Project Implementation Rubric

Structure (1 mark)
Organise your submission in a manner that facilitates the identification of relevant code for the various stages of your solution, e.g., data pre-processing, model training, etc.

Document your code such that someone outside of your team has a good chance of being able to follow it, should they wish to reproduce your results.

Prepare a README that includes: (1) appropriate attribution to your sources: any datasets, models, codebases you used; (2) links to any models that you trained and stored on the cloud; and (3) links to huge (>10MB) datasets you used and/or cleansed versions thereof, stored on the cloud.

Data (4 marks)
Demonstrate how you collected/built/pre-processed your datasets.

Illustrate how you checked for issues such as sampling bias, data imbalance and unreliability of data. Demonstrate how you attempted to address these issues, if applicable.

If applicable, demonstrate how you split your data into subsets for developing and evaluating your solution.

Solution (4 marks)
If you are training your own models, clearly show: how you defined the architecture, how you provided parameters and how you supplied it with training data (or any other resources, e.g., embeddings).

If some of your approaches involve feature engineering, clearly show how features were extracted.

Make your solution easily configurable, i.e., it should be possible for someone outside of your team to reconfigure your architecture/model parameters--it will also help you in your experimentation/evaluation.

If you are reusing existing models, demonstrate how you creatively adapted them for your own purposes.

Evaluation (3 marks)
Demonstrate your experiment set-up, i.e., how you systematically evaluated your chosen approaches.

Select suitable evaluation metrics and carefully consider how you should report overall performance.

Ensure that any comparison that you are performing is fair, and that there is no data leakage from the training/development stage to the evaluation stage.

Live Demo (6 marks)
Ensure that your solution can be run or applied during the live project demo, i.e., it should be straightforward to load any previously saved models.

Implement a simple functionality that allows a user to supply some input. Given new input, the solution should run and produce output in real time, i.e., within seconds.

Demonstrate deep understanding of your implementation by giving a walk-through of your Data, Solution and Evaluation, and answering challenge questions during the live demo.

Overall Aim and Learning (2 marks)
Be creative: where possible, demonstrate in your Solution and/or Evaluation how you incorporated what you have learned so far from COMP34812 and/or your own ideas.

Consider how your submission, overall, has allowed you to achieve the project aim you set out. If you have had to revise your project aim from the proposal, clarify/explain this in your README.