

specifications: 1. Data Management and Compliance: - Ability to import past student submissions and corresponding marking records in bulk. - Anonymisation of student identifiers to comply with UNSW data governance policies. - Secure storage of datasets and models in a UNSW-approved hosting environment. 2. Marking Guideline Generation - Use coordinator-marked exemplar assessments and prior-year marking practices as training data. - Automatically generate structured marking guidelines in natural language, highlighting key criteria, performance levels, and exemplar feedback. - Provide exportable marking guideline summaries for use as demonstrator training materials (e.g., PDF, HTML). 3. Consistency Benchmarking - Benchmark demonstrator marking against coordinator standards. - Identify patterns of leniency, harshness, and inconsistency through statistical and AI-based comparisons. - Visualise benchmarking results with clear dashboards (e.g., consistency scores, variance charts). 4. Moderation Reporting - Generate AI-driven moderation reports that flag discrepancies between markers. - Reports should include evidence-based recommendations (e.g., "Marker A tends to under-mark relative to coordinator standards by 12%"). - Allow coordinators to download and archive moderation reports for recordkeeping. 5. System Integration and Usability - Web-based interface accessible via UNSW credentials (Single Sign-On preferred). - Modular design to allow integration with UNSW-supported platforms (e.g., Moodle, Turnitin). - User-friendly dashboard for coordinators to upload datasets, view reports, and manage outputs. - Dedicated view for demonstrators to access AI-generated marking guidelines and training resources. 6. Evaluation and Feedback Collection - In-built survey tool for demonstrators and coordinators to provide usability feedback. - Analytics module to measure variance reduction across markers before and after system use. 7. Security and Governance - Compliance with UNSW data privacy and hosting standards. - Access controls ensuring only course coordinators and approved staff can view moderation reports. - Audit trail documenting all system activities, including dataset uploads and report generation.

### Required Skills

Students working on this project should have: 1. Programming Skills: Good knowledge of Python for AI/ML tasks and some experience with web development. 2. AI/ML Basics: Understanding of machine learning, especially natural language processing (NLP) and working with large language models. 3. Data Handling: Ability to clean, preprocess, and anonymise data, with awareness of privacy and ethical issues. 4. Databases and Backend: Basic skills in storing and managing structured data securely. 5. UI/Visualisation: Ability to build simple dashboards and reports and use visualisation tools. 6. Testing & Evaluation: Skills in testing software and gathering feedback from users.

### Expected Outcomes

By the end of the project, students are expected to deliver: 1. Source Code: Fully functional code for the AI-assisted moderation system, including data processing, guideline generation, benchmarking, and reporting features. 2. Working Prototype: A web-based application hosted in a secure environment, with coordinator and demonstrator views. 3. Sample Datasets (Anonymised): Example input and output files showing how the system processes assessments and produces reports. 4. System Documentation: Technical documentation