	infrastructure to incorporate AI solutions into the business.	

PROFESSIONAL QUALIFICATION LEVELS

1. Associate AI Engineer

An Associate AI Engineer (Assoc AI Engineer) is an individual who has the necessary technical skills and knowledge to start working on an AI project for a business organisation that has some data capabilities.

1.1 Required technical knowledge and skills

Candidates need not have working experience in AI-related roles but must be proficient in programming, understand databases and are able to run basic SQL queries. The candidate should also have some knowledge of software engineering, AI cloud services, data engineering and data management.

1.3 Assessment

There are two parts to the Assoc AI Engineer assessment:

a) Technical test

The technical test is a home-based assessment that is emailed to the candidate's registered email address. Candidates have to adhere to all the instructions stated in the technical test document and submit their solution by the stipulated deadline. Late entries will not be entertained.

Candidates will be given a dataset and are required to submit a solution consisting of following sections:

a. Exploratory data analysis
b. End-to-end machine learning pipeline

The preferred programming language is Python or R. Instructions on packaging and submitting the deliverables are provided in the test document. Mock test sample is available for download on the webpage.

AIP will evaluate each candidate's submission based on a set of assessment rubrics. Candidates who pass the technical test will receive an email notification to attend an assessment interview in person.

AIP will run through plagiarism check for all submissions. Candidates caught cheating will be disqualified and no refund shall be given. AIP decision will be final.

b) Assessment Interview

During the assessment interview, candidates will need to do a simple presentation of their submitted codes to a panel comprising industry AI professionals and Chartered AI Engineers. They should be prepared to explain the concepts that they utilised in the technical test and will also be asked questions to ascertain their technical knowledge. For more details, please refer to the CAIE assessment rubrics in Annex A.

Candidates are required to bring their own laptops/computing devices for the interview.

1.4 Application process

The CAIE application form is available on AISG's website at https://www.aip.org.sg

Information on the opening and closing dates for applications will be announced through the AIP mailing list, website and social media channels.

Candidates will need to prepare and submit:

a) An updated LinkedIn profile or curriculum vitae (CV)

A **non-refundable** application fee will be collected upon submission of the application. Candidates will receive an email confirmation if their registration is successful.

1.5 Application fee

There is a **non-refundable** application fee of S$500 nett.

1.6 Professional qualification validity

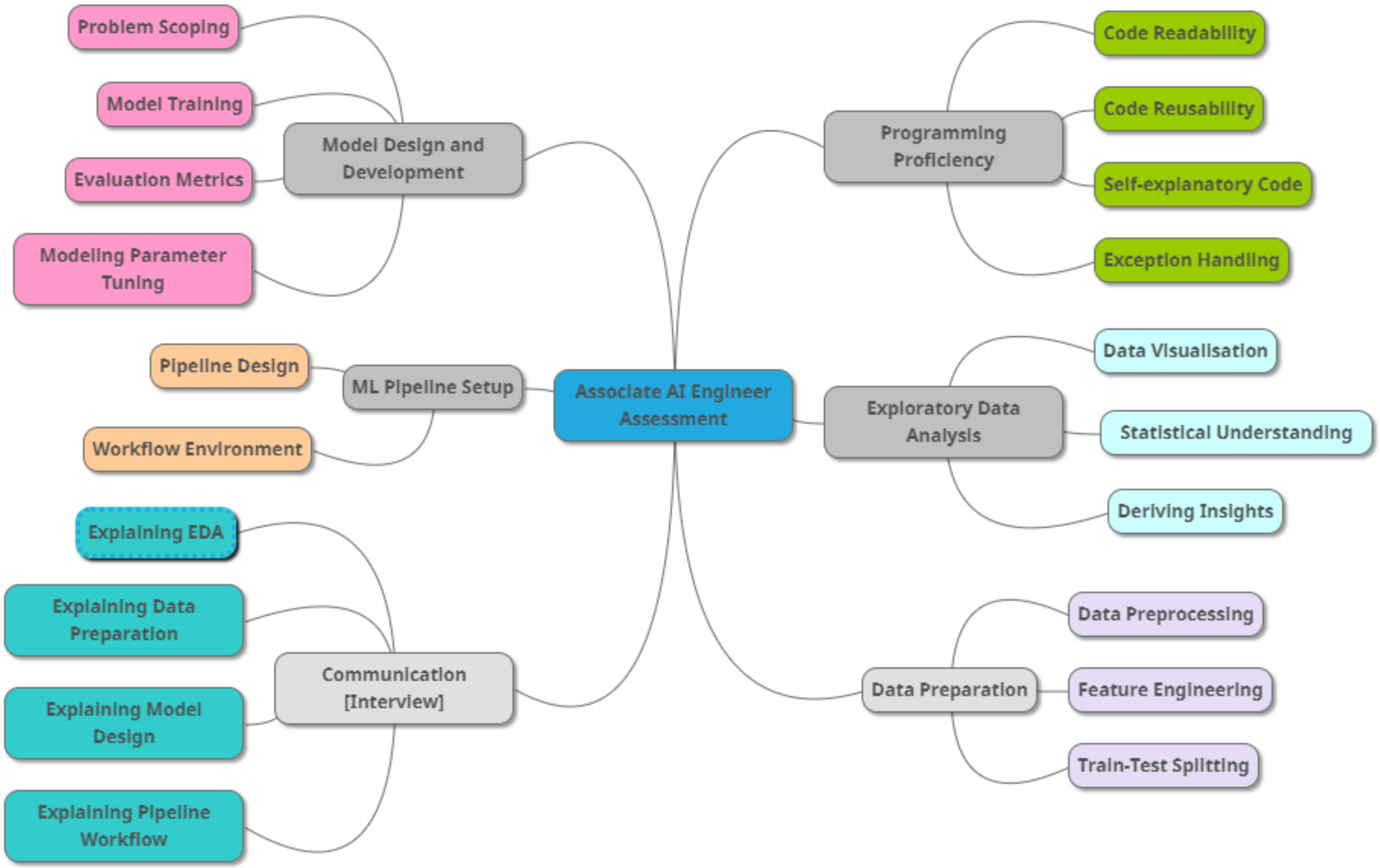
Candidates who fulfil the requirements for Assoc AI Engineer will receive an electronic certificate issued by AIP. Additional charges apply if the candidate wishes to receive a hardcopy of the certificate.

Qualified engineers will be allowed to display their professional credentials and the CAIE logo on their business cards, email signature and other corporate marketing materials.

The qualification is valid for two years. Upon expiry, renewal is required for continued display of the accreditation. Please refer to the sections on **display of credentials** and **professional qualification renewal** for more details.

ANNEX A1:

Assessment rubrics for Associate AI Engineer



1. Programming Proficiency Assessment	
Code readability	● Code is organised in a clean and readable format ● Code layout style, such as indentation and maximum line length, is consistent with industry conventions such as PEP8 (https://www.python.org/dev/peps/pep-0008)
Code reusability	● Code is structured to minimise repeated code (e.g. using Functions)
Self-explanatory code	● Proper naming conventions for variables, functions, classes ● Include documentation with in-line comments in source code ● Intent of Functions and Classes is stated clearly
Exception handling	● Includes codes to catch and handle exceptions such as insufficient data, empty dataset, wrong data type etc

2. Exploratory Data Analysis (EDA) Assessment	
Data visualisation	● Uses appropriate visualisation tools to show the relationship between variables
Statistical understanding	● Generates and provides good explanation of descriptive statistics (mean, median, mode, standard deviation, and variance) ● Includes analysis of relationship between variables
Deriving insights	● Extracts and explains insights from the EDA

	● Links insights into a coherent story ● Engineers features based on the insights drawn

3. **Data Preparation Assessment**	
Data preprocessing	● Performs Missing Data Analysis and takes care of the missing data appropriately by removing data records with missing values or performing data imputation ● Performs Outliers Analysis and takes care of outliers appropriately by removing outlier records or replacing outliers data ● Performs data investigation to check for erroneous data and performs appropriate data preprocessing to correct the erroneous data
Feature engineering	● Performs basic transformations on data; e.g. mathematical transformation and binning of numerical data; string replacement, substring extraction and concatenation of multiple strings for string data ● Performs basic feature engineering to improve model accuracy
Train-test - validate splitting	● Performs basic train-test-validate data split for model building to ensure that the final model is not overfitted and model testing is unbiased

4. Model Design and Development Assessment	
Problem scoping	● Able to define modeling objective based on problem statement ● Able to list down appropriate assumptions and considerations related to the problem scoping
Model training	● Displays good conceptual understanding of ML algorithms and models ● Builds an appropriate model for the task using an established ML framework
Evaluation metrics	● Evaluates model performance using suitable metrics ● Able to explain the core concepts used in model selection, such as the trade-off between variance and bias
Modelling parameter tuning	● Takes into account different modelling parameters and architectures for comparison

5. ML Pipeline Setup Assessment	
Pipeline design	● Designs modular pipeline to ingest data, perform data cleaning and data transformation, train models, generate evaluation metrics and make inferences ● Automates pipeline workflow ● Functional pipeline runs successfully end-to-end ● Includes a README file to describe how to run the pipeline
Workflow environment	● Sets up an ML workflow environment and file structure to facilitate pipeline ● Includes a library versioning requirements file, e.g. “requirements.py” or “conda.yml” depending on the setup option

6. Communication [during interview]	
Explaining EDA	● Able to draw insights using EDA ● Able to collate insights into a coherent story and articulate it clearly
Explaining data preparation	● Able to justify the methods used to input missing values, correct erroneous data or handle outliers ● Able to show how the new features help with model improvement ● Able to explain the rationale for adopting the train-test-validate data split strategy
Explaining model design	● Able to explain how the solution addresses the problem statement ● Able to clearly articulate the benefits and drawbacks of the ML solution

Explaining pipeline workflow	● Able to explain how the workflow is designed ● Able to explain how to embed the evaluation of the solution developed in the workflow ● Able to identify and address blockers that may arise in the workflow (data curation, model training and inferencing, deployment)

Official (Open)

Rubrics [Group – Technical Skills and Knowledge]

	Excellent	Good	Satisfactory	Need Improvement
	8 – 10 points	5 - 7 points	3 - 4 points	0 - 2 points
Overall Analysis (10 points)	• In-depth Data Analysis and insights • In-depth analysis of models' performance and possible follow-up actions	• Some Data Analysis and insights • Some analysis of models' performance and possible follow-up actions	• Limited data analysis and insights • Some incorrect analysis of models' performance and possible follow-up actions	• Minimal or incorrect data analysis and insights • Incorrect or no analysis of models' performance or possible follow-up actions
Overall Design (10 points)	• Sound justification of choice of models • Pipeline handles all aspects from data ingestion to model evaluation	• Some valid justification of choice of models • Pipeline handles most aspects from data ingestion to model evaluation	• Some invalid justification of choice of models • Pipeline handles some aspects from data ingestion to model evaluation	• Little or invalid justification of choice of models • Pipeline handles minimal aspects from data ingestion to model evaluation
Use of tools and libraries (10 points)	Seamlessly applies docker and kedro in building and deploying the machine learning pipeline	Some use of both docker and kedro in building and deploying the machine learning pipeline	Some use of either docker or kedro but not both in building and deploying the machine learning pipeline	Did not apply docker nor kedro in building and deploying machine learning pipeline
Version Control (10 points)	Code changes are consistently tracked and uploaded to a version control repository, with detailed commit messages and consistent version history.	Code changes are regularly tracked and uploaded to a version control repository with unclear commit messages.	Code changes tracked with inconsistent commits and unclear commit messages	Little to no evidence of version control. Repository may exist but is not utilized effectively to track changes

Rubrics [Group – Non-Technical]

	Excellent	Good	Satisfactory	Need Improvement
	8 – 10 points	5 - 7 points	3 - 4 points	0 - 2 points
Presentation Content Organisation (10 points)	• The presentation is organised in a logical sequence that flows naturally. • Presentation load is well distributed.	• The presentation is organised in a sequence that can be followed. • Presentation load is evenly distributed.	• The presentation is organised. • Imbalanced presentation (some members presented much more than others).	• The presentation is difficult to follow • Presentation done by less than 50% of the members
Work Independence (10 points)	Consistently works independently, requiring minimal guidance, and proactively solves problems.	Works independently on most tasks but occasionally seeks clarification or assistance.	Requires frequent guidance and supervision to complete tasks. Struggles to work independently.	Requires consistent guidance and supervision to continue with the project

Rubrics [Individual – Technical Skills and Knowledge]

	Excellent	Good	Satisfactory	Need Improvement
	12 – 15 points	8 - 11 points	4 - 7 points	0-3 points
Code Walk-through (15 points)	Delivers a clear, thorough, and well-structured code walk-through, explaining logic, decisions, and functionality in detail.	Provides a solid code walk-through that covers key components and logic, though some details may be less clear.	Presents a basic walk-through with limited explanation of the code and its logic.	Code walk-through is incomprehensible and sometimes illogical
Code Quality (15 points)	• Code is always structured to minimise repetition. • Code is always organized in a readable format • Intent of all functions and classes are stated clearly in the code • Extensive use of configuration files	• Code is mostly structured to minimise repetition. • Code is mostly organized in a readable format • Intent of most functions and classes are stated clearly in the code • Some use of configuration files	• Code is rarely structured to minimise repetition. • Code is rarely organized in a readable format • Intent of some functions and classes are stated clearly in the code • Minimal use of configuration files	• Code is not structured to minimise repetition. • Code is unorganized • Intent of functions are not stated • No use of configuration files

Rubrics [Individual – Non-Techincal]

	Excellent	Good	Satisfactory	Need Improvement
	8-10 points	5-7 points	3-4 points	0-2 points
Handling Q&A (10 points)	Answers questions clearly and confidently. Handles challenging or unexpected questions with ease, providing thorough and insightful responses.	Responds to most questions effectively. May need some time to think or seek clarification but provides reasonable answers overall.	Struggles to answer questions fully or clearly, often providing vague or incomplete responses. Has difficulty handling challenging questions.	Unable to answer most of the questions. Answers may not be correct.

Project Guidelines

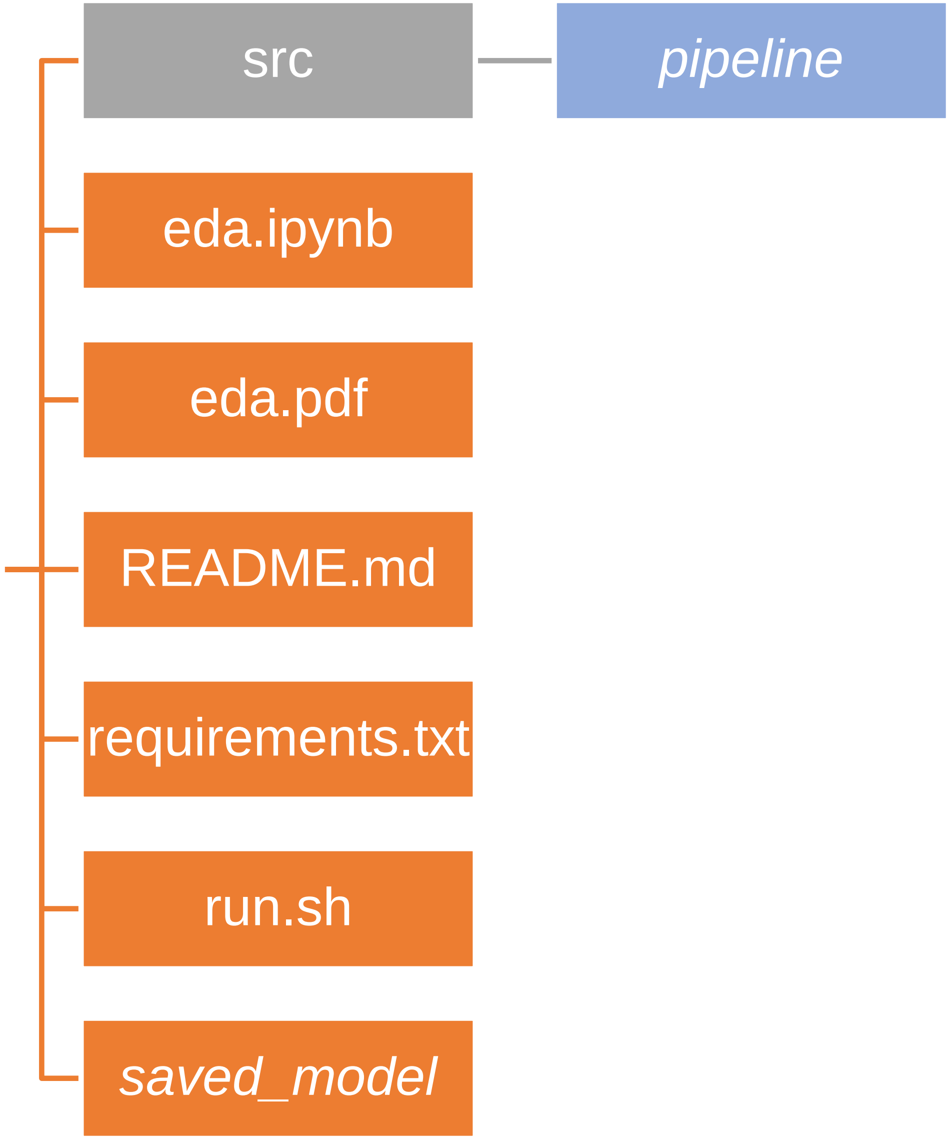
Project – 40% of the Learning Unit

PROJECT	Project presentation (team)	Week 7
	Project work submission (team)	Week 7

Objective

- Perform EDA and document insights
- Document choice of machine learning models
- Build ML pipeline
- Practise presentation skills

Project Requirement



src

pipeline

eda.ipynb

eda.pdf

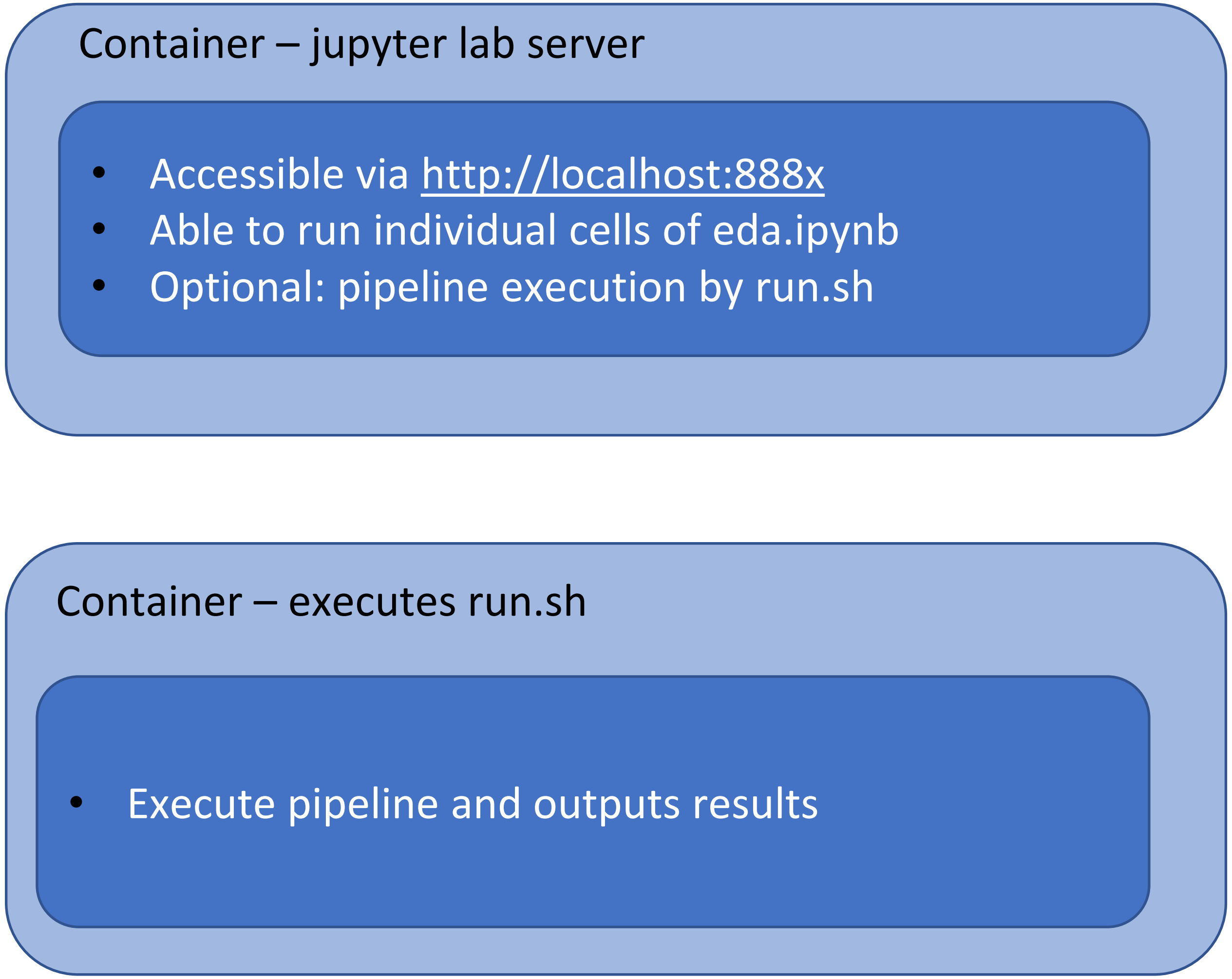
README.md

requirements.txt

run.sh

saved_model

Project Requirement (Bonus)



Container – jupyter lab server

- Accessible via http://localhost:888x
- Able to run individual cells of eda.ipynb
- Optional: pipeline execution by run.sh

Container – executes run.sh

- Execute pipeline and outputs results

Project Requirement

- At least n tasks
  (n is the number of members in the team, min 3 pax in the team)
  - EDA (Data Cleaning / Data Analysis / Feature Engineering)
  - Model Training
  - Model Inference & Reporting
- Each member to present and do code walk through on key parts of their contributions
- Version Control
  - Code changes should be consistently tracked and uploaded to a version control repository with clear commit messages.

Group Formation

- 4 students per group
- Submit students name list (with a team name) on BrightSpace under Assignment -> Group Name List Submission
- One submission per group
- Group information submit by end of week 4 (16 May 2025)

Deliverables – Group Presentation (Week 7)

- Each team will present your project work during class in a 20-minute presentation
  - 5-minute presentation, 5-minute code walk through, 10-minute Q&A
  - Each team member will be asked to briefly walk through the code he/she manages.
  - Indicate contributions by each team member.
- Presentation should include (but not limited to) the following:
  - Highlight key parts of your EDA (null, erroneous value handling, insights, feature engineering)
  - Machine Learning Pipeline (e.g ingestion, hyperparameter tuning, model selection)
  - Comment on Model's Accuracy
  - Further Enhancement or improvements (if any)
  - Others

Deliverables – Group Reflection & Submission (Week 7)

- ~~Reflection:~~ (No reflection)
  - ~~Each team member writes their reflection in the same document. Indicate your name on the section that you have written.~~
  - ~~Reflection to focus on knowledge, skills and attitudes that you have personally changed, acquired or action plans related to KSA.~~
  - ~~Not more than 400 words per person.~~
- Please put all work files and reflection document in ONE Zip folder and submit to BrightSpace, following naming convention below.

Zip Folder Name: EGT309_T1/2_<project team name>.zip.

- All codes files, presentation slides and reflection document should be compressed into the folder for submission

Submission Deadline

~~6 June 2025, 2359~~

8 June 2025 2359

- Submission time will be captured based on BrightSpace record
- Refer to Extension and Late Submission Policy 2024
- Zero mark will be given for late submission after 5 calendar days from due date.

Sharing yours or your team's work with other learners or teams is considered an act of plagiarism.
ALL parties involved, including those who share their work, will face discipline under NYP's Academic Integrity Policy.

SAMPLE

Associate AI Engineer Technical Test

January 2025

Deadline: [redacted]

Submissions after the deadline will not be accepted. Please submit the completed assessment early to ensure a smooth submission process.

This technical assessment consists of the following two main tasks:

1. **Exploratory Data Analysis (EDA)**
2. **End-to-end Machine Learning Pipeline**

The assignment project background:

Olist is a Brazilian e-commerce marketplace like Lazada, Taobao and Shopee, it is a sales platform that connects small retailers with customers.

The following is a summary of the buying process from a customer perspective. A customer can place a purchase order for multiple items from a vast selection of sellers on Olist. The sales order will then be fulfilled by a logistic partner. The customer will be notified of the estimated delivery date once shipment is confirmed. Upon delivery of the order, the customer will receive a survey to provide feedback.

The datasets provided here are randomly extracted real commercial data from Olist store. The anonymized data consists of order information for about 100k orders from year 2016 to 2018. The datasets include order details such as order status, price, payment, freight information, customer location, product attributes, and customer reviews.

One possible strategic approach to drive profitable sales for an e-commerce platform is to determine who are their high value loyal customers, such that the business can design and deploy more effective and targeted marketing campaigns.

For this assignment, help Olist leverage on data analytics and machine learning, focusing their business objective to build their customer base and increase future sales revenue.

Source:
Brazilian E-Commerce Olist Store

The assignment project objective is:

To identify potential repeat buyers

Please attempt all requirements stated in the following sections and package a submission in zipped file formatting containing the deliverables specified.

1. Data Description

Dataset Download:

Please download the following datasets in csv format:

1. olist_customers_dataset.csv
2. olist_geolocation_dataset.csv
3. olist_order_items_dataset.csv
4. olist_order_payments_dataset.csv
5. olist_order_reviews_dataset.csv
6. olist_orders_dataset.csv
7. olist_products_dataset.csv
8. olist_sellers_dataset.csv
9. product_category_name_translation.csv

Data Summary:

This dataset provides randomly extracted data for orders made between 2016 and 2018.

Dataset Dictionary:

Please refer to the following data dictionary in excel format:

- olist_datadict.xlsx

Data source:

Brazilian E-Commerce Olist Public Dataset

Dataset License:

CC BY-NC-SA 4.0

Instructions:

Please make reasonable assumptions and explain the rationale of the assumptions based on the data provided and problem statement stated.

2. Exploratory Data Analysis (EDA)

Using the dataset provided, perform an EDA and create an interactive notebook that can be used to present and explain the findings of your analysis. (Python or R programming language preferred.)

The report should contain appropriate and sufficient visualisations and explanations to help assessors understand how insights are derived, as well as their implications on the design of your machine learning models.

All analysis related to data preparation, input selection and feature engineering should also be included in the EDA.

(Optional) You can also provide supplementary online dashboard presentations, e.g., online dashboard using Tableau or Power BI. Please include the link to this supplementary dashboard in your submitted **README.md**.

Deliverables:

1. EDA Notebook in Python or R Programming Language: an executable ".ipynb" or ".Rmd" file with the exact naming convention as follows: - **"eda.ipynb" / "eda.Rmd".**

2. Other programming languages are also allowed, but please make sure that all codes are included, with the results and findings write-up shown together with the codes. All scripts included should be running properly.

3. Please provide a copy of the Jupyter Notebook or Rmd (R Markdown) in pdf format as well.

4. Please make sure the EDA notebook is neat and well structured, and insights presented are focused and well-summarized.

5. Do not submit more than 20 pages (A4 size) for your EDA notebook submission. Additional information (beyond the 20 pages) can be included as Appendix in your submission, but Appendix will not be included for assessment.

6. The following is an example of folder structure to be delivered:

```
├── src
│   ├── dataprep
│   └── model
├── saved_model
├── README.md
├── eda.ipynb / eda.Rmd
├── eda.pdf
├── requirements.txt / packrat.lock
└── run.sh
```

Evaluation:

You will be assessed on the clarity of visualisations, depth of insights, presentation flow and structure of your analysis.

~~If you are shortlisted for the interview,~~ you are expected to be able to explain the thought processes and decisions you made throughout the analysis, and to demonstrate that you understand the underlying machine learning concepts. You may be requested to run your EDA notebook during the ~~interview~~ presentation, so please make sure the EDA notebook is able to run on your laptop.

Note: All groups will have to make a presentation

3. End-to-end Machine Learning Pipeline

Design and create a simple machine learning pipeline that will ingest/process the filtered dataset and feed it into appropriate machine learning algorithm(s), returning suitable metrics and outputs.

Deliverables: In place of 1-3, you can create a dockerfile, specify instructions to build and run docker image in your README.md

1. A folder named "src" containing Python modules/classes or R scripts, or other programming language scripts. Note that Python or R programming language is preferred, but other languages are also acceptable; all codes must be well structured and documented for **readability** and **can run successfully.**

2. An executable **bash script "run.sh"** at the root folder of your submission.

3. A "**requirements.txt" file or "packrat.lock"** file, or equivalent, at the root folder.

4. A "**README.md**" file that sufficiently explains the pipeline design and its usage. You are required to explain the thought process behind your submitted pipeline in the README.

5. The README is expected to contain the following:

    a. Full name (as in NRIC) and email address. Include names and Student ID of all team members

    b. Overview of the submitted folder and the folder structure.

    c. Include information about the **programming language (with version)** used, the run environment prerequisite (including OS platform and version) and the list of libraries or packages (with version) required for both the EDA and pipeline in the submission.

    d. Overview of key findings from the EDA conducted and the choices made in the pipeline based on these findings, particularly any feature engineering. Also include URL link for the supplementary EDA online dashboard, if any (optional)

    e. Instructions for executing the pipeline and modifying any parameters.

    f. Description of logical steps/flow of the pipeline. If you find it useful, please feel free to include suitable visualization aids (e.g., flow charts) within the README.

    g. Explanation of your choice of models for each Machine Learning task.

    h. Evaluation of the models developed. All metrics used in evaluation should be explained.

    i. Other considerations for deploying the models developed.

    j. **Do not submit more than 10 pages (A4 size) for your "README.md" submission**. Additional information (beyond the 10 pages) can be included as Appendix in your submission, but Appendix will not be included for assessment.

Pipeline Requirements:

1. All codes for the pipeline must be submitted. Codes submitted must be structured with welldefined functions, with good documentation.

2. A bash script named “run.sh” to run the above-mentioned modules/classes/scripts. DO NOT submit a Windows batch (“*.bat”) script in replacement of the bash script.

3. DO NOT install your dependencies in the “run.sh”; this will be taken care of when we assess the assignment if you have created your “requirements.txt” correctly.

4. Relevant training/evaluation metric(s) outputs to be generated upon completion.

5. Pipeline made easily configurable to enable easy experimentation of different algorithms and parameters, as well as different ways of processing data (e.g., use of a config file, environment variables, or command line parameters).

6. Python and R Programming Language are preferred for the submission. For Python, use only versions 3.7 and above. For R, use only versions 4.0 and above.

7. Other programming languages are also allowed, but all scripts must be running properly.

8. Please make sure that the pipeline codes can be executable successfully. README.md should include clear and comprehensive setup/running instruction.

9. Include at least one saved model in the folder “saved model” from your pipeline output.

10. DO NOT include the original raw data file in your submission.

11. The following is an example of folder structure to be delivered:

```
├── src
│   ├── dataprep
│   └── model
├── saved_model
├── README.md
├── eda.ipynb / eda.Rmd
├── eda.pdf
├── requirements.txt / packrat.lock
└── run.sh
```

Evaluation:

You will be assessed on the quality of your code in terms of clean separation of functionality, ease of use and readability. Code reusability between the tasks will be viewed favourably.
If you are shortlisted for the interview, you are expected to be able to explain the thought processes and decisions you made throughout your code, and to demonstrate that you understand the underlying machine learning concepts. You may be requested to run your pipeline during the interview, so please make sure the scripts are able to run on your laptop.

Submission Format

Submission Specifics:

1. Your work should be uploaded as a **“*.zip” file** to the designated upload link (details below).
2. The zip file size should not exceed **20MB**.
3. The zip file is to be named according to the following naming convention:

EGT309_T<1/2>_<project_team_name>.zip

~~**“<firstname>_<lastname>_<student id>.zip”**~~

~~e.g., “john_lim_2xxxxxT.zip”~~

4. The zip file should have a folder structure similar to the following:

Example:

~~~~
```
├── src
│   ├── dataprep
│   └── model
├── saved_model
├── README.md
├── eda.ipynb / eda.Rmd
├── eda.pdf
├── requirements.txt / packrat.lock
└── run.sh
```
~~~~

IMPORTANT NOTE:

- Non-conformance to the specified conventions/formats may negatively impact your evaluation.
- AIP will run through plagiarism checks for all submissions.
- **Candidates caught cheating will be disqualified.**

@end