**Project management:**

* Establish the conditions/assumptions.
* Decide and set up the management structure (GIT/TRELLO/TEAMS/…)

**Wat zijn de “grote project onderdelen”:**

* Data collection.
  + Decide with which “sensor” type (lidar/sonar) to use in 1st instance
  + Identify the labels for the dataset.
    - If S3 > S1 Angle = -22
    - If S1 > S3 Angle = 0
    - If S2 > S1 Angle = 22
  + Understand the present code.
  + Run de circuit 3 times and store the data
* Data interpretation.
  + Feature selection.
* Develop the neural network.
  + Define problem type – regression/classification
  + Define inputs – outputs.
    - How to “load the data”
    - Define structure NN (number of layers, initial weights and bias).
  + Identify training intensity.
  + Establish loss function to be used.
* Train, test and validate the model.
  + Build and execute the training module.
* Implementation
  + Connect the developed module to the existing software
  + Test the integrated software.
  + Transform de notebook to a “transferrable” software package

**Deliverables:**

* Vastlegging van inzichten en experimenten in een Jupyter Notebook
* Korte onderbouwing van gemaakte keuzes (ontwerp, implementatie)
* Overzicht van de testresultaten (nauwkeurigheid, foutmarges) en advies (bruikbaarheid?)
* Uitdraai van git historie (geeft een beeld van software development proces)
* De uiteindelijke broncode moet als zip-bestand worden ge-upload naar je eigen Teams kanaal. Volg hiervoor de stappen op <https://github.com/AlxcNL/MakeAIWork2/blob/main/PROJECT_EXPORT.md>

Upload bovenstaande **vóór de deadline op maandag 8 mei, 12:00 uur** naar:

Teams > Applied Artificial Intelligence - Make IT Work >   
Jouw Private Channel > Files > Deliverables periode 2 > Project 2

Zorg daarbij voor de volgende mappenstructuur:

* Notebooks – NN-car\_V2 / model\_Lider\_vs1
* Onderbouwing – lower in this document
* Advies – no need to do something with this subject
* Git-historie
* Broncode

**Questions (to the teachers) about the deliverables:**

* Would it be sufficient to base the project only on sonar? = OK; doch het is makkelijk Lidar toe te voegen.
* What does advice mean – copy
* What should the “broncode” contain
* Why is our loss curve “spiky”? = geen probleem

***To do:***

* Finalize the model and the implementation:
  + The car drives using the following (amended) files
* W**e need to provide (in a separate document) an argumentation of the main choices:** 
  + Problem solving approach: use of Relu
  + Usage of the number of layers and nodes.
    - Sonar:
    - Lidar:
  + Which number of epochs used and the learning rate:
    - The number of Epochs and the learning rate is a bit different between Sonar and Lidar:
      * Sonar: with a “low” learning rate (0.0001) yields the best accuracy rate. This in combination with 100 or 250 epochs
      * Lidar:
  + Which choices made with the integration of “trained self steering module” in the existing software
    - We decided to “refactor” the “hardcoded copy” by adding code for the integration of our Sonar and Lidar models in the “…..”Sweep class allowing to “manage” the steering angle through the trained model(s). Consequently we changed
  + Lidar versus sonar -
* **Create an overview of the test results of the self driving car:**
  + **See “Training Differentiations\_ovv.xls”**
* **Develop the final “broncode”**