Toronto Metropolitan University
Department of Electrical, Computer, & Biomedical Engineering
Faculty of Engineering & Architectural Science

COE/ELE 70AB Milestones Compliance Report (MCR)

Project Title	MZ02 - AI-Powered Clinical Decision Support - Predicting Drug–Drug Interactions
MCR Number	IV (weeks 10 & 11)
Project Manager for the MCR period	Kartike Chaudhari, "Student A", responsible for weeks 10 & 11 activities and report in Week 11, Friday afternoon)
Team Players for the MCR period	Taha Ghori, Ahraz Kibria, Raidah Nazimuddin, Kartike Chaudhari
Faculty Supervisor	Muhammad Rehman Zafar

1. Tasks Outlined for the Reporting Period: MCR IV – Weeks 10 & 11

Group: Develop a detailed project management plan (Gantt chart, task breakdown) for the implementation phase (COE/ELE70B).

Student A: Draft the "Data" and "Methodology" sections of the final report. Contribute to the creation of the implementation Gantt chart.

Student B: Draft the "AI Model Design" section of the final report. Create detailed model flowcharts. Contribute to the creation of the implementation Gantt chart.

Student C: Draft the "System Architecture" and "Deployment Plan" sections of the final report. Contribute to the creation of the implementation Gantt chart.

Student D: Draft the "Testing and Evaluation" section of the final report.

2. **Progress Made in Reporting Period**(Provide detailed information on the progress that you (as a group and individual) made during the reporting period. You can include figures, datasheets, flowcharts, etc. and additional information as requested by your FLC. You should use your progress to justify compliance with the tasks outlined for the reporting period, as per the milestones submitted to your FLC in Week 3.

Student A:

During Weeks 10 and 11, my primary focus was drafting the "Data" and "Methodology" sections of the final 70A report and contributing to the development of the implementation Gantt chart. This required consolidating all of my prior work from MCR I, II, and III—particularly the data pipeline architecture and the finalized Data Management & Preprocessing Plan—into polished, formal documentation that fits seamlessly into the final design report.

For the Data section, I fully documented our selection of the DDIExtraction 2013 corpus (Herrero-Zazo et al., 2013), describing its structure, annotation scheme, sources, and advantages as a benchmark for biomedical relation extraction tasks. I also highlighted its known limitations, such as annotation inconsistencies (Kim et al., 2015) and class imbalance, which shaped several later design choices. I then formalized the complete data management framework developed in MCR III, including the Google Cloud Storage directory layout, the versioning strategy for raw and processed data, and the unified schema shared across the team. This ensures that all downstream components, particularly Student B's model architecture and Student D's evaluation system, operate on consistent, reproducible data.

I also expanded the description of the preprocessing pipeline into a detailed narrative suitable for a final report. This includes XML ingestion and validation, entity tagging using standardized <DRUG1> and <DRUG2> markers (Fu et al., 2023), tokenization using a PubMedBERT-compatible tokenizer (Gu et al., 2020), and the construction of balanced training splits using a combination of weighted sampling and controlled oversampling (Brownlee, 2020; Google, n.d.). Each transformation step now includes supporting explanations that clarify its role in overall model performance and data integrity. This version of the pipeline description is the most complete and refined iteration to date and is aligned fully with the team's finalized model design.

For the Methodology section, I integrated the entire system workflow into a cohesive explanation spanning preprocessing, model inference, and post-processing. This section describes how raw XML data progresses through the pipeline and becomes model-ready tensors, which are then passed into the PubMedBERT-based relation classifier (Gu et al., 2020) designed by Student B. I also summarized the handoff of model outputs to Student D's risk-scoring algorithm and contextualized both within the broader system architecture defined by Student C. To support clarity, I prepared a simplified data flow diagram that illustrates the path from raw text to final risk output. This section ensures that the entire methodology appears as a unified system rather than four isolated components.

I also worked closely with the team in drafting the implementation Gantt chart for 70B. My contributions focus on outlining the tasks related to re-implementing the preprocessing pipeline in a production environment, structuring GCS ingestion processes, writing unit tests for validation, and ensuring compatibility with the training environment on Google Cloud Platform. The resulting schedule now clearly defines data-related tasks, their dependencies, and their sequencing within the overall project timeline.

Student B:

The core of the model is the Microsoft/BiomedNLP-PubMedBERT-base-uncased-abstract-fulltext pre-trained Transformer encoder. This model is chosen over general-domain models as it has been pre-trained on a large-scale corpus of biomedical text, providing a significant advantage in understanding biomedical terminology (Gu et al., 2020). To effectively perform relation extraction, the model will use an entity-marking strategy where the two drug mentions in a candidate pair are marked in the input text with special tokens, such as [DRUG1] and [/DRUG1]. This approach explicitly provides the model with the precise boundaries of the entities to be related (Baldini Soares et al., 2019). To create a fixed-size representation for each drug mention, the final hidden states from the encoder corresponding to the start-marker tokens ([DRUG1] and [DRUG2]) will be extracted, producing two 768-dimensional vectors: drug1_vec and drug2_vec.

On top of this base encoder, several "heads" will be added to perform the specific tasks. The primary relation head, responsible for the DDI classification, takes the concatenated drug1_vec and drug2_vec vectors as input. This 1536-dimension input is passed through a dense layer with 768 units and a GELU activation (Hendrycks & Gimpel, 2016), used to maintain architectural consistency with the BERT encoder (Devlin et al., 2018). This is followed by a dropout layer and layer normalization, leading to a final output layer with a softmax activation to classify the relation into one of k classes (e.g., "Interaction-Type-A," "No-Interaction"). A secondary, auxiliary Token Classification (NER) head will be added as a multi-task objective. This head takes the entire last hidden state sequence from PubMedBERT, passes it through a dropout layer, and then through a dense output layer with a softmax activation applied to every token to predict its entity class (e.g., B-DRUG, I-DRUG, O).

The model will be trained using the AdamW optimizer, which is the standard for Transformers as it correctly implements decoupled weight decay for better regularization (Loshchilov & Hutter, 2019). The total loss function will be a weighted sum of the primary relation loss (Categorical Cross-Entropy) and the auxiliary NER loss (Categorical Cross-Entropy), balanced by a scalar hyperparameter, w_aux. A formal hyperparameter optimization study will be conducted using Vertex AI Vizier, with the objective of maximizing the Validation PR-AUC, the most informative metric for this imbalanced dataset. This study will explore a defined search space for key parameters, including the learning rate (log-scaled between 1e-6 and 5e-5), batch size (8, 16, or 32), weight decay (linear between 0.0 and 0.1), number of warmup steps (100 to 1000), head dropout rate (0.1 to 0.3), and the auxiliary loss weight (0.2 to 0.8).

Student C:

1. Implementation Gantt Chart (Group Task)

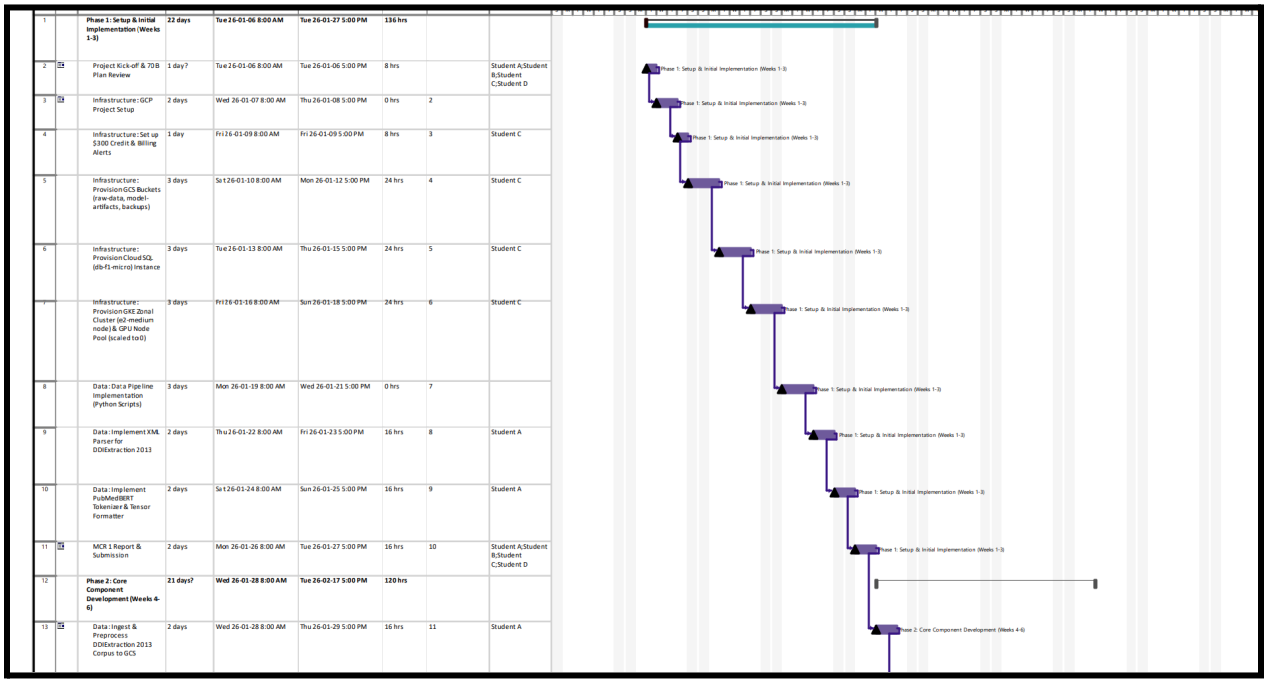
I collaborated with Students A, B, and D to develop the detailed project management plan (Gantt chart) for the COE/ELE 70B implementation phase. My primary contribution was to define and sequence all tasks related to infrastructure and deployment, based on my finalized MCR 3 design.

This included:

- Breaking down the GCP project setup into discrete tasks (Billing, GCS buckets, Cloud SQL, GKE cluster).
- Allocating time for the development of each microservice (Orchestrator, Risk Scoring, etc.).
- Defining the critical path for system integration, ensuring the NLP Inference Service and Risk Scoring Service are built and tested before the Orchestrator's core logic can be finalized.
- Ensuring the plan aligned with the COE 70B timeline and deliverables.

2. Drafted Final Report Sections (Individual Task)

I have drafted the "System Architecture" and "Deployment Plan" sections for the final 70A report. This content is derived directly from the finalized Infrastructure Design and Cost Analysis Document completed in MCR 3. These draft sections (included in Part 2 of this document) detail the finalized microservice architecture, the specific GCP service configurations (GKE Zonal cluster, db-f1-micro instance), the "Student Project" cost model, and the final budget, which confirms the $0 out-of-pocket cost via the GCP free credit.



		Phase 1: Setup & Initial Implementation (Weeks 1-3)	22 days	Tue 26-01-06 8:00 AM	Tue 26-01-27 5:00 PM	136 hrs		
2		Project Kick-off & 70B Plan Review	1 day?	Tue 26-01-06 8:00 AM	Tue 26-01-06 5:00 PM	8 hrs		Student A;Student B;Student C;Student D
3		Infrastructure: GCP Project Setup	2 days	Wed 26-01-07 8:00 AM	Thu 26-01-08 5:00 PM	0 hrs	2	
4		Infrastructure: Set up $300 Credit & Billing Alerts	1 day	Fri 26-01-09 8:00 AM	Fri 26-01-09 5:00 PM	8 hrs	3	Student C
5		Infrastructure: Provision GCS Buckets (raw-data, model-artifacts, backups)	3 days	Sat 26-01-10 8:00 AM	Mon 26-01-12 5:00 PM	24 hrs	4	Student C
6		Infrastructure: Provision Cloud SQL (db-f1-micro) Instance	3 days	Tue 26-01-13 8:00 AM	Thu 26-01-15 5:00 PM	24 hrs	5	Student C
7		Infrastructure: Provision GKE Zonal Cluster (e2-medium node) & GPU Node Pool (scaled to 0)	3 days	Fri 26-01-16 8:00 AM	Sun 26-01-18 5:00 PM	24 hrs	6	Student C
8		Data: Data Pipeline Implementation (Python Scripts)	3 days	Mon 26-01-19 8:00 AM	Wed 26-01-21 5:00 PM	0 hrs	7	
9		Data: Implement XML Parser for DDIExtraction 2013	2 days	Thu 26-01-22 8:00 AM	Fri 26-01-23 5:00 PM	16 hrs	8	Student A
10		Data: Implement PubMedBERT Tokenizer & Tensor Formatter	2 days	Sat 26-01-24 8:00 AM	Sun 26-01-25 5:00 PM	16 hrs	9	Student A
11		MCR 1 Report & Submission	2 days	Mon 26-01-26 8:00 AM	Tue 26-01-27 5:00 PM	16 hrs	10	Student A;Student B;Student C;Student D
12		Phase 2: Core Component Development (Weeks 4-6)	21 days?	Wed 26-01-28 8:00 AM	Tue 26-02-17 5:00 PM	120 hrs		
13		Data: Ingest & Preprocess DDIExtraction 2013 Corpus to GCS	2 days	Wed 26-01-28 8:00 AM	Thu 26-01-29 5:00 PM	16 hrs	11	Student A

Figure 1: Snippet of our Gantt chart and tasks for the upcoming semester (Submitted separately)

Student D:

During Weeks 10 and 11, my main focus was drafting the “Testing and Evaluation” section of the final 70A report and contributing to the implementation Gantt chart for 70B. This work expanded on the Evaluation Metrics and Testing Methodology Document I completed in MCR III, transforming it into a formal framework that defines how model performance and reliability will be verified during implementation.

In the Testing and Evaluation section, I documented the full testing pipeline, including unit, integration, and end-to-end validation procedures. I specified that model performance will be assessed using Precision, Recall, and PR-AUC, as accuracy is a well-known misleading metric for imbalanced datasets, while PR-AUC is more informative (Saito & Rehmsmeier, 2015). I also described the Stratified 10-Fold Cross-Validation process, which is essential to ensure each fold contains a representative sample of the rare, positive interaction classes and thus provides an unbiased performance estimate (DigitalOcean, 2025).

I refined the calibration plan by outlining how Temperature Scaling will be applied as a post-processing step. This is a critical safety measure, as modern neural networks are often "poorly calibrated" and "overconfident" in their raw probability outputs (Guo et al., 2017). The calibration will be validated using the Expected Calibration Error (ECE) metric (Pleiss, 2017). This ensures that probability scores from the AI model remain clinically interpretable and safe for decision support.

I also summarized the error analysis process, which categorizes model misclassifications into clinically relevant types based on established clinical NLP taxonomies (Fu et al., 2024), allowing us to guide future improvements.

Lastly, I worked with the team on the 70B implementation Gantt chart, defining milestones for model evaluation, calibration verification, and system validation. These contributions ensure that testing activities are clearly scheduled and integrated into the overall implementation plan.

3. **Difficulties Encountered in Reporting Period** (Provide detailed information on the difficulties and issues that you encountered during the reporting period and how you plan to address these in the following periods)

Student A:

The main challenge during this period involved transforming a large body of technical work from previous MCRs into cohesive, high-quality report sections without losing precision or clarity. Because the Data and Methodology sections depend heavily on the work of other team members, ensuring consistency in terminology, assumptions, and data formats required ongoing collaboration. For example, the final data schema had to be repeatedly cross-verified with Student B's finalized model input requirements and Student D's evaluation metrics to avoid mismatches in tensor formats or label definitions. Maintaining internal consistency across multiple interdependent documents was time-consuming but necessary to ensure the accuracy and professionalism of the final report.

Student B:

When planning the model, we hit a few big challenges. The main one was the black box problem. We had to choose: do we use a simpler model that's easy to explain, or the powerful Transformer we planned, which can't explain its reasoning? In a hospital, the "why" is just as important as the answer, so this was a huge trade-off. We're betting we can add features to help it explain itself, but it's a big risk since doctors won't trust a system that can't back up its alerts.

Student C:

The primary difficulty during this period was a technical one related to the Gantt chart creation. Exporting our CSV-based plan into Microsoft Project (and Project 365) proved challenging. We initially faced significant import errors, including resource over-allocation warnings (due to assigning resources to summary tasks) and a failure to map Predecessors and WBS fields.

This required multiple iterations of the CSV file. We resolved it by creating a purpose-built file with a dedicated WBS column, a Duration column (instead of Start/Finish), and a Task Mode column set to "Auto-Scheduled." This new format allowed the project management software to import and render the full Gantt chart and timeline correctly.

Student D:

The primary challenge during this reporting period involved translating complex evaluation concepts into concise, clear documentation for the final report. Describing advanced testing procedures such as calibration and cross-validation in an accessible yet technically accurate manner required several revisions. Additionally, integrating the calibration workflow into the team's broader implementation plan proved difficult, as it needed to align precisely with the outputs of Student B's model and Student C's deployment framework. I addressed these issues through collaborative reviews and by refining the testing sequence to ensure compatibility across all subsystems.

4. **Tasks to Be Completed in the Next Reporting Period** (Outline the tasks to be completed in the next reporting period. Please note this should match with your milestones submitted to your FLC in Week 3; however, in consultation with (and approval of) your FLC, you can modify this to accommodate incomplete tasks from the previous period. Here, you should also identify the Project Manager for the next period.

Student A:

In the next reporting period, I will finalize revisions to both the Data and Methodology sections based on group feedback and assist in integrating all completed sections into the final 70A report. I will also help prepare the 70A presentation, particularly the slides covering dataset selection, preprocessing, and the methodological workflow. Finally, to support an efficient transition into COE/ELE70B, I will package and prepare the preprocessing scripts and unit-test outlines to ensure the implementation phase begins with a thoroughly validated and reproducible foundation.

Student B:

In the final report period, I will help to compile the final report and contribute with my AI model. I will assist in following the specified format and help to prepare my team to present this report, giving them a full rundown of the AI model and design. In collaboration with the other team members, we will be able to use this final compiled report as a springboard for starting the project in the winter term.

Student C:

In the next reporting period, I will complete the final revisions to the System Architecture and Deployment Plan sections based on team and supervisor feedback. I will assist in merging all finalized content into the complete 70A report to ensure technical and visual consistency across all sections. I will also prepare presentation slides explaining the architecture design, cloud deployment workflow, and scalability plan for the oral exam. Finally, I will validate the infrastructure configuration templates and deployment steps so that the system is ready for implementation at the start of COE/ELE 70B.

Student D:

In the next reporting period, I will finalize revisions to the Testing and Evaluation section of the 70A report and ensure that all metrics and validation methods are accurately reflected. I will also help prepare presentation slides covering evaluation metrics, validation strategies, and risk calibration for the oral exam. To support a smooth transition into COE/ELE 70B, I will begin organizing the evaluation scripts, calibration checks, and testing templates that will be used during the implementation phase.

Bibliography (APA 7)

Baldini Soares, L., FitzGerald, N., Ling, J., & Kwiatkowski, T. (2019). Matching the Blanks: Distributional Similarity for Relation Learning. arXiv preprint arXiv:1906.03158. https://doi.org/10.48550/arXiv.1906.03158

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv preprint arXiv:1810.04805. https://doi.org/10.48550/arXiv.1810.04805

Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., Naumann, T., Gao, J., & Poon, H. (2G). Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. arXiv preprint arXiv:2007.15779. https://doi.org/10.48550/arXiv.2007.15779

Hendrycks, D., & Gimpel, K. (2016). Gaussian Error Linear Units (GELUs). arXiv preprint arXiv:1606.08415. https://doi.org/10.48550/arXiv.1606.08415

Loshchilov, I., & Hutter, F. (2019). Decoupled Weight Decay Regularization. arXiv preprint arXiv:1711.05101. https://doi.org/10.48550/arXiv.1711.05101

(Anonymous, n.d.). How does the training time of BERT models compare on T4 and A100 GPUs? Massed Compute. Retrieved October 31, 2025, from https://massedcompute.com/faq-answers/?question=How%20does%20the%20training%20time%20of%20BERT%20models%20compare%20on%20T4%20and%20A100%20GPUs?

(GCPInstances, n.d.). GCP e2-medium. Retrieved October 31, 2025, from https://gcpinstances.doit.com/

(Google, n.d.-a). Free cloud features. Google Cloud. Retrieved October 31, 2025, from https://cloud.google.com/free/docs/free-cloud-features

(Google, n.d.-b). GPU pricing. Google Cloud. Retrieved October 31, 2025, from https://cloud.google.com/compute/gpus-pricing

(Google, n.d.-c). Google Kubernetes Engine (GKE) pricing. Google Cloud. Retrieved October 31, 2025, from https://cloud.google.com/kubernetes-engine/pricing

(Google, n.d.-f). Storage pricing. Google Cloud. Retrieved October 31, 2025, from https://cloud.google.com/storage/pricing

(NetApp, 2020, July 30). Google Cloud SQL pricing and limits: A cheat sheet. Retrieved October 31, 2025, from https://www.netapp.com/blog/gcp-cvo-blg-google-cloud-sql-pricing-and-limits-a-cheat-sheet/

DigitalOcean. (2025, February 5). *K-Fold Cross-Validation: The ultimate guide to robust model evaluation*. DigitalOcean Community. Retrieved October 31, 2025, from https://www.digitalocean.com/community/tutorials/k-fold-cross-validation-python

Fu, S., Shen, F., Chen, Y., Wen, A., Liu, S., Li, D.,... & Wang, L. (2024). A taxonomy for advancing systematic error analysis in multi-site electronic health record-based clinical concept extraction. *Journal of the American Medical Informatics Association*, ocae101. https://doi.org/10.1093/jamia/ocae101

Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On calibration of modern neural networks. *Proceedings of the 34th International Conference on Machine Learning, 70*, 1321–1330.

Brownlee, J. (2020, August 17). *5 effective ways to handle imbalanced data in machine learning*. Machine Learning Mastery. https://machinelearningmastery.com/5-effective-ways-to-handle-imbalanced-data-in-machine-learning/

Fu, S., Chen, Y., Hogan, W. R., He, J., Liu, S., Shen, F., Hanna, J., Kesterson, J., Zheng, T., Manion, F. J., & Wang, L. (2023). An end-to-end framework for named entity recognition and relation extraction for clinical trial eligibility criteria. *Frontiers in Neuroscience*, *17*. https://doi.org/10.3389/fnins.2023.1266771

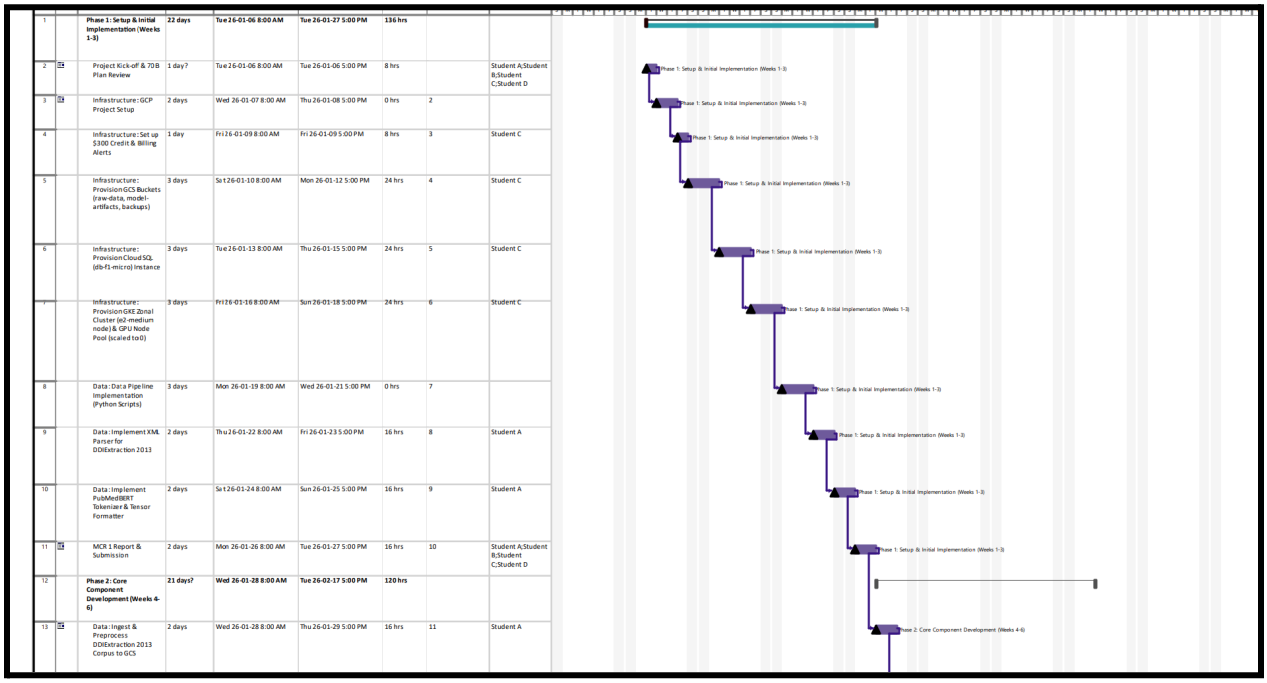
Google. (n.d.). *Datasets:* Class-imbalanced *datasets*. Machine Learning Crash Course. Retrieved October 31, 2025, from https://developers.google.com/machine-learning/crash-course/overfitting/imbalanced-datasets

Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., Naumann, T., Gao, J., & Poon, H. (2020). Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. *arXiv preprint arXiv:2007.15779*. https://doi.org/10.48550/arXiv.2007.15779

Herrero-Zazo, M., Segura-Bedmar, I., Martínez, P., & De la Peña, V. (2013). The DDI task: Extraction of drug-drug interactions from biomedical texts. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval 2013)* (pp. 375–385). Association for Computational Linguistics. https://aclanthology.org/S13-2062

Kim, S., Liu, H., Yeganova, L., & Wilbur, W. J. (2015). Extracting drug-drug interactions from literature using a rich feature-based linear kernel approach. Journal of Biomedical Research, 29(3), 206–214. https://doi.org/10.7555/jbr.29.20140121

Appendix: Design Documents



		Task Name	Duration	Start	Finish	Work	Predecessors	Resource Names
1		**Phase 1: Setup & Initial Implementation (Weeks 1-3)**	**22 days**	**Tue 26-01-06 8:00 AM**	**Tue 26-01-27 5:00 PM**	**136 hrs**		
2		Project Kick-off & 70B Plan Review	1 day?	Tue 26-01-06 8:00 AM	Tue 26-01-06 5:00 PM	8 hrs		Student A;Student B;Student C;Student D
3		Infrastructure: GCP Project Setup	2 days	Wed 26-01-07 8:00 AM	Thu 26-01-08 5:00 PM	0 hrs	2	
4		Infrastructure: Set up $300 Credit & Billing Alerts	1 day	Fri 26-01-09 8:00 AM	Fri 26-01-09 5:00 PM	8 hrs	3	Student C
5		Infrastructure: Provision GCS Buckets (raw-data, model-artifacts, backups)	3 days	Sat 26-01-10 8:00 AM	Mon 26-01-12 5:00 PM	24 hrs	4	Student C
6		Infrastructure: Provision Cloud SQL (db-f1-micro) Instance	3 days	Tue 26-01-13 8:00 AM	Thu 26-01-15 5:00 PM	24 hrs	5	Student C
7		Infrastructure: Provision GKE Zonal Cluster (e2-medium nodes) & GPU Node Pool (scaled to 0)	3 days	Fri 26-01-16 8:00 AM	Sun 26-01-18 5:00 PM	24 hrs	6	Student C
8		Data: Data Pipeline Implementation (Python Scripts)	3 days	Mon 26-01-19 8:00 AM	Wed 26-01-21 5:00 PM	0 hrs	7	
9		Data: Implement XML Parser for DDIExtraction 2013	2 days	Thu 26-01-22 8:00 AM	Fri 26-01-23 5:00 PM	16 hrs	8	Student A
10		Data: Implement PubMedBERT Tokenizer & Tensor Formatter	2 days	Sat 26-01-24 8:00 AM	Sun 26-01-25 5:00 PM	16 hrs	9	Student A
11		MCR 1 Report & Submission	2 days	Mon 26-01-26 8:00 AM	Tue 26-01-27 5:00 PM	16 hrs	10	Student A;Student B;Student C;Student D
12		**Phase 2: Core Component Development (Weeks 4-6)**	**21 days?**	**Wed 26-01-28 8:00 AM**	**Tue 26-02-17 5:00 PM**	**120 hrs**		
13		Data: Ingest & Preprocess DDIExtraction 2013 Corpus to GCS	2 days	Wed 26-01-28 8:00 AM	Thu 26-01-29 5:00 PM	16 hrs	11	Student A