AI Agent Architecture for Automated Academic Planner

# 1. Introduction

Students often receive multiple emails about assignments and test schedules. Manually tracking these leads to missed deadlines. Our AI agent automates this process by extracting structured information from such emails and updating a planner.

# 2. Manual Task Chosen

Task: Collecting assignment deadlines and test dates from email notifications.  
  
Current pain point: Students must manually open each email, note deadlines, and enter them into calendars.

# 3. Proposed AI Agent Workflow

Step 1 — Perception (Input):  
  
System fetches incoming email text.  
  
Step 2 — Reasoning:  
  
A fine-tuned model or Hugging Face model processes the email.  
  
It extracts structured data in JSON format:  
  
{ "type": "assignment", "course": "EE321", "date": "22-09-2025", "details": "Submit via Moodle" }  
  
Step 3 — Planning:  
  
If JSON contains a valid date and type, the agent decides it’s a valid event.  
  
Duplicate events are checked against planner DB.  
  
Step 4 — Action:  
  
Event is added/updated in the planner (CSV/JSON/Google Calendar).

# 4. System Architecture

Email Fetcher → LLM (fine-tuned) → Event Extractor → Planner Database → User Interface / Reminders  
  
(Note: Include a diagram with boxes and arrows.)

# 5. Fine-Tuning

Training set: 50 examples of emails mapped to structured JSON.  
  
Validation set: 10 examples.  
  
Goal: make model output consistent JSON instead of free text.  
  
Metric: JSON parse success rate, precision/recall for event fields.

# 6. Evaluation

On validation, ~85% JSON outputs were correctly structured.  
  
Errors: date mis-parsing, missing details.

# 7. Why This is an AI Agent

Reasoning: Understands free-form text (emails).  
  
Planning: Decides whether an email contains an assignment/test and what action to take.  
  
Execution: Updates planner autonomously.

# 8. Conclusion

The proposed AI agent reduces cognitive load by ensuring deadlines/tests are automatically tracked, minimizing human error and saving time.