|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ITU Logo | INTERNATIONAL TELECOMMUNICATION UNION  **TELECOMMUNICATION STANDARDIZATION SECTOR**  STUDY PERIOD 2017-2020 | | FG-AI4H-I-037 | |
| **ITU-T Focus Group on AI for Health** | |
| **Original: English** | |
| **WG(s):** | | DAISAM, DASH | E-meeting, 7-8 May 2020 | |
| **DOCUMENT** | | | | |
| **Source:** | | WG Chairs | | |
| **Title:** | | FG-AI4H assessment platform | | |
| **Purpose:** | | Discussion | | |
| **Contact:** | | Steffen Vogler Bayer AG Germany | | Email: steffen.vogler@bayer.com |
| **Contact:** | | Marc Lecoultre mllab.ai Switzerland | | Email: ml@mllab.ai |
| **Contact:** | | Luis Oala Fraunhofer HHI Germany | | Email: luis.oala@hhi.fraunhofer.de |

|  |  |
| --- | --- |
| **Abstract:** | Since the DASH/DAISAM Workshop in Berlin in January 2020 we have been exploring options to implement an assessment platform that can be used to perform health AI evaluation for the different topic groups. So far, this has resulted in two code bases which we are currently working on: (a) custom assessment platform and (b) evalai-based assessment platform. |

Since the DASH/DAISAM Workshop in Berlin in January 2020 we have been exploring options to implement an assessment platform that can be used to perform health AI evaluation for the different topic groups. So far, this has resulted in two code bases which we are currently working on: (a) custom assessment platform and (b) evalai-based assessment platform. Code repositories for both frameworks can be found and are maintained at:

<https://dev.azure.com/mllabai/FG-AI4H%20Assessment%20Platform>

# (a) Custom Assessment Platform

<https://dev.azure.com/mllabai/_git/FG-AI4H%20Assessment%20Platform>

A basic manual for how to interact with the code can be found in the ReadMe of the repo. Currently it goes as follows:

1. Clone the repo

```shell

git clone https://gitlab.hhi.fraunhofer.de/fgai4h/assessment-platform.git

```

2. From directory run the below to build docker image

```shell

sudo docker build -t assessment-webservice .

```

3. Run the docker image

```shell

sudo docker run assessment-webservice

```

4. Run the docker image in interactive modes (for dev)

```shell

sudo docker run -it assessment-webservice /bin/bash

```

5. Get container id

```shell

sudo docker container ls

```

6. Pick container id and run the following to obtain container ip

```shell

sudo docker inspect "container id" | grep "IPAddress"

```

7. Query the running container for results (mock atm) using the container ip

```shell

curl --request POST 'http://172.17.0.2:5000/assessment\_api/getResultMock' --data '{"image\_path":"./testset/patch1.png"}' --header 'Content-type:application/json'

```

8. Run the full test scenario with CSV output for PDF creation

```shell

python Local\_2\_API.py <parentfolder-of-test-images>

```

9. Stop container

```shell

sudo docker stop $(sudo docker ps -aq)

```

10. Stop and Delete images to make space for new build

```shell

docker stop $(docker ps -aq) #Stop all running containers

docker rm $(docker ps -aq) #Remove all containers

docker rmi $(docker images -q) #Remove all images

```

Conceptually, the workflow looks as follows:

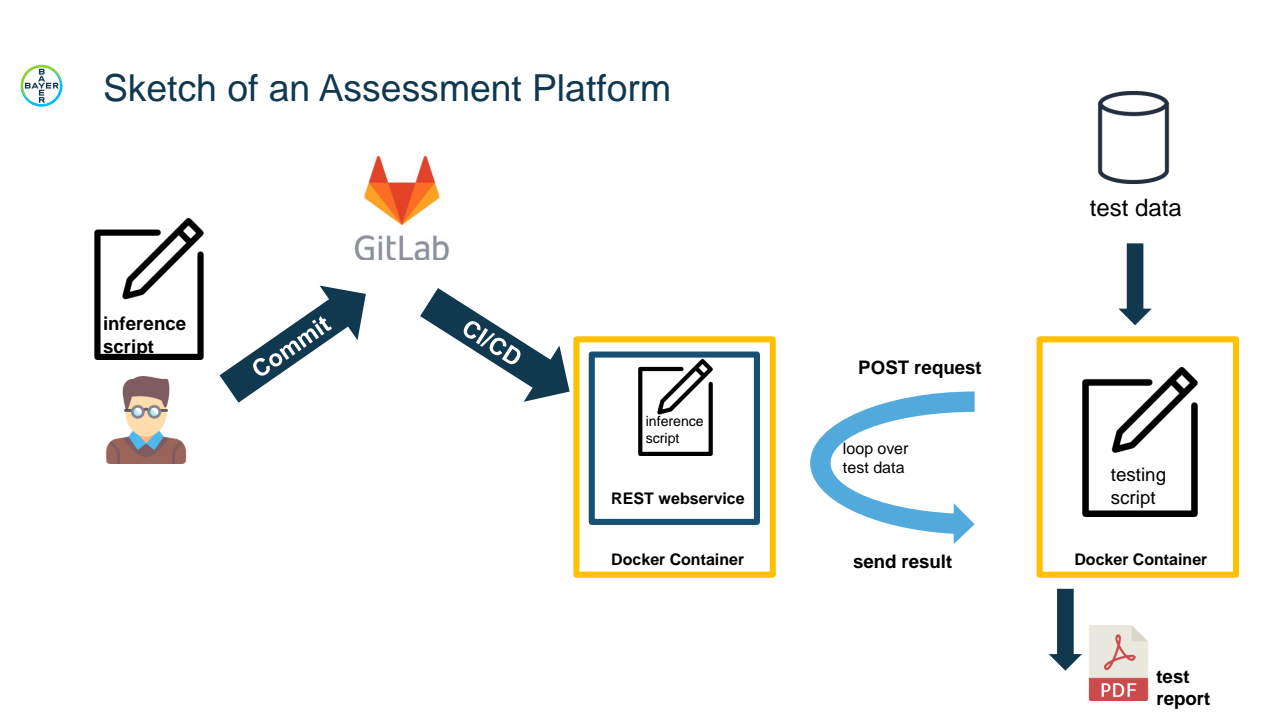


Figure 1 – Sketch of an assessment platform

# (b) evalai-based Assessment Platform

<https://dev.azure.com/mllabai/FG-AI4H%20Assessment%20Platform/_git/EvalAI.git>

A basic manual for how to interact with the code can be found in the ReadMe of the repo. Currently it goes as follows:

1. Preliminaries
   1. Install Docker [***Install Docker Engine on Ubuntu***](https://docs.docker.com/engine/install/ubuntu/)
   2. Install Docker CE [***https://docs.docker.com/compose/install/#install-using-pip***](https://docs.docker.com/compose/install/#install-using-pip)
2. Install evalai (reference: [***Installation — EvalAI 1.1 documentation***](https://evalai.readthedocs.io/en/latest/installation.html))
   1. git clone https://github.com/Cloud-CV/EvalAI.git evalai && cd evalai
   2. docker-compose up –build
      1. User GUI can be accessed at localhost:8888/
      2. Admin GUI can be accessed at localhost:8000/admin

The platform currently knows the following user roles

Admin: someone who manages the evalai instance

Host: someone who hosts and manages a challenges on an evalai instance

User: someone who participates in a challenge and uploads preds/models to the platform

A full documentation is maintained at <https://evalai.readthedocs.io/en/latest/index.html> for reference.

# Next Steps

We will announce our plans for the next steps at the upcoming meeting to discuss with the FG-AI4H plenary.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_