## Training program

### Target:

* Enough skills to work on real project & bring values

### Main skills to evaluate:

* Self-learning
* Problem-solving skills
* Presentation and Communication

### Supporting information

* Port: <http://192.168.100.131:5678/lab>?
* Daily in [Scrum Process Updated](https://docs.google.com/document/d/1rBCPi-KH1tsa4sJa_HE7B9OVQZkBL-6Y0Tdv4l8_37I/edit#heading=h.ohp05892hj30)
* [LAB Server Guideline](https://docs.google.com/presentation/d/17GZZsN8nZ_GaNRHtx5_NYzSDrtQAopZY_pwlSd70iOI/edit?usp=sharing)
* Please request to create personal folder for reporting
  + GANTT chart for planning
  + Report file for updating current process

### Knowledge Sets:

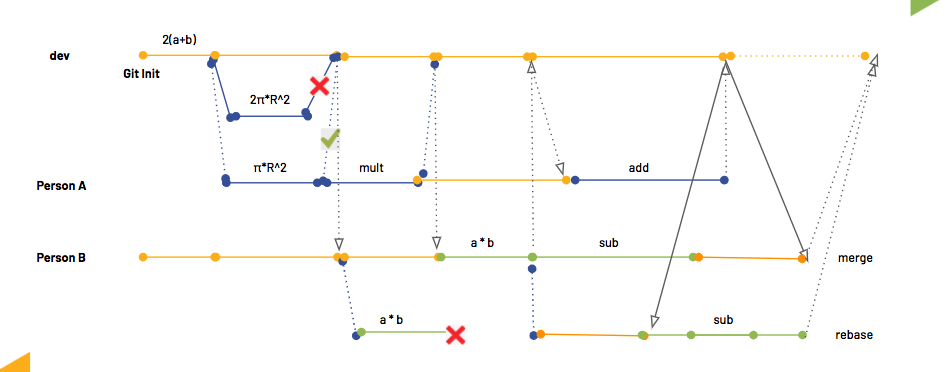
**Prefer Read & Write report in English**

1. Github (how to interact with each other) (1 days)  
   Guideline: [NDT\_Git Introduction](https://docs.google.com/presentation/d/1MD4YylNhgczxlfYWTRba6dc7L9qLw2Px44zzLiLVjjI/edit?usp=sharing), [Video](https://drive.google.com/file/d/1sJ1oLrA_Gv0tvlh93b1WmPQCdk0L4pbC/view?usp=sharing)

Output: URL to each repo

* 1. ([Source videos](https://drive.google.com/drive/folders/1Sgg_KV7kJNKIYh-sFGaPgjCKJrZQQSW4)) Create repo follow video clip
  2. (URL 2) Test: only look at the diagram, recreate repo  
     Recreate the repo with 2 following information. Duration: 30’

<https://gist.github.com/dinhtungtp/759106c8242316ec4be36d9f21ccc542>



1. Data Engineering: Pandas and Data Analysis => *need to add training data here*
2. Language & Framework: Pytorch & Pytorch Lightning  
   Guideline: [NDT\_Pytorch Lightning](https://docs.google.com/presentation/d/13ZDZO0bvFPHmAVUwwpZvVzUNXpRBh6w6FdCaLWeGdtE/edit?usp=sharing), [Video](https://drive.google.com/file/d/10x9e9q5z_HHrhgkKXM1yIA9HtONihmJ3/view?usp=sharing)
   1. Run training code (scratch) (3 days)
      * Data: **/home/tungnguyendinh/.fastai/data/pascal\_2007/**
      * Output: URL to ipython notebook
        + Multi-label classification problems
        + Requirement: tensorboard logging, visualization (filter visualization), results > 92%
   2. Convert to pytorch lightning (1.5 days)
      * Output: URL to ipython notebook
        + Rerun and keep the same results with normal code
        + Create callback: logging, visualization, load pretrained model
   3. Run pytorch lightning with new problems (0.5 days)
      * Data: **/home/tungnguyendinh/.fastai/data/oxford-iiit-pet/images**
      * Output: URL to ipython notebook
        + Pet breeds problem
        + Reuse callback
        + Run few functions of trainer: *auto\_lr\_find, accumulate\_grad\_batches, limit\_training\_batches, num\_sanity\_val\_steps*
3. Paper reading

Paper reading guideline: [Link](https://mahogany-hortensia-08b.notion.site/Guideline-to-read-research-papers-c8eab18a8d234f03ad4c7f2846c4dfbc)

* 1. Task 1: Paper-wise problem: Understanding a solution in video processing: SlowFast paper (1 days)
     + Read SlowFast Paper and present the paper
     + Output: URL to GG Docs
       - What is the meaning and paper, their main techniques, and why does they help
       - Present how you follow guideline (need to include images)
  2. Task 2: Field-wise problem: Find general ideas of main paper in Object Detection (3 days)
     + R-CNN, Fast-CNN, Faster-CNN, R-FCN, FPN
     + SSD, YOLOv3, RetinaNet…
     + Output: URL to GG docs
       - How these techniques different from each other
       - Present how you follow guideline (need to include images)
       - Presentation

1. Read and Implement other codebase (3 days)

* Retina repo: <https://github.com/yhenon/pytorch-retinanet>
* Output:
  + - (URL to docs) Draw repo diagram
    - (URL to codebase) Add gradcam visualization
      * <https://github.com/jacobgil/pytorch-grad-cam>

1. Follow team process (0.5 days)
   1. Team Process: <https://pixta.esa.io/posts/21154>
   2. [LAB Server Guideline](https://docs.google.com/presentation/d/17GZZsN8nZ_GaNRHtx5_NYzSDrtQAopZY_pwlSd70iOI/edit#slide=id.p)
   3. Output: Doing test on GG form
      * Answer questions <https://forms.gle/GwbPDRyUmWb8u6AJ9>
2. (Required Home/Office) Deep Learning Specialization Course (theory) (3 days)
   1. Finish 5 courses
      * Output: (URL to excel summary of the course)
        + GG Docs/excel, presentation
        + Test
        + Example of output: [CS50 course](https://docs.google.com/spreadsheets/d/0B1rcjy2xlbTUaWdBLTFDM2tlQkk/edit?usp=sharing&ouid=103251380706581023835&resourcekey=0-vs2uUEApmaioesjA82EsVQ&rtpof=true&sd=true) summary
3. (Required Home/Office) Clean code (2 days)
   1. Read 1 chapter and share
      * Output: (URL to docs that summary the content of a chapter and instruction)
        + Esa document, presentation
        + Test
        + Requirement: Main rules & examples
4. (Optional) FastAI (framework + practical)

**Naming of docs / ipython note:**

* Format [Section + Name]
* Example:
  + 3a. Understanding SlowFast paper
  + 4. Repo diagram
  + 4. Gradcam to a Codebase

### 

### Training Time

* Maximum Learning In Office Time: 1 month (current estimate 17 days)
* Work on real project: 2 months
* Criteria to pass internship:
  + (tentative) Learning time at the company < 1 month
  + Pass all the output at the end of the internship

Right access, when to share rights

When to allocate to real project

* Project need and person have a suitable skill
* Person is strong, and be able to join full time project
* Do additional & meaningful projects

## Working program

Potential projects:

- FR: connect mask vs no-mask face

Metric learning is an approach based directly on a distance metric that aims to establish similarity or dissimilarity between images.

- Auto retraining for IC, RA (15/4) - intern

- Quantization: RA, JIC (intern)

- Extend RA to new domain (number plate)

- MultiGAN

- Data Annotation: Human in a Loop (AI + human + process)

Real & current project: IAA, LTR

## Event

* Team Seminar: 2pm Friday
* Sharing real project: Tuesday (30’)