AAIE (Artificial Assessment Intelligence for Educator) Platform Use Case Specification

Version: 1.2

Scope: This document defines the core use cases for the AAIE platform, focused on educator and system admin interactions for prompt feedback, AI usage detection, rubric alignment, and reporting.

Actor Descriptions – AAIE Platform

Phase	Actor Name	Type	Description
Phase 1	**Educator**	Primary User	An instructor or academic staff member who uploads prompts and student responses for evaluation. Educators are responsible for selecting rubrics, reviewing AI-generated feedback, and downloading evaluation reports. They do **not** have access to model configuration or synthetic data generation tools.
	System Admin / Developer	Technical Actor	A backend administrator or developer responsible for managing the system's infrastructure, deploying model APIs, configuring workflows, and generating synthetic datasets. They ensure system functionality, integrate model updates, and manage GitHub workflows.
	LLM Model (External System)	Support System	This actor refers to the connected language model (e.g., GPT or Claude) that receives prompt and response data and returns feedback, rubric alignment, similarity scores, and hallucination detection. It operates via API and is monitored by the development team.
Phase 2	**Student** *(Reference Only)*	Future User	While not implemented in the current sprint, this actor is considered for design purposes. A student would eventually interact with the platform to submit responses, receive feedback, and track improvements via Ontrack. Currently, all student responses are uploaded by educators or simulated via GenAI.
	Role-Based Access Control (RBAC)	Support System	Authentication Layer: RBAC is enforced right after login. The system checks the user's role before granting access to any function. Frontend Restrictions: Users see only the UI components allowed by their role. Backend Gatekeeping: API endpoints enforce RBAC rules, ensuring unauthorized actions (e.g., a student accessing educator tools) are blocked. Audit Logging: All role-based access attempts are logged for transparency and security auditing.
	Prompt Owner / Researcher	Optional	This actor may refer to a project stakeholder interested in analyzing evaluation results or performing academic integrity analysis. Their role overlaps with educator but may include additional permissions for accessing analytics or summary reports.
	Ontrack *(System Component)*	System (Internal) Actor	Ontrack is the current platform used to monitor student progress. It supports student interaction for assignment submission and receiving feedback. While future integration plans include deeper student interactivity and learning analytics, view AI/LLM-generated feedback, and assign rubric scores.

Phase 1 & 2	**GitHub Workflow / Repository** *(System Component)*	Passive Actor	The GitHub repository and pull request workflows play a crucial role in the development pipeline, especially for managing documentation updates, reviewing diagrams, and tracking team contributions. While not a person, it functions as a system actor in the technical flow.

Phase 1:

Use Case 1: Submit Prompt for Evaluation

Actor: Educator
Goal: Submit a prompt and student response to receive structured feedback and analysis.

Main Flow:

1. Educator logs in via AAIE portal.
2. Navigates to "Submit Prompt" section.
3. Uploads or writes prompt and associated student response.
4. Selects evaluation rubric (or uploads a new one).
5. Submits form.

System Response:

- Saves submission.
- Sends data to LLM for feedback generation.
- Invokes AI usage detection.
- Aligns result with rubric.

Postcondition: Prompt submission is processed and queued for evaluation.

Alternative Flows:

- If rubric not selected → prompt error.
- If LLM API unavailable → show retry message.

Use Case 2: View Evaluation Results

Actor: Educator
Goal: View feedback, rubric scores, and similarity/AI usage detection.

Main Flow:

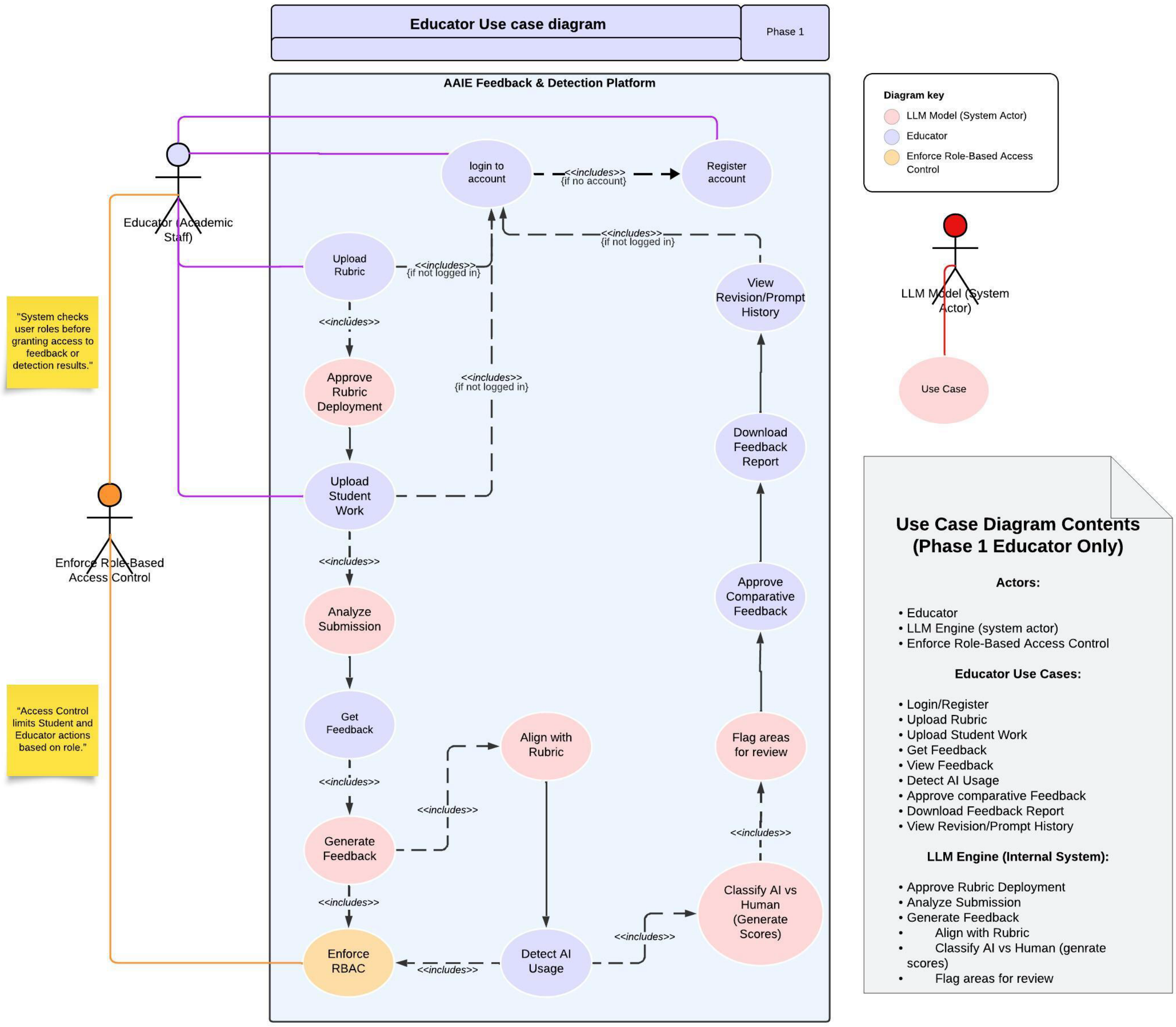
1. Educator logs in and navigates to "My Submissions."
2. Selects a previously submitted prompt.
3. Views AI-generated feedback aligned with rubric.
4. Checks AI usage detection results and similarity match.
5. Downloads full evaluation report.

System Response:

- Displays scores and breakdowns.
- Renders downloadable report.

Postcondition: Educator reviews feedback for decision-making.

Use Case Document – Educator Interaction with AAIE Platform (Phase 1)



Educator Use case diagram
Phase 1
AAIE Feedback & Detection Platform
Educator (Academic Staff)
login to account
<<includes>> {if no account}
Register account
<<includes>> {if not logged in}
Upload Rubric
<<includes>> {if not logged in}
View Revision/Prompt History
<<includes>>
"System checks user roles before granting access to feedback or detection results."
Approve Rubric Deployment
<<includes>> {if not logged in}
Download Feedback Report
Upload Student Work
Enforce Role-Based Access Control
<<includes>>
Approve Comparative Feedback
Analyze Submission
"Access Control limits Student and Educator actions based on role."
Get Feedback
Align with Rubric
Flag areas for review
<<includes>>
<<includes>>
<<includes>>
Generate Feedback
Classify AI vs Human (Generate Scores)
<<includes>>
<<includes>>
Enforce RBAC
<<includes>>
Detect AI Usage
Diagram key
LLM Model (System Actor)
Educator
Enforce Role-Based Access Control
LLM Model (System Actor)
Use Case

Use Case Diagram Contents (Phase 1 Educator Only)

Actors:

- Educator
- LLM Engine (system actor)
- Enforce Role-Based Access Control

Educator Use Cases:

- Login/Register
- Upload Rubric
- Upload Student Work
- Get Feedback
- View Feedback
- Detect AI Usage
- Approve comparative Feedback
- Download Feedback Report
- View Revision/Prompt History

LLM Engine (Internal System):

- Approve Rubric Deployment
- Analyze Submission
- Generate Feedback
- Align with Rubric
- Classify AI vs Human (genrate scores)
- Flag areas for review

Purpose:

This document outlines the functional interactions between the **Educator**, the **LLM Engine (System Actor)**, and the **Access Control Component** (May be transfer to phase 2) within the AAIE system. It describes user actions (use cases), internal system processes, and the role-based access enforcement.

Actors:

1. Educator (Academic Staff)

The main human user. Responsible for uploading student work, initiating feedback processes, and reviewing outputs.

- Triggers most use cases.
- Must be authenticated (via login).
- Has specific permissions based on their role.

2. LLM Engine (System Actor)

Represents the backend AI component responsible for:

- Detecting AI-generated content.
- Generating feedback.
- Classifying human vs AI writing.
- Aligning feedback with rubrics.

3. Role-Based Access Control (RBAC) → Transferred to phase 2

A system-enforced actor/module ensuring:

- Only permitted users access certain features.
- Students and educators have clearly segmented roles.
- Security and compliance with data access policies.

Use Cases (Educator Perspective):

Use Case	Description
Register Account	If the educator is new to the system, they must register.
Login to Account	Authenticates user before allowing access to any platform functionality.
Upload Rubric	Educator uploads marking rubric to align system feedback.
Approve Rubric Deployment *(LLM)*	Backend reviews the rubric and sets it as active for evaluations.
Upload Student Work	Allows educators to upload student assignments for evaluation.
Detect AI Usage	Starts an analysis for AI-generated content detection.
Analyze Submission *(LLM)*	The system scans the file, segments content, and checks against AI patterns.
Classify AI vs Human *(LLM)*	System generates a confidence score comparing human and AI features.
Generate Feedback *(LLM)*	The LLM generates formative feedback for the submission.

Align with Rubric *(LLM)*	Ensures generated feedback corresponds to educator-provided rubric.
Flag Areas for Review *(LLM)*	Highlights segments that may need manual checking.
Approve Comparative Feedback	Educator reviews and confirms the system's feedback.
Download Feedback Report	Educator can download the structured report for archival or sharing.
View Revision/Prompt History	Access logs of past submissions, prompts, and revision actions.
Enforce Role-Based Access Control (RBAC) *(System)* → *transferred to phase 2*	System checks role before granting access or triggering logic.

Notes:

- This use case diagram is **Educator-focused (Phase 1)**. Future phases (e.g., Student or Admin views) will have additional actors and interactions such as Enforce Role-Based Access Control (RBAC).
- All actions are dependent on the user's role and prior steps (e.g., login must precede all actions).
- The LLM Engine is a **passive system actor**; it performs logic based on triggers from the Educator interface.

Key Roles and Their Permissions:

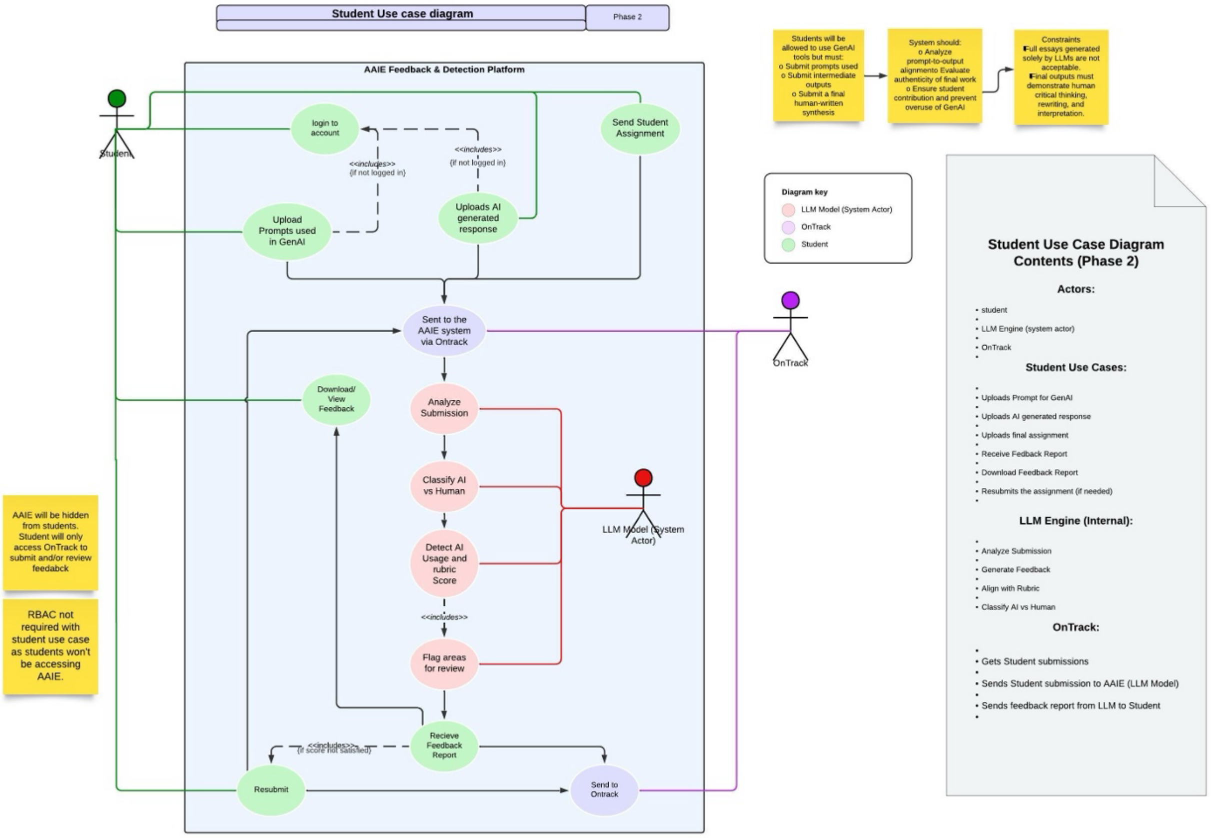
Role	Permissions & Access
Educator	- Upload and evaluate student submissions - Submit rubrics and generate feedback - Access feedback reports - View and audit past prompt history
Admin / System Developer	- Manage user roles and permissions - Deploy and update APIs - Monitor system operations and logs

Related Artifacts

System Design and Context Diagram
Requirement Analysis and Documentation
Prompt Evaluation and Response Enhancement
Backend Logic Design and Class Diagram
Use Case and Persona Documentation (Future Scope)
API & Technical Alignment (Define Core API Calls)
Frontend Flow and Activity Diagram
Database Design (Domain Model)
Identify Datatypes, Input Methods, and Validation Documentation

Phase 2:

Use Case Document – Student Interaction with AAIE Platform (Phase 2)



Students will be allowed to use GenAI tools but must:
o Submit prompts used
o Submit intermediate outputs
o Submit a final human-written synthesis

System should:
o Analyze prompt-to-output alignmento Evaluate authenticity of final work
o Ensure student contribution and prevent overuse of GenAI

Constraints
Full essays generated solely by LLMs are not acceptable.
Final outputs must demonstrate human critical thinking, rewriting, and interpretation.

AAIE will be hidden from students. Student will only access OnTrack to submit and/or review feedabck

RBAC not required with student use case as students won't be accessing AAIE.

Student Use Case Diagram Contents (Phase 2)

Actors:

- student
- LLM Engine (system actor)
- OnTrack

Student Use Cases:

- Uploads Prompt for GenAI
- Uploads AI generated response
- Uploads final assignment
- Receive Fedback Report
- Download Feedback Report
- Resubmits the assignment (if needed)

LLM Engine (Internal):

- Analyze Submission
- Generate Feedback
- Align with Rubric
- Classify AI vs Human

OnTrack:

- Gets Student submissions
- Sends Student submission to AAIE (LLM Model)
- Sends feedback report from LLM to Student

Focus: Captures the flow of how students interact indirectly with AAIE via the OnTrack system, enabling prompt analysis, feedback generation, and AI detection.

Purpose

This use case illustrates how students engage in academic work using GenAI tools and OnTrack, and how the backend AAIE system supports analysis, feedback, and evaluation. It emphasizes system automation, AI detection, and student transparency, without providing students direct access to the AAIE platform.

Actors

Actor	Description
Student	Uploads responses via OnTrack, uses GenAI to assist with responses, and downloads feedback. They do **not** interact directly with the AAIE backend.
OnTrack	The university’s learning management platform through which students upload assignments and view feedback. Acts as the **bridge** between students and the AAIE platform.

LLM Model (System Actor)	Responsible for analyzing submissions, classifying AI-generated content, detecting usage, and scoring rubric alignment inside the AAIE system. Operates entirely behind the scenes.

+ Phase 1 Actors

Main Use Cases (Student Perspective):

1. Login to Account

If not logged in, redirect to authentication.

Students access their GenAI or OnTrack account to start the academic task workflow.

2. Upload Prompts Used in GenAI

Students disclose the prompts they used to generate AI responses.

This promotes transparency and allows the system to later verify AI usage and integrity.

3. Upload AI-Generated Response

Students submit their GenAI-assisted work.

Submissions are shared to OnTrack and sent into the AAIE system for analysis.

4. Send Student Assignment (via OnTrack)

OnTrack sends the student submission to the AAIE backend.

Acts as the integration point that moves submissions from front-facing platforms into the backend analysis flow.

5. Analyze Submission (System Use Case)

LLM model analyzes the input using multiple layers:

- **Classify AI vs Human**: Identifies whether the response is likely written by a human or AI.
- **Detect AI Usage and Rubric Score**: Compares the structure, clarity, and relevance against rubric criteria.
- **Flag Areas for Review**: Marks segments that need human or educator review.

6. Download / View Feedback

Students receive evaluated feedback from AAIE, routed back via OnTrack.

7. Resubmit (If Needed)

In case feedback requires revision, students may revise and resubmit the updated work for reevaluation.

System Notes & Decisions

- **RBAC Not Required Here**: Because students do not directly access AAIE, RBAC for students is not implemented in this flow.
- **Student Privacy Maintained**: Students interact only with OnTrack, keeping backend detection hidden to avoid anxiety and maintain academic integrity.
- **Use of GenAI**: Students are allowed to use GenAI, but must declare prompts and submit a **final human-edited version**.
- **System Advice (from PO)**: System should not just evaluate for AI usage, but also include:
  - Prompt quality
  - Rubric alignment
  - Authenticity scoring

Design Considerations

- **Two-Tier Evaluation**: The platform uses GenAI detection and rubric-based scoring for multidimensional feedback.
- **Feedback Report**: Includes breakdowns by rubric area and AI-use likelihood.
- **No Student Access to AAIE**: Ensures academic integrity and enforces platform security.

Considerable use cases included in the future scope:

Deploy/Update LLM Model

Actor: System Admin / Developer
Goal: Update the backend model used for prompt evaluation.

Main Flow:

1. Admin logs into system dashboard.
2. Selects "Model Deployment" section.
3. Uploads or configures API endpoints.
4. Runs test prompt and evaluates response.
5. Finalizes deployment.

System Response:

- Validates endpoint.
- Integrates model into feedback loop.
- Begins tracking hallucination and bias logs.

Postcondition: New model is now serving API requests.

Generate Synthetic Prompts and Responses

Actor: System Admin / Developer
Goal: Populate platform with synthetic data using GenAI.

Main Flow:

1. Admin navigates to "Data Simulation."
2. Enters desired scenario (e.g., argumentative prompt).
3. System generates synthetic student response.
4. Pairs response with rubric.
5. Submits automatically to pipeline.

System Response:

- Processes as normal student submission.
- Flags as "Synthetic" for later analysis.

Postcondition: Platform contains testable data for rubric alignment and detection.

Monitor and Evaluate Model Output

Actor: LLM / Model Team
Goal: Track feedback quality, hallucination rates, and model performance.

Main Flow:

1. Team accesses system logs or dashboards.
2. Reviews flagged responses (bias/hallucination).
3. Reviews rubric match scores.
4. Marks bad outputs for improvement.
5. Refines prompt format or LLM endpoint.

Postcondition: Model outputs iteratively improved over time.

Security and Access Notes

- Role-based access (RBAC) ensures only authorized users can submit or configure models.
- Educators cannot access model configuration panels.
- Data flagged as synthetic is excluded from final reports.

Access Control Logic:

RBAC is enforced at multiple interaction points:

- If the user is not logged in, no data submission or access is allowed.
- Students and educators have different interface views and API permissions.
- The system internally verifies access rights before exposing detection results or feedback.