AI211 Project Implementation Rubrics

Class	Rubric	Evaluation Criterion Excellent (100%)	Satisfactory (50%)	Poor (0%)	Comments (If any)
Implementation (6 points)	Setup & Use in AI & Robotics Lab (1 point)	The experimental setup including some part of training and testing was done in AI & Robotics lab using the GPUs for training, where applicable.	The training was done on Google Colab but the inference and some other minion tasks are performed in AI & Robotics Lab, where applicable.	AI & Robotics lab was not used for either training or testing of the model, where applicable.	For projects not requiring training, these marks will be adjusted in Code Readability and Replication
	Code Readability and Replication (1 point)	All code is well-documented, clear, and easy to follow. Code is well-organized and follows best practices for coding style. The code runs without errors and produces expected results. The code is easily reusable and can be adapted for other projects. The code is optimized for performance and runs efficiently. The code is fully automated, allowing for easy replication of the results.	Most code is documented and clear, but there may be some parts that are difficult to follow. Code is mostly well-organized and follows some best practices for coding style. The code runs without major errors, but may require some troubleshooting. The code is somewhat reusable, but may require some modification for use in other projects. The code runs reasonably efficiently, but may not be fully optimized. The code is somewhat automated, but may require some manual intervention.	Code is poorly documented and difficult to follow. Code is disorganized and does not follow best practices for coding style. The code does not run or produces errors that are difficult to resolve. The code is not reusable and cannot be adapted for other projects. The code runs very slowly or inefficiently. The code requires significant manual intervention and is not automated.	
	Completeness of solution (1 point)	The code is well-documented, with clear explanations of its purpose, dependencies, inputs, and outputs. The source files are submitted as ipynb files with compiled results in the same directory as the report in a Codes folder. All necessary modules and dependencies are included and properly installed. The code is free of errors and runs smoothly on all expected inputs. The code is efficient and well-optimized, with minimal redundancy and no unnecessary operations. The code includes appropriate tests and error handling to ensure proper functionality.	The code includes basic documentation, but could be more detailed or clear in its explanations. The source files are submitted as ipynb files with compiled results in the same directory as the report in a Codes folder. All necessary modules and dependencies are included, but may require manual installation or setup. The code runs without errors on most expected inputs, but may encounter issues with unusual or edge cases. The code is reasonably efficient, but could be more streamlined or optimized. The code includes some tests and error handling, but may be incomplete or insufficient.	The code is poorly documented or completely undocumented, making it difficult to understand its purpose or functionality. The source files are submitted as ipynb files with compiled results in the same directory as the report in a Codes folder. Required modules or dependencies are missing, making it impossible to run the code without additional setup or installation. The code contains significant errors or bugs that prevent it from running properly. The code is highly inefficient or redundant, with excessive computations or redundant operations. The code does not include any tests or error handling, making it highly susceptible to errors or crashes.	
	User interface design (2 points)	The GUI is intuitive and easy to use, with clear labels and instructions. The GUI is visually appealing, with a well-designed layout and appropriate use of color and graphics. The GUI provides all necessary functionalities, and allows users to easily navigate through different features and options. The GUI includes error messages and prompts that help prevent mistakes and guide users towards correct usage. The GUI is responsive and runs smoothly without any lag or crashes.	The GUI is generally easy to use, but may require some trial and error or exploration to find all necessary functionalities. The GUI has an adequate layout and design, but could benefit from some improvement in terms of visual appeal or organization. The GUI provides most necessary functionalities, but may be missing some advanced features or customization options. The GUI includes some error messages and prompts, but could benefit from more thorough guidance for users. The GUI runs mostly smoothly, but may experience occasional lag or minor glitches.	The GUI is difficult to use or navigate, with confusing labels or instructions. The GUI is visually unappealing, with a cluttered or disorganized layout and poor use of color and graphics. The GUI is missing important functionalities or options, or is unable to perform necessary tasks. The GUI lacks error messages or prompts, which can lead to mistakes or user frustration. The GUI experiences frequent lag or crashes, making it difficult or impossible to use effectively.	
	Reference to existing code and possible improvements (1 point)	Inside the code files, the author has clearly mentioned all modules/places where a part is taken from an already existing code. The author commented on changes made in the existing code which bring improvement in his/her results, with the older setting mentioned in comments.	Inside the code files, the author does not mention all modules/places where a part is taken from an already existing code. The author commented on changes made in the existing code which bring improvement in his/her results, with the older setting not mentioned in comments.	Inside the code files, the author does not mention at all where a part is taken from an already existing code. The author makes no comments on changes made in the existing code which bring improvement in his/her results.	