibung	Optimistisch (Sp)	Wahrscheinlich (Sp)	Pessimistisch (Sp)
				8,0	12,0	18,0
	Mockup	Team develops mockup to visualize interface which will be in use at PresentationLab and live demonstration				
			choose collaboration tool to create mockups	1	1	2
			create 2 mockups to display user interface	2	2	3
	3D print	Finally supporting equipment for raspberry pi will be created via 3D print at ProjectKitchen				
			request date and time to visit ProjectKitchen for 3D print	1	1	2
	Software final development	Software gets final design and bugfixes				
			browses alternative recognition software for handling out/in scope	1	3	5
			discussion with supervisor about specified requirement of daily/current counter	1	1	1
			adapt software to recognize people on whole body or bodyparts only	2	4	5
	Connect Hardware + Software	Raspberry Pi gets Software installed and will be connected to camera modul and supporting tech gear				
	Interface PresentationLab	Presentation Lab gets an own interface to use Visitor-Counter at location				
	Live Demonstration	Raspberry Pi functionality will be demonstrated installed at Presentation Lab				

eite, p(range)

Erwartet nach PERT 1:4:1 (Sp)	Kalibrierter erwarteter Aufwand (Ph)	p(range) (5-100)	Divisor (0,25-6)	Standard-Abweichung (Ph)	Varianz	Aufwand (T-Shirt: XS-XXL)
12,33	12,33			0,882	0,778	
		99,73	6,00			
1,17	**1,17**	99,73	6,00	0,167	0,028	S
2,17	**2,17**	99,73	6,00	0,167	0,028	M
		99,73	6,00			
1,17	**1,17**	99,73	6,00	0,167	0,028	S
		99,73	6,00			
3,00	**3,00**	99,73	6,00	0,667	0,444	M
1,00	**1,00**	99,73	6,00	0,000	0,000	S
3,83	**3,83**	99,73	6,00	0,500	0,250	M
		99,73	6,00			

https://github.com/unravelet/theObserver/blob/main/doc/Effort_Estimation/DAGoPERT_2016_slim.xlsx [link to GitHub repository => excel sheet of effort estimation DAGoPERT]

Phase 3 (Semester #5) - Story Points measured by 1SP == 2h effort

A	B	C	D	E	F	G	
ID	Themes / Areas / Arbeitspakete Toplevel	Epics / User Stories / Arbeitspakete TopLevel	User Stories / Detail Level / Beschreibung	Optimistisch (Sp)	Wahrscheinlich (Sp)	Pessimistisch (Sp)	Doku Überlegun Diskus
				81,0	114,0	182,0	
	2. Detection-Software	Detection software - find better working pre-defined solution					
			Browse for new/more adequat solution	3,0	5,0	8,0	
			Implementation on Windows system	2,0	3,0	5,0	
			Testing of performance and functionality on Windows system	1,0	2,0	3,0	
			Install flask (python webserver to define routes -> sending camera stream to webpage)	1,0	2,0	3,0	
			create new demo video to test functionality	1,0	1,0	2,0	
		Detection software - implement on Raspberry Pi 4					
			Reset Raspberry Pi	1,0	1,0	2,0	
			Install all necessary packages / DB	2,0	3,0	5,0	
			Install source code synch application	1,0	2,0	3,0	
			Prepare directory for project	1,0	1,0	2,0	
			Execute and test (live demo) existing project on Raspberry Pi and test functionality	2,0	2,0	3,0	
			Setup autostart	1,0	2,0	3,0	
	5. Database	Statistical evaluation - store data in DB					
			Request supervisor for database (Grafana + InfluxDB) access + introduction	2,0	2,0	3,0	
			Setup datatable columns	2,0	3,0	5,0	
			Implement connection in Python script	1,0	2,0	3,0	
			Synch new implementation to Raspberry Pi	1,0	1,0	2,0	
			Testing	1,0	3,0	5,0	
	3. Hardware	3D print - Raspberry Pi case					
			Request 3D print at ProjectKitchen (made appointment)	1,0	1,0	2,0	
			Send data to print	1,0	1,0	2,0	
			Wait for print	1,0	1,0	1,0	
			Pickup printed 3D case	1,0	1,0	2,0	
			Style printed 3D case (coloring)	2,0	2,0	3,0	
			Build Raspberry Pi hardware setup	2,0	2,0	5,0	
	4. Website	Interface - local access point (hardware)					
			Request Supervisor specifications (hardware / location)	1,0	1,0	2,0	
			setup autostart (displaying webpage)	1,0	2,0	3,0	
	1. General Administrative	Paper					
			Identify structure of documentation	1,0	2,0	3,0	
			Install / configure LaTeX	1,0	2,0	3,0	
			literature research	1,0	2,0	3,0	
			prepare citations	1,0	2,0	3,0	
			add citations	2,0	3,0	3,0	
			write paper	2,0	3,0	5,0	
			refactor paper	1,0	2,0	3,0	
		Scientific Methods					
			develop research question	1,0	1,0	2,0	
			compare quantity / quality / mixed methods approach	1,0	1,0	2,0	
			create questionnaire	1,0	2,0	3,0	
			create checklist for interview	1,0	1,0	2,0	
			scientific research interview	1,0	2,0	3,0	
		Marketing Jingle Video					
			plan what scenes to video-capture	1,0	1,0	2,0	
			create video material	1,0	2,0	3,0	
			select eye-catching scenes for video from video resources	1,0	2,0	2,0	
			create audio material	1,0	2,0	3,0	
			select music	1,0	1,0	2,0	
			edit video	2,0	3,0	5,0	
			convert video file	1,0	1,0	2,0	
		Project Diary					
			add mandatory resources	1,0	2,0	2,0	
		Final Presentation Powerpoint					
			create powerpoint similar to last semester	1,0	2,0	2,0	
		Source Code					
			clean / refactor repository	1,0	1,0	2,0	
			create new branch	1,0	1,0	2,0	
	4. Website	Framework					
			research frameworks	1,0	1,0	2,0	
			choose framework	1,0	1,0	1,0	
			setup framework	1,0	2,0	3,0	
		UI					
			Design UI	1,0	1,0	2,0	
			UI mockup	1,0	1,0	2,0	
		coding					
			implement skeleton	2,0	2,0	3,00	
			connect database	2,0	3,0	5,00	
			plot statistics	2,0	3,0	5,00	
			implement views	3,0	3,0	5,00	
			admin mode for video	1,0	2,0	2,00	
			implement functionalities	3,0	3,0	5,00	
			testing	2,0	2,0	3,00	
			debugging	2,0	3,0	5,00	

https://github.com/unravelet/theObserver/blob/main/doc/Effort_Estimation/DAGoPERT_Semester_5.xlsx