Teachers' To Do

- <do Jan>Add students to Keycloak (Jan) including group and role, e.g. WS2021_IP17</do>
- <do>Setup a editable Wiki branch for student documentation</do>
- <do Jan>Setup Nextcloud Docker for files Sharing</do>
- <do>Setup Discord Server including categories (IP ...) and channels (General Discussion? Announcements?) → Responsive comm.</do>
- <do>(Mattermost?)</do>
- <do> (in process) Missing orders! Powerback (350€ incl. tax), Drivers (291€ incl. tax)</do>

Schedule

Date	Location	Topics	Session Outcome	Assignments
30.09.2021	IoT Lab (02 02 510)	Introduction		
07.10.2021	IoT Lab (02 02 510)	Assembly of Jetsons		
14.10.2021	IoT Lab (02 02 510)	Playing with object detection		
21.10.2021				
28.10.2021				
04.11.2021				
11.11.2021				
18.11.2021				
25.11.2021				
02.12.2021				
09.12.2021				
16.12.2021				
~~23.12.2021~~				
~~30.12.2021~~				
06.01.2022				
13.01.2022				
20.01.2022				

Work Packages

Software AI

- Implement Object detección with DetectNet
- Create a wrapper to communicate the neural network with Snap!

Requirements:

- Modular Design

- Good coding practices
- Use standard communication protocols

Software Snap

- Interfacing Snap (Front-end) with Jetson Nano (back-end)
  - WebSocket API ?
  - Web Serial API ?
  - Web USB API ?
  - Custom module inside Snap! ?

Requirements:

- Modular Design
- Good coding practices
- Use standard communication protocols

Hardware Jetbots

- Assemble prototype
- Create assembly instructions
- Electronics wiring diagrams

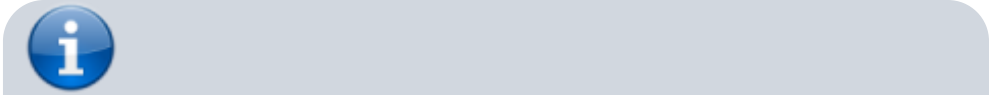
Requirements:

- Easy assembly
- Assembly using common tools
- No specialized machinery
- Modular design with the possibility to remove/place the Jetson Nano easily

Deliverables

Main goal: Develop, design and implement teaching materials on artificial intelligence (AI) for schools (primary schools, age 9-11 years).

- SNAP! course for kids
- Jetbot assembly tutorial
- Replicable projects using SNAP! as frontend:
  - Image classification
  - Predefined object detection with camera: (or rtp, rtsp, rtmp) with ssd-mobilenet
  - Control the Jetbot robot
  - Training convolutional neural network (CNN) for detection of new objects





The complexity of all projects should be hidden, so all projects should be easy to implement using SNAP! as the interface.

Deliverables requirements

- **Learning path (easy to follow)**: The teaching material should be easy to follow, with a defined structure that connects each of the topics in a way that is easy to absorb. The material can even have a story telling that proposes solutions to current problems such as global warming, garbage in the oceans, deforestation and more.
- **Content**: The content of the material may be supported by different multimedia means such as videos, images, audio, etc. The way of presenting the content must be precise and easy to understand.
- **Interactive**: The content must be interactive, with gamification strategies that encourage curiosity to continue learning.
- **Learning experience**: The learning experience must be supported by proposed mini-projects, challenges, examples and a final project that unifies all that has been learned.
- **Appropriate use of licenses**: The teaching material must be open and accessible to everybody under the license CC Attribution-Noncommercial-Share Alike 4.0 International so it is important to make use of resources (videos, audios, images and others) that have a license that allows copying, redistribution, modification and transformation, as far as possible *you should generate your own content under the same license of the course*.

Steps

- OER prototyping on Wiki
- Structure, Material, Content
- Later implementation on Moodle, openEdX, etc.
- Workshop with NVIDIA material for Jetson (understanding complexity!):
  - Dustin Franklin's material first (detectnet, IC, OD, IS)
  - Dana Sheahen's material (Jupyter, image classification)
  - Jim Benson's material (Jetbot, Linux, setup of Jetsons)
- Learn SNAP!
- Concept to communicate between detectnet and SNAP!
- Teaching material

From:
https://wiki.eolab.de/ - **HSRW EOLab Wiki**

Permanent link:
https://wiki.eolab.de/doku.php?id=ip:ws2021:lets_plaiy:todo_t:start&rev=1634216552

Last update: **2021/10/14 15:02**