AstralAssist Documentation

CSEN 296B, Final Project Report

Sahana Chandramohan
Anish Katragadda
Tara Khambadkone
Sarvesh Nathan
Vedaant Vyas

Section I: Project Introduction

Project Overview:

AstralAssist is a multi-agent application developed by Red Team A, designed to function as a technological virtual assistant. Its capabilities include managing emails, handling documents, and processing financial reimbursements while adhering to established HR policies.

Repository Information:

The source code and project materials for AstralAssist are maintained on GitHub and can be accessed at: https://github.com/AnishKatr/AIGovernanceProject.git

Section II: Final Team Member Responsibilities & Contributions

The Red Team comprises of 5 students, whose responsibilities & contributions take the following shape:

Sahana Chandramohan

- Responsibilities
    - Team Management (TM)
    - Architecture (A) Lead
    - System User Guide (SUG) Lead
    - Red Team Test Suite (RTTS) Development
    - MAESTRO Threat Modeling (MTM) Development
    - Governance Connection (GC) Support
    - System Audit (SA) Lead
- Contributions
    - TM: Organized & led all weekly team meetings + supplied agenda & took notes
    - A: Lead internal architecture meeting & consolidated team ideas
    - A: Designed & implemented architecture diagram in Miro
    - A: Updated architecture diagram
    - A: Developed and wrote up architecture description
    - SUG: Led internal system user guide meeting & consolidated team ideas
    - SUG: Designed & implemented system user guide
    - RTTS: Developed test suite outline for teammates
    - RTTS: Researched & filled out outline for Agency & Reasoning Threats
    - RTTS: Researched & filled out outline for Multi-Agent System Threats
    - MTM: Developed threat modeling suite outline for teammate
    - MTM: Researched & filled out outline for Agent Frameworks
    - MTM: Researched & filled out outline for Deployment & Infrastructure

- MTM: Researched & filled out outline for Evaluation & Observability
- MTM: Researched & filled out outline for Agent Ecosystem
- GC: Researched & filled out Risk Register for R-05
- GC: Researched & filled out Risk Register for R-06
- GC: Researched & filled out Risk Register for R-07
- GC: Researched & filled out Risk Register for R-08
- GC: Researched & filled out Risk Register for R-09
- GC: Researched & filled out Risk Register for R-10
- GC: Researched and developed IRP for 5 threats
- SA: Developed outline for NIST AI RMF application for teammates
- SA: Researched & filled out outline for Manage step in NIST AI RMF
- TM: Formatted, developed and wrote final report
- TD: Testing of completed application

Anish Katragadda

- Responsibilities:
  - Technical Development (TD) Lead
  - Architecture Support I
  - Red Team Test Suite Development
  - System Audit
- Contributions:
  - A: Co-lead Architecture design
  - A: Researched architecture pipeline
  - A: Designed technical stack in order to deploy full stack RAG application
  - RTTS: Researched and filled outline for **Memory Attack Surface**
  - TD: Outlined Tech Stack to be Next.JS Frontend, Flask Backend, LLM's through GROQ API, along with Pinecone for Vector database applications
  - TD: Researched Slack API and Integration
  - TD: Vector DB Integration
  - TD: Designed and implemented LLM Agent
  - TD: Scaled Agent with fully integrated RAG Pipeline
  - TD: Created CI/CD development pipeline
  - TD: Frontend fully live hosted and deployed for public use
  - TD: Custom HR API fully hosted and deployed
  - TD: Custom Drive API fully hosted and deployed
  - TD: Custom Email API fully hosted and deployed
  - TD: Integrated all API's and front end in order to achieve MVP
  - TD: Bug tested feature functionality (QA) ensuring fully intended architecture functionality is available

Tara Khambadkone

- Responsibilities:
    - Technical Development (TD) Support II
    - System User Guide Support I
    - Red Team Test Suite Development
    - Governance Connection Lead
    - System Audit
- Contributions:
    - RTTS: Research & filled out outline for Authentication & Identity
    - MTM: Research & filled out outline for Foundation Models
    - MTM: Research & filled out outline for Data Operations
    - MTM: Research & filled out outline for Security & Compliance
    - TD: Researched and took notes on proposed tech stack (Flask + Next.js)
    - TD: Researched and took notes on APIs required for expense management system/HR system
    - TD: Implemented custom HR API using FastAPI
    - TD: Collaborated with dev team to strategize HR API integration with other APIs for expense management system
    - SA: Researched & filled out outline for Govern step in NIST AI RMF
    - GC: Developed outlines for risk register and incident response plan for governance team to fill out
    - GC: Researched & filled out Risk Register for R-01
    - GC: Researched & filled out Risk Register for R-02
    - GC: Researched & filled out Risk Register for R-03
    - GC: Researched & filled out Risk Register for R-04
    - GC: Researched & filled out Risk Register for R-05
    - GC: Researched and developed IRP for 5 threats

Sarvesh Nathan

- Responsibilities:
    - Technical Development (TD) Support I
    - Architecture Support (AS) I
    - Red Team Test Suite Development
    - System Audit
- Contributions:
    - AS: Helped in designing & provided feedback for architecture diagram
    - AS: Provided ideas on how we could add RESTful endpoints to the Technical Architecture diagram
    - RTTS: Researched & filled out outline for Human-in-the-loop Threats

- RTTS: Aggregated the points in the outline into a paragraph and reinforced the importance of HITL using research articles
- TD: Researched & filled out outline for integration of Multi-Agents with Google API
- TD: Researched RAG pipeline and how it can be used to integrate Google API with Expense and Drive Agents
- SA:Researched and filled out information on the MAP section of NIST AI RMF
- TD: Created the Google Drive API script, including all the features such as being able to search and filter options, and downloading and saving files.
- TD: Created a spreadsheet csv template used to store employee data in HR/Drive integration
- TD: Did HR/Drive integration, where you can get the employee database from the HR API and then it gets saved as a csv file in google drive.
- TD: Added an update feature in the integration file which shows the most recent csv file every time the HR file generates a new database of new employees every time it is run, and once it is saved as a csv file, it replaces the previous file with the more recent update.

Vedaant Vyas

- Responsibilities:
  - Technical Development (TD) Support I
  - System User Guide (SUG) Support I
  - Red Team Test Suite (RTTS) Development
  - System Audit (SA)
- Contributions:
  - TD: Researched and took notes on how email client can be integrated with Astral Assist
  - TD: Researched on using Google API suite for sending emails.
  - TD: Researched NEXT.js and Flask for bridge between front and back end development.
  - RTTS: Researched and filled out outline for Tool Based & Execution Threats.
  - RTTS: Summarized Tool Based & Execution Threats in the form of a paragraph and outline.
  - SUG: Provided preliminary ideas on what the System User Guide can contain.
  - TD: Designed and implemented the complete Gmail API integration pipeline for Astral Assist, including authentication flow, credential management, and secure token handling, and logging
  - TD: Developed a Python-based email generation and sending module capable of constructing messages dynamically from JSON specifications.
  - TD: Designed a workflow where LLM-generated or system-generated fields (recipient, subject, body, attachments) are automatically transformed into valid outbound emails.
  - TD: Performed end-to-end testing using team accounts to validate delivery, formatting, and error-handling edge cases.
  - TD: Integrated the email automation module with outputs produced by the HR API, enabling automated construction of HR-related email templates based on personnel data. Coordinated with HR API developer for smooth workflow.

- TD: Implemented validation logic to ensure HR-provided fields populate safely and consistently into the email schema.
- SA: Completed the *MEASURE* component of the NIST Cybersecurity Framework for all six threats in the project's threat taxonomy, ensuring each threat included clear, actionable metrics and evaluation criteria.

Section III: Architecture Diagrams

The internal system design of AstralAssist is illustrated through two complementary diagrams: the technical architecture and the functional architecture, presented below in Figures 1 and 2. They detail the inner workings and relationships between the user, agents, memory entities, external applications, and important system software components.

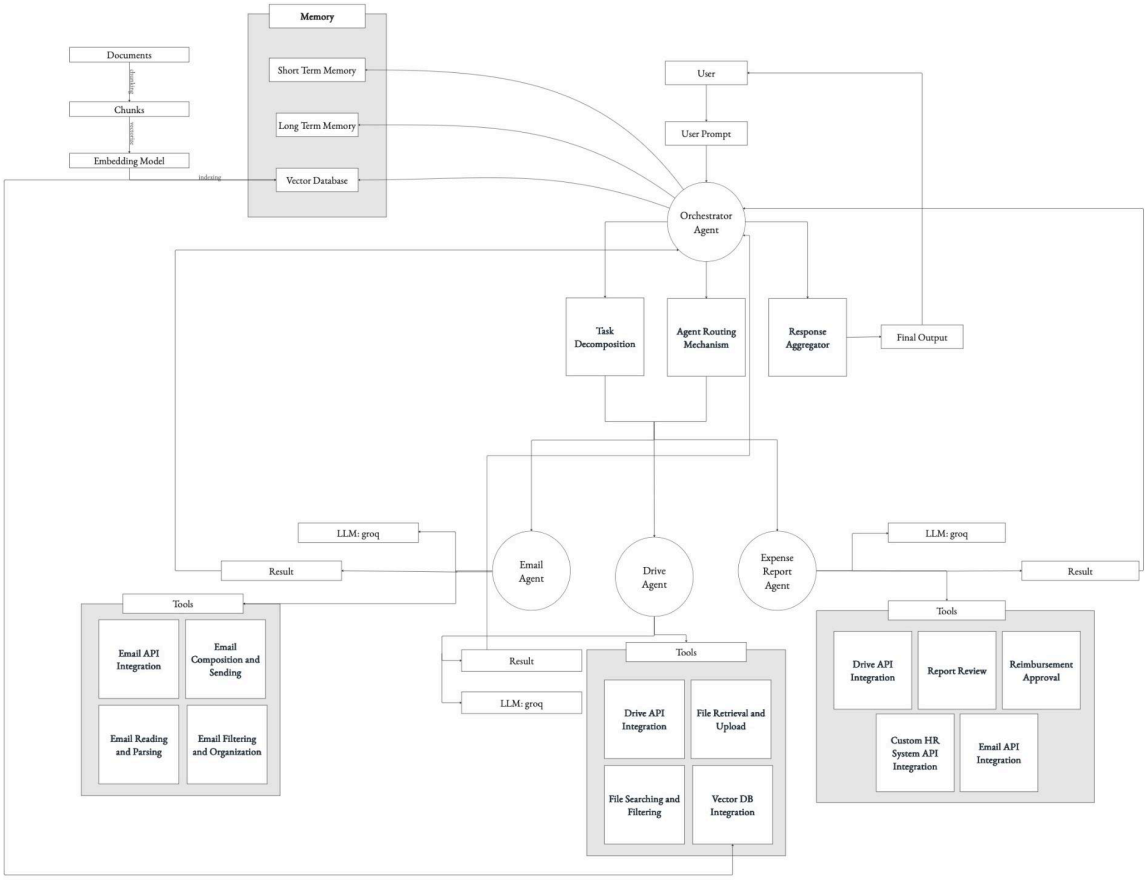

Figure 1: AstralAssist Technical Architecture

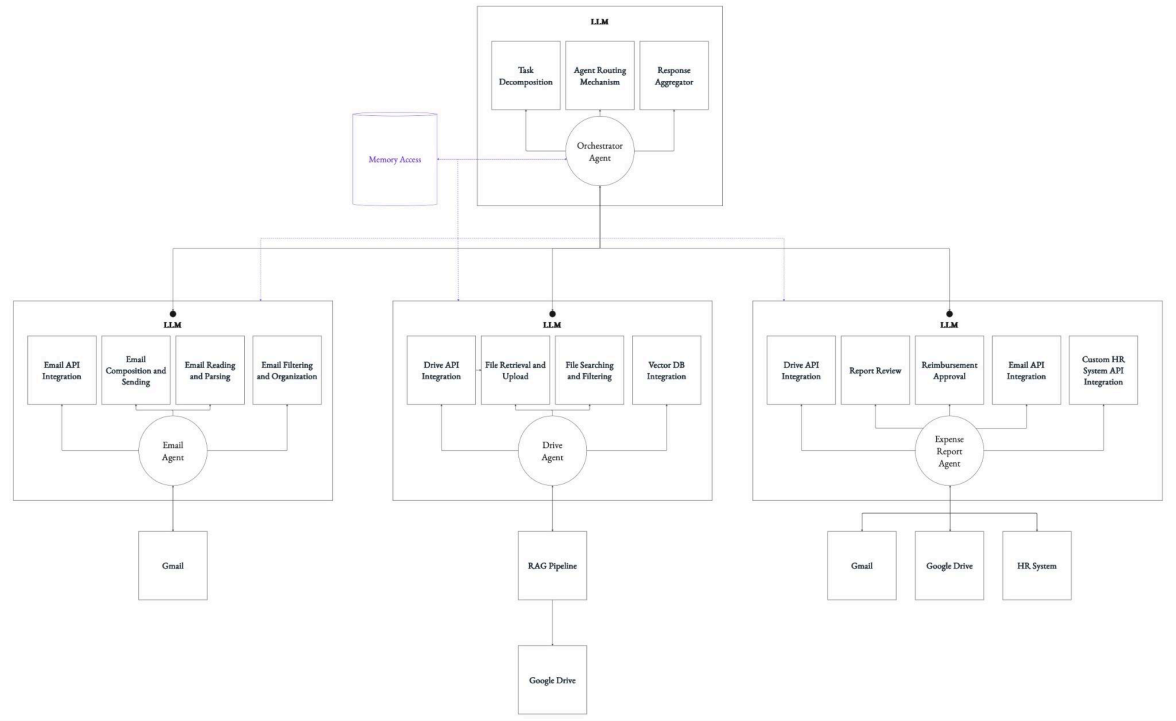

Figure 2: AstralAssist Functional Architecture

The technical and functional architecture of AstralAssist is designed to provide secure, auditable, and efficient multi-agent AI workflows. At its core is a large language model (LLM) provided by Groq, which supports task decomposition, agent routing, and response aggregation. When a user submits a prompt, the Task Decomposition component breaks it into smaller, actionable sub-tasks suitable for specialized agents. These sub-tasks are then directed to the appropriate agent through the Agent Routing Mechanism, which ensures that tasks are processed by the Email Agent, Drive Agent, or Expense Report Agent as required. The Response Aggregator collects the outputs from each agent, synthesizing them into a coherent final result. Memory access is handled through both short-term and long-term memory, enabling session-level context retention and persistent document storage, including embeddings and indexed chunks managed via a RAG pipeline and vector database. The Orchestrator Agent coordinates all workflows, ensuring proper task execution and aggregation of results.

The Email Agent manages email-related tasks, integrating with email APIs to send, receive, parse, filter, and compose messages, and to issue notifications for approvals or denials. The Drive Agent handles document management using Drive API integration, including file retrieval, upload, searching, filtering, and vector database integration for RAG-supported operations. The Expense Report Agent processes employee expense reports by leveraging Drive API integration, report review, reimbursement approval, email API integration, and a custom HR system API. This agent is responsible for updating financial records, issuing reimbursements, and notifying employees of the outcomes.

Throughout the system, memory is carefully managed: user prompts, documents, emails, and embeddings are chunked, indexed, and stored with provenance tracking to ensure security, data integrity, and auditability. All interactions between agents, the orchestrator, memory, and LLMs are monitored and logged to provide traceable decision-making. This architecture ensures that each user request is securely decomposed, routed, executed, and aggregated, resulting in outputs that maintain consistent reasoning, adhere to policy, and reflect reliable, auditable operations.

Section V: User Guide

The user guide outlines the implemented policies and request formats while providing comprehensive instructions for interacting with the application. The complete guide is presented in the following text:

ASTRALASSIST USER GUIDE

Section I: Introduction

This User Guide provides comprehensive documentation for the AI-Powered Personal Assistant System, an artificial intelligence-based platform designed to serve as your digital personal assistant for various workplace tasks, including automated expense reimbursement processing. This document is intended for all employees who interact with the personal assistant system. The guide delineates the established policies, required request formats, operational procedures, and instructions for proper system interaction.

The Personal Assistant System is a multi-agent artificial intelligence platform that can manage your emails, organize and retrieve documents from cloud storage, process expense reimbursement requests, and perform various administrative tasks on your behalf. The system operates through natural language interaction, allowing you to make requests conversationally while the system orchestrates the necessary actions across multiple integrated platforms including email systems, cloud storage services, and human resources databases. This guide should be reviewed thoroughly by all users prior to their first interaction with the system. Failure to comply with the policies and formats outlined herein may result in request rejection or processing delays.

Section II: Established Policies & Compliance Requirements

All users of the Personal Assistant System must adhere to the policies established in this section. These policies have been developed to ensure system security, protect sensitive information, maintain compliance with organizational standards, and enable efficient request processing.

Prompt and Request Submission Policies

Character limits are enforced to ensure optimal system performance and prevent processing errors. Individual prompts or requests submitted to the personal assistant are limited to prevent excessive processing demands. Users should provide concise, clear instructions that adequately explain their needs without excessive elaboration. Requests exceeding reasonable length may experience processing delays or require breaking into multiple separate requests.

External links are prohibited in prompts submitted to the personal assistant. Users must not include hyperlinks to external websites, cloud storage services not integrated with the system, or any other external resources within their requests. All materials the assistant needs to access must be stored in organizational cloud storage platforms where the system has appropriate access permissions. Requests containing external links will be rejected for security reasons.

Private and proprietary data entry restrictions serve to protect sensitive organizational information and comply with data security protocols. Users should not enter private or proprietary data into prompts beyond what is necessary for the assistant to complete the requested task. For expense reimbursement specifically, users should provide only standard expense details such as date, amount, category, vendor name, and business purpose. Users must not include confidential client information, proprietary business strategies, classified project details, or any other sensitive data unless directly necessary for the task at hand.

Code and functions are prohibited in user prompts. Users must not attempt to include executable code, scripts, system commands, or any form of programmatic instruction within their requests to the personal assistant. The assistant operates through natural language understanding and does not execute user-provided code. All requests must be expressed in plain language describing the desired outcome or action.

Human Resources System Integration Policies

When the personal assistant processes expense reimbursement requests, all requests remain subject to existing organizational policies regarding allowable expenses, spending limits, approval authorities, and reimbursement eligibility. Reimbursement threshold policies determine the processing pathway for expense requests based on total amount. Requests below five hundred dollars are processed automatically by the assistant without human intervention, provided they comply with all policy requirements. Requests between five hundred and two thousand dollars may be automatically processed by the assistant but are flagged for management review and verification. Requests exceeding two thousand dollars invariably require human-in-the-loop review by an authorized approving official before reimbursement is issued.

Human-in-the-loop review is implemented for expense requests that exceed automatic approval thresholds, violate policy parameters in ways that may warrant exception approval, involve unusual expense categories or circumstances, or are flagged by the system's anomaly detection algorithms as potentially requiring additional scrutiny. When human review is triggered, the assistant compiles all relevant information and routes it to an appropriate reviewer who exercises final decision-making authority.

Data Access, Privacy, and Consent Requirements

By using the Personal Assistant System, employees explicitly consent to the following data processing activities. The personal assistant requires read access to your email communications to perform email management functions including sending, receiving, organizing, and searching emails on your behalf. The assistant will only access emails when specifically requested to do so or when necessary to complete tasks you have assigned.

The assistant requires read and limited write access to your cloud storage accounts including Google Drive and OneDrive. This access enables the assistant to search for documents, retrieve files, organize folders, and upload documents as needed to complete your requests. The assistant does not modify or delete your documents without explicit instruction but may create new folders or move documents as part of organizational tasks you request.

For expense reimbursement functionality, the assistant requires read access to your employee profile information stored in human resources databases, including employee name, identification number, department assignment, manager identification, and contact information. The assistant also requires read access to your financial information, specifically bank account details for reimbursement processing. This highly sensitive information is accessed only when processing approved expense reimbursements to initiate electronic funds transfer. Financial information is encrypted in transit and at rest, and access is logged for security audit purposes.

The assistant has write access to create transaction records in human resources and financial systems when processing expense reimbursements. These records document submitted requests, processing actions, approval decisions, and reimbursement payments, and become part of your permanent employee file. All system interactions and data access by the assistant are logged for security and audit purposes.

Employees are responsible for ensuring that their human resources profile information, particularly bank account details, remains current and accurate. The assistant relies on the information stored in human resources systems and cannot process reimbursements if account information is outdated or incorrect.

Section III: Interacting with AstralAssist

AstralAssist is designed to understand natural language requests and execute tasks on your behalf. Unlike traditional form-based systems, you interact with the assistant conversationally, describing what you need in plain language. This section explains how to effectively communicate with your assistant for various tasks.

General Interaction Principles

When making requests to your personal assistant, clarity and specificity yield the best results. Describe what you want to accomplish, provide necessary context, and specify any relevant details such as dates, names, or file locations. The assistant can handle complex multi-step requests but may ask clarifying questions if your initial request is ambiguous. You can interact with the assistant through the web-based interface, by prompting it.

The assistant will confirm understanding of your request and notify you when tasks are completed or if any issues arise that require your attention.

Email Management Requests

Your personal assistant can send, receive, organize, and search your email communications. To send an email, you might say "Send an email to John Smith in the marketing department with the subject 'Q4 Planning Meeting' letting him know that I would like to schedule a meeting next week to discuss the product launch timeline." The assistant will compose and send the email on your behalf, confirming the action with you before sending.

To organize emails, you can request actions such as "Move all unread emails from last week about the Atlas project into a folder called Atlas Project Correspondence" or "Find all emails from Sarah Johnson in the past month and mark them as important." The assistant uses its understanding of your email structure and content to execute these organizational tasks.

To search for specific emails or information, you can make requests like "What did Michael say about the budget in his email last Tuesday?" or "Find the email containing the meeting notes from the client presentation on AI agent integration in September." The assistant will search your email, retrieve relevant messages, and provide you with the information or forward the messages to you as appropriate.

Document Management Requests

Your personal assistant can search for, retrieve, organize, and manage documents stored in your cloud storage accounts. To retrieve a document, you might request "Find the presentation file about Q3 sales results that I created last month" or "Get me the contract document for the Acme Corporation partnership." The assistant will search your cloud storage, locate the relevant document, and either provide you with the file or a link to access it.

To organize documents, you can make requests such as "Create a new folder called Client Proposals and move all PDF files from my Documents folder that contain the word 'proposal' into it" or "Organize all my expense receipts from October into a folder structure by date." The assistant will execute these organizational tasks, creating folders and moving files as specified.

Expense Reimbursement Requests

Your personal assistant can process expense reimbursement requests on your behalf, handling the entire workflow from submission through approval and payment. To submit an expense reimbursement request, provide the assistant with the necessary information in a conversational format. For example, you might say "I need to submit an expense reimbursement for a flight to Chicago on October 15th that cost $487.60. It was for a

client meeting with Acme Corporation representatives to discuss the Q4 partnership. The receipt is in my Google Drive in the folder Expense Receipts, October 2025."

The assistant will process this request by retrieving your employee profile information, locating the receipt document in your cloud storage, validating the expense against organizational policies, and submitting the reimbursement request. The assistant will confirm receipt of your request and provide you with a tracking identifier. You will receive email notifications as the request progresses through validation, approval, and payment stages. Any transactions above $500 will require human approval.

For more complex expense requests involving multiple items, you can provide the information in a natural conversational flow. For instance: "I attended a conference last week and need to submit several expenses. First, there was the conference registration fee of $895 paid to the Association for Computing Machinery on October 18th. Then I had a hotel stay at the Marriott from October 19th to October 21st that cost $650 total. I also had a business dinner with a prospective client on October 20th at The Capital Grille for $156.75. All the receipts are in my OneDrive folder called October Expenses. These were all for the AI Research Conference in San Francisco, which was approved as part of my professional development plan."

The assistant will parse this information, identify the individual expense items, retrieve all necessary receipts, and create appropriate reimbursement requests. The assistant may ask clarifying questions if any required information is missing or ambiguous, such as "I found three PDF files in your October Expenses folder. Which one is the receipt for the Capital Grille dinner?"

You do not need to format your expense requests according to rigid templates when working with the personal assistant. However, you should include certain essential information for the assistant to process the request: the date the expense was incurred, the amount, the vendor or merchant, the expense category (travel, meals, office supplies, professional development, etc.), the business purpose or justification, and the location of receipt documentation in your cloud storage. The assistant will organize this information into the proper format required by the expense reimbursement system.

Required Information for Expense Reimbursements

While the personal assistant accepts requests in natural conversational language, certain information is mandatory for expense reimbursement processing. Ensure your request includes the transaction date when the expense was incurred, the expense category selected from travel, meals and entertainment, office supplies, professional development, client-related expenses, technology and equipment, or other with specification. You must also provide the vendor or merchant name, the total amount in dollars including taxes and fees, a clear business purpose explaining why the expense was necessary for organizational business, and the location of receipt or invoice documentation in your cloud storage including the specific folder path and filename if possible. Finally, you must submit proof of approval for the incurred expense.

The assistant will prompt you for any missing information rather than rejecting your request. For example, if you say "I need to be reimbursed for a business lunch yesterday that cost about $125," the assistant will respond with questions like "What was the name of the restaurant?" and "What was the business purpose of this lunch?" and "Do you have the receipt uploaded to your cloud storage?"

Documentation Requirements

All expense reimbursement requests must be supported by appropriate documentation. For expenses exceeding twenty-five dollars, original itemized receipts are required showing the vendor name, transaction date, itemized listing of purchased goods or services, total amount including taxes, and payment method. For lodging expenses, the hotel folio showing nightly rate, dates of stay, and breakdown of charges is required. For meal expenses, the itemized restaurant receipt is required, and you should provide information about attendees and business purpose if not evident from the receipt itself.

For mileage reimbursement claims, provide origin and destination addresses, business purpose of travel, date of travel, and calculated mileage. For expenses involving client entertainment or meals, include the names and organizational affiliations of all attendees, their business relationship to the organization, and the specific business purpose or topics discussed. When making your request to the personal assistant, simply indicate where these receipts are stored in your cloud storage, and the assistant will retrieve them as part of processing your request.

All receipts and supporting documentation must be stored in your organizational cloud storage accounts (Google Drive or OneDrive) in accessible locations. Acceptable file formats include PDF, JPEG, PNG, and TIFF. Receipts should be clear, complete, and legible, as poor quality images that do not allow verification of expense details will result in request delays or rejection. Upload receipts to your cloud storage before requesting reimbursement through your personal assistant.

Section VI: Red Teaming Test Suite

The following section implements the Red Teaming Guide, applying the ASI threat taxonomy to AstralAssist. Each threat category includes definitions, relevant attack surfaces, potential issues, application objectives, mitigation measures, success criteria, and monitoring strategies.

1. Agency & Reasoning Threats

Agency & Reasoning Threats occur when AI agents reason or act beyond their intended scope. This includes misinterpretation of user intent, flawed task decomposition, and unsafe reasoning that leads to unintended or harmful actions.

Aspect	Details
Relevant Attack Surfaces	Data: Biased/poisoned/incomplete data in vector database and memory layers; sensitive user data influencing reasoning outcomes. Model: LLM hallucinations; unsafe reasoning from model decision boundaries or internal representations. System & Infrastructure: Orchestrator routing logic/API endpoints mishandling reasoning tasks; response aggregator combining unverified outputs. User Interaction: Ambiguous, manipulated, or adversarial prompts; social engineering via input manipulation. Supply Chain: Vulnerabilities in third-party LLM libraries or embedding models; unverified model updates altering reasoning behavior.
Related Issues	Hallucinated or unsafe outputs; prompt injection; cross-context leakage; excessive authority granted to agents; lack of explainability or validation; unauthorized/unintentional actions.
Application Objectives	Collects user prompts, email contents, documents, file metadata, embeddings, and memory for reasoning; LLM decision-making to route and execute user tasks.
Risks & Vulnerabilities	Unauthorized or incorrect agent actions; exposure of private user data via hallucinated reasoning; misalignment between system reasoning and user goals; privacy risks from misrepresented outputs.
Mitigation Measures	Constrain agent permissions (least privilege); implement validation checkpoints; sanitize prompts; human review for high-impact actions; audit reasoning chains for traceability.
Success Criteria	Explainability, validation, and consistency of reasoning.
Monitoring Measures	Log all reasoning steps, task routing decisions, and LLM outputs; anomaly detection on reasoning sequences; maintain model version control; trigger alerts for unsafe or unexpected agent behavior.

---

2. Memory-Based Threats

Memory-Based Threats target AstralAssist's short-term memory, long-term memory, and vector database. Malicious or sensitive content can persist, resurface in later workflows, and be treated as trusted context.

Includes memory poisoning, stored prompt injection, cross-tenant/user leakage, and misuse of retained enterprise data.

Aspect	Details
Relevant Attack Surfaces	Document ingestion pipeline; email/Drive connectors; agents' memory write paths; vector database and embedding index; memory access authorization and tenancy; orchestrator/agent workflows reading stored context.
Related Issues	Persistent prompt injection; poisoned policy or financial guidance; silent goal drift; exposure of HR or financial records to unrelated users.
Application Objectives	Supports secure, auditable automation for HR, expenses, documents, and internal Q&A. Preserves integrity, confidentiality, least privilege, and regulatory compliance.
Relevant Data	User prompts, interaction history, internal documents/files, emails, expense reports, invoices, HR records, embeddings in vector databases.
Risks & Vulnerabilities	Poisoned content instructing agents to approve vendors or loosen permissions; cross-tenant retrieval; deleted/revoked items still influencing responses.
Mitigation Measures	Enforce strict memory scoping per tenant/user/agent; treat retrieved memory as untrusted; validate/sanitize inputs before memory promotion; attach provenance metadata; enforce data retention/deletion policies.
Success Criteria	No unauthorized cross-tenant retrieval; zero high-impact actions from unverified memory; correct deletion/retention behavior; traceable containment of memory poisoning attempts.
Monitoring Measures	Log all memory writes/retrievals with actor, tenant, timestamp, sensitivity; continuously verify retrieval constraints; alert on anomalies; maintain versioned records and investigation tools.

3. Tool & Execution-Based Threats

Occurs when attackers abuse legitimate tools or execution mechanisms to invoke malicious actions, such as running malicious code, overloading resources, or manipulating system functions to gain unauthorized access or evade detection.

Aspect	Details

Relevant Attack Surfaces	System and Infrastructure layer; APIs, servers, schedulers, or automation components.
Related Issues	Unauthorized code execution; resource abuse; privilege escalation; increased operational costs.
Application Objectives	Agents perform tasks using tools (email, expense reports). Sensitive user data may be exploited if tools are manipulated.
Risks & Vulnerabilities	Unauthorized access to user data; manipulation or corruption of files; privilege abuse allowing attackers access to administrator-level functions.
Mitigation Measures	Baseline software auditing; remove outlier tools; limit programs running with elevated privileges; secure coding practices; robust safeguards in tool development.
Success Criteria	Agents only execute authorized actions; users cannot manipulate tools to perform unintended actions.
Monitoring Measures	Network monitoring; anomaly detection; robust, automated logging of tests and outputs; logs cross-referenced with prompts and access events.

4. Authentication & Identity Threats

Occurs when attackers gain higher privileges or access sensitive information by bypassing security controls or impersonating privileged entities.

Aspect	Details
Relevant Attack Surfaces	System (API keys, tokens, agent impersonation, log tampering); user interaction (malicious prompts through UI).
Related Issues	Data leaks, system compromise, API abuse, unauthorized transactions.
Application Objectives	Read/write access via APIs (Gmail, GDrive, Stripe), user interactions, human approval for transactions ≥ $500, chained tool/API usage.
Risks & Vulnerabilities	Email impersonation; API token theft; data manipulation; log tampering; unauthorized approvals.
Mitigation Measures	Least privilege access; secure API key storage; continuous logging and monitoring; input sanitization; sandboxed agent execution.

Success Criteria	Privacy, security, and data integrity maintained.
Monitoring Measures	Agents log each action with timestamp; logs monitored manually or programmatically for anomalies.

5. Human-in-the-Loop (HITL) Threats

Occurs when automated agents execute high-impact actions without necessary human review or confirmation.

Aspect	Details
Relevant Attack Surfaces	User Interaction (prompt manipulation/social engineering), System & Infrastructure (API misconfiguration or logic flaws).
Related Issues	Financial errors, compliance breaches, data leaks.
Application Objectives	AstralAssist processes sensitive HR, financial, and operational data requiring oversight.
Risks & Vulnerabilities	Automation bias, false reimbursements, data exposure, skipped human review.
Mitigation Measures	Enforce HITL gates for high-risk operations; require explicit reviewer confirmation; log all human decisions.
Success Criteria	Zero instances of autonomous approvals or unauthorized Drive access without human review.
Monitoring Measures	Append-only audit logs capturing reviewer identity, timestamp, context; integration with Drive Activity API; automated alerts for workflow violations.

6. Multi-Agent System Threats

Risks from coordination and communication between multiple AI agents, including unauthorized access, message tampering, data leakage, or coordination failures.

Aspect	Details

Relevant Attack Surfaces	Shared memory; cross-agent data exchange; overlapping user tasks; supply chain threats from agents.
Related Issues	Race conditions; inconsistent shared memory; message tampering; unverified/corrupted outputs aggregated.
Application Objectives	Orchestrated multi-agent tasks; context sharing via memory/vector databases; integration with third-party APIs.
Risks & Vulnerabilities	Cross-agent data leakage; inconsistent/corrupted outputs; insecure message transmission; privilege escalation.
Mitigation Measures	Strong agent authentication; role-based access control; encrypted inter-agent communication; sandboxed memory; API security audits.
Success Criteria	Secure agent interactions; consistent, access-controlled shared memory; conflict-free multi-agent coordination.
Monitoring Measures	Continuous monitoring of messages and task results; log authentication events and data access attempts; detect anomalies in shared memory.

Section VII: Threat Modeling

The following evaluates the application against the MAESTRO framework using the threat mapping from the Red Team test suite for each layer of the framework, and recommends the defenses that are needed to remediate those threats.

1. Foundations Model

Threat Name	Threat Definition	ASI Threat Category	Attack Surfaces	Affected Components
Adversarial Attacks	Maliciously crafted inputs designed to elicit incorrect or unexpected model behavior at inference time	Agency and Reasoning, Multi-Agent Systems, Memory, Tool and Execution	Model, User Interaction, System	All agents, policy enforcement

Backdoor Attacks	Inserting a malicious hidden trigger into training data which throws off model behavior on that specific input	Agency and Reasoning	Data, Model	Drive agent
Model Extraction/System Prompt Extraction	Using maliciously crafted queries to reconstruct the model's core logic or its system prompt	Authentication and Identity	Model	All agents

Cross Layer Propagation

Upstream Dependencies: Vulnerabilities from earlier layers might include denial of service attacks and data tampering.
Downstream Effects: Vulnerabilities at this layer could cause persistent flaws in reasoning and compromise multi-agent orchestration.
Cascading Risk: Compounded impacts include compromised multi-agent orchestration across layers.

Likelihood × Impact = Risk

- Likelihood: Medium
- Impact: Very High
- Overall Risk Score (LxI): 4/5
- Justification: The foundation model is central to AstralAssist. Vulnerabilities here can have large-scale consequences, especially if malicious outputs are committed to memory.

Planned Mitigations

- Preventative Controls: Instruct the model to detect malicious input through system prompt.
- Detective Controls: Input filtering and sanitization.
- Responsive Controls: System shutdown if cascading risk is too high.

---

2. Data Operations

Threat Name	Threat Definition	ASI Threat Category	Attack Surfaces	Affected Components

Data Poisoning	Compromising agent behavior by manipulating training data	Agency and Reasoning, Multi-Agent System	Data	All agents
Data Exfiltration	Theft of sensitive data	Authentication and Identity, Tool and Execution	Data, User Interaction	All agents
Compromised RAG Pipeline	Using prompt injection	Agency and Reasoning, Authentication and Identity, Memory	Data, User Interaction	Drive Agent

Cross Layer Propagation

- Upstream Dependencies: Vulnerabilities may arise from unchecked malicious user prompts (e.g., SQL injection) or unauthorized API use.
- Downstream Effects: May disrupt agent orchestration.
- Cascading Risk: Agent orchestration may be thrown off.

Likelihood × Impact = Risk

- Likelihood: Low
- Impact: Medium
- Overall Risk Score (LxI): 2/5
- Justification: Effects are most significant for the Drive agent due to its RAG pipeline.

Planned Mitigations

- Preventative Controls: Least-privilege data access, API authentication.
- Detective Controls: Alerts when agent reasoning significantly deviates, alerts when RAG database is altered.
- Responsive Controls: System shutdown.

---

3. Agent Frameworks

Threat Name	Threat Definition	ASI Threat Category	Attack Surfaces	Affected Components

Emergent Deviation Threat	Agents deviate from intended task objectives due to recursive reasoning loops or conflicting goals	Agency & Reasoning	Model, Data, User Interaction	Task planner, reasoning module, role manager
Role Escalation Attack	A compromised or misconfigured agent assumes privileges or functions outside its intended scope	Privilege & Boundary Violations	System & Infrastructure, Supply Chain	Orchestration framework, access control modules
Feedback Loop Exploitation	Malicious inputs exploit reasoning chains to induce biased or harmful decision outputs	Model Manipulation	Data, User Interaction	Memory store, reasoning pipeline, input preprocessor

Cross Layer Propagation

- Upstream Dependencies: Poor data validation from preprocessing layers can propagate faulty reasoning signals.
- Downstream Effects: Incorrect routing or outputs may corrupt evaluation metrics or user-facing outputs.
- Cascading Risk: Erroneous reasoning may amplify across agents, creating self-reinforcing behavior loops.

Likelihood × Impact = Risk

- Likelihood: Medium
- Impact: High
- Overall Risk Score (LxI): 4/5
- Justification: Multi-agent reasoning systems are inherently complex; minor deviations cascade quickly.

Planned Mitigations

- Preventative Controls: Role-based access control, bounded reasoning contexts, policy-based orchestration rules.
- Detective Controls: Behavior anomaly detection, real-time tracing of inter-agent messages.
- Responsive Controls: Auto-isolation of deviating agents, system rollback to safe checkpoints.

---

4. Deployment & Infrastructure

Threat Name	Threat Definition	ASI Threat Category	Attack Surfaces	Affected Components
API Injection & Exploit	Exploit open/unvalidated API endpoints to execute unauthorized commands	System & Infrastructure	API endpoints, server logic	Web gateways, inference API
Configuration Drift	Mismatched infrastructure settings expose vulnerabilities or open ports	Operational Risk	Cloud resources, container orchestration	Docker images, Kubernetes nodes
Supply Chain Compromise	Malicious libraries or hardware injected through build/deployment pipelines	Supply Chain	Dependencies, CI/CD integrations	Model weights, runtime binaries

Cross Layer Propagation

- Upstream Dependencies: Misconfigured environment variables may leak secrets.
- Downstream Effects: Insecure deployments can expose user data and corrupt evaluation layers.
- Cascading Risk: Exploited infrastructure may lead to model exfiltration and persistent backdoors.

Likelihood × Impact = Risk

- Likelihood: Medium
- Impact: Very High
- Overall Risk Score (LxI): 4.5/5
- Justification: Cloud exposure risks are critical but partially mitigated by platform security tools.

Planned Mitigations

- Preventative Controls: Infrastructure-as-Code auditing, signed builds, dependency scanning.
- Detective Controls: Intrusion detection systems, cloud security posture monitoring.
- Responsive Controls: Automated rollback, key revocation, incident response playbooks.

---

5. Evaluation & Observability

Threat Name	Threat Definition	ASI Threat Category	Attack Surfaces	Affected Components
Metrics Poisoning	Manipulating feedback data or evaluation metrics to mask unsafe behavior	Data Integrity	Data pipelines, feedback logs	Evaluation dashboards, metric aggregators
Logging Leakage	Sensitive user or model data captured unintentionally in logs	Data Privacy	Log files, monitoring agents	Observability tools, telemetry collectors
Shadow Metric Drift	Unnoticed metric drift hides degradation in system performance or bias	Model Performance Risk	Evaluation framework	Scoring functions, analytics backend

Cross Layer Propagation

- Upstream Dependencies: Faulty model outputs feed misleading metrics.
- Downstream Effects: Misleading dashboards guide incorrect retraining or deployment decisions.
- Cascading Risk: Feedback corruption impacts reasoning and user-facing trust metrics.

Likelihood × Impact = Risk

- Likelihood: High
- Impact: Medium
- Overall Risk Score (LxI): 4/5
- Justification: Observability vulnerabilities impact the entire system.

Planned Mitigations

- Preventative Controls: Secure logging policies, sanitized, validated feedback data.
- Detective Controls: Drift detection tools, anomaly monitoring for metric spikes.
- Responsive Controls: Rapid alerting, retraining validation checkpoints.

---

6. Security & Compliance

Threat Name	Threat Definition	ASI Threat Category	Attack Surfaces	Affected Components

Lack of Logs	No record of agent operations; no tracking of potential threats	Multi-Agent System, Human-in-the-Loop	System	All agents
Lack of Explainability	Absence of explainability in reasoning chains	Agency and Reasoning, Multi-Agent System	Model, System	All agents
Insufficient Technical Defenses	Other threat layers lack proper implementation of critical defenses	All categories	System	All agents, entire system

Cross Layer Propagation

- Upstream Dependencies: Biased data, poorly trained model, lack of secure multi-agent system.
- Downstream Effects: Poor model performance, agent orchestration, and multi-agent compromise.
- Cascading Risk: Untracked vulnerabilities may lead to persistent attacks.

Likelihood × Impact = Risk

- Likelihood: 5
- Impact: 5
- Overall Risk Score (LxI): 5/5
- Justification: Multi-agent systems make cascading risk highly likely.

Planned Mitigations

- Preventative Controls: Threat modeling, red teaming, risk register, incident response plan, established logging system.
- Detective Controls: Regular monitoring, logging, system audits.
- Responsive Controls: Execute incident response plan.

---

7. Agent Ecosystem

Threat Name	Threat Definition	ASI Threat Category	Attack Surfaces	Affected Components
Ecosystem Manipulation	External actors influence inter-agent communication or shared memory for malicious outcomes	External Influence	APIs, communication channels	Shared memory, routing APIs

Model Interoperability Exploit	Misaligned models or third-party integrations cause unexpected agent behaviors or data leaks	Supply Chain	Third-party APIs, dependency interfaces	API connectors, data translators
Social Engineering via Agent Output	Adversaries exploit conversational outputs to deceive users or chain instructions	User Interaction	Chat interfaces, output channels	User interaction layer, content moderation filter

Cross Layer Propagation

- Upstream Dependencies: Weak reasoning or role definition from agent frameworks increases ecosystem exposure.
- Downstream Effects: Misuse of data or actions executed in external systems.
- Cascading Risk: Coordinated manipulation across agents could escalate to systemic misinformation or resource misuse.

Likelihood × Impact = Risk

- Likelihood: High
- Impact: Very High
- Overall Risk Score (LxI): 5/5
- Justification: Ecosystem vulnerabilities have broad external exposure and user impact potential.

Planned Mitigations

- Preventative Controls: Output moderation, API authentication, least-privilege permissions.
- Detective Controls: Real-time anomaly detection for inter-agent communications.
- Responsive Controls: Agent isolation protocols, human-in-the-loop review for critical outputs.

Section VIII: Governance Connection

Risk Register

Risk ID	Risk Description	Likelihood	Impact	Severity	Mitigations in Place	Owner
R-01	Prompt injection leading to PII exfiltration from	Medium	High	Sev 1	Non-persistent employee database;	HR Database

	HR database				prompt filtration	Owner
R-02	Hallucination causing agents to drift from designated functions/workflows	Medium	Medium -High	Sev 2-3	Strong/robust dispatch logic built into orchestrator agent	Dev Team
R-03	Exceeding third-party API rate limits preventing system from fully functioning	High	Medium	Sev 1	Enforcing rate limits and cooldowns	Dev Lead
R-04	Prompt injection leading to unauthorized financial transactions	Medium	High	Sev 1	Require explicit user confirmation for financial requests	Dev Team
R-05	Email impersonation for social engineering	Low	High	Sev 2	Require explicit user confirmation for emails to external domains or containing financial requests, rate limiting on email sends	Governance Lead
R-06	Poisoned documents injected into RAG store causing retrieval drift, unsafe responses, or backdoor behaviors	Medium	High	Sev 1	Attested sources, provenance checks, strip active content, periodic re-indexing	Dev Team
R-07	High-volume or structured queries used to extract proprietary model parameters or embeddings	Medium	High	Sev 2	Rate limiting, anomaly detection, limit output verbosity, watermarking	Dev Team
R-08	AI agents produce discriminatory or unfair decisions (HR screening, expense approvals)	Medium	High	Sev 3	Bias testing, fairness audits, human-in-the-loop , logging decisions	Governance Lead
R-09	Remote Code Execution attempt on MCP server	Medium	High	Sev 1	Disable shell tools, strict schema	Dev Team

	through prompt injection or tool schema manipulation				validation, allowlist enforcement, code signing	
R-10	Compromised third-party model, dataset, or package introducing backdoors or malicious behavior	Low	Very High	Sev 2	Integrity checks, vendor attestation, SBOM, periodic audits, signed weights	Dev Lead

Incident Response Plan

Prompt injection leading to PII exfiltration:

Prompt injection could allow an attacker to exfiltrate personally identifiable information from the HR database. Detection is implemented via prompt filtration, and in severe cases, a killswitch disables the HR API. Containment involves disconnecting the HR database from the system and strengthening guardrails against malicious prompts. Recovery is achieved by re-enabling the HR API with additional data protection measures. This scenario is governed by Article 10 (Data & Data Governance), which ensures proper governance over sensitive data, Article 15 (Accuracy, Robustness, and Cybersecurity), which mandates that AI systems are secure against malicious inputs, Article 62 (Reporting of Serious Incidents), which requires reporting such breaches, and Article 72 (Post-Market Monitoring and Market Surveillance), which emphasizes ongoing monitoring to prevent recurrence.

Hallucination causing agents to drift from designated functions/workflows:

AI agents may experience goal drift or hallucination, deviating from their assigned tasks. Detection relies on logs that reveal agents performing unintended actions. The killswitch disables affected agents or the entire system if necessary. Containment involves isolating the agents, identifying the impacted workflow, and enhancing their robustness. Recovery re-enables the agents once safeguards are applied. Relevant EU AI Act Articles include Article 14 (Human Oversight)to ensure human monitoring of agents, Article 15 to improve robustness and prevent hallucination, and Article 72 for post-market monitoring to identify emerging drift in deployed agents.

Exceeding third-party API rate limits preventing system from fully functioning:

When AI agents exceed API rate limits, system functionality may be hindered. Detection is performed through logs monitoring API usage, with a killswitch that disables affected API functions. Containment includes enhancing rate-limit enforcement, and recovery restores API functions after adjustments. This threat aligns with Article 15 (Accuracy, Robustness, and Cybersecurity), which emphasizes technical safeguards to prevent operational failures, and Article 72 (Post-Market Monitoring) to ensure monitoring identifies repeated or unusual API usage patterns.

Prompt injection leading to unauthorized financial transactions:
Malicious prompts could cause agents to execute financial transactions without proper authorization. Detection involves prompt filtration and logging HITL bypass attempts. The killswitch disables the expense management agent. Containment ensures adherence to HITL policies, adds prompt filters, and includes additional verification steps. Recovery re-enables the agent gradually, initially limiting transaction amounts. This threat is directly relevant to Article 5 (Prohibited AI Practices) for preventing high-risk behaviors, Article 10 (Data Governance) for secure handling of financial data, Article 14 (Human Oversight) to maintain oversight, Article 15 (Robustness and Cybersecurity) to reduce susceptibility to prompt attacks, and Articles 62 and 72 for incident reporting and post-market monitoring.

Email impersonation for social engineering:
Attackers could exploit AI agents to craft misleading emails. Detection involves monitoring logs and human reports. The killswitch disables the email agent, while containment involves verification of email content, AI-based social engineering checks, and alerts to the organization. Recovery reinstates the email agent after adjustments. This threat implicates Article 5 (Prohibited Practices), Article 10 (Data & Governance) for secure handling of communications, Article 14 (Human Oversight) to monitor outputs, Article 15 (Robustness) to prevent automated misuse, and Article 72 for ongoing monitoring of communication safety.

Poisoned or manipulated RAG documents causing retrieval drift, unsafe outputs, or backdoor behaviors:
Malicious or manipulated documents in the RAG store could compromise outputs. Detection is done through drift alerts, untrusted source checks, and anomaly detection. The killswitch freezes the vector database, quarantines untrusted documents, rebuilds embeddings from verified sources, and strips active content. Recovery restores normal operation after reindexing and validation. Relevant EU AI Act guidance includes Article 10 (Data & Governance) for provenance control, Article 15 (Robustness & Cybersecurity) to ensure system resistance to malicious inputs, and Article 72 (Post-Market Monitoring) for continuous surveillance of RAG pipeline integrity.

Adversary attempts to extract model parameters or embeddings through structured/frequent queries:
Frequent or structured queries may allow attackers to infer sensitive model details. Detection involves monitoring query spikes, unusual patterns, or verbosity requests. The killswitch disables affected agent endpoints, and containment blocks offending IPs, enforces rate limits, and rotates keys if needed. Recovery restores services with enhanced safeguards. This threat is covered by Article 15 (Robustness & Cybersecurity) to defend against model theft, Article 72 (Post-Market Monitoring) to ensure ongoing detection, and Article 73 (Serious Incident Reporting) if intellectual property is compromised.

AI agents generating discriminatory responses (HR, finance, or operational workflows):
Agents may output biased or unfair decisions. Detection relies on fairness metrics, user complaints, and logs of demographic-correlated outputs. The killswitch disables affected agents or switches them to HITL mode. Containment involves fairness audits, data sampling correction, and updating decision logic. Recovery re-enables agents with improved fairness constraints and periodic audits. Relevant articles include Article 10

(Data & Governance) to ensure fair and representative datasets, Article 14 (Human Oversight) to intervene in biased decisions, Article 15 (Accuracy/Robustness) to enforce consistent outputs, and Article 72 (Post-Market Monitoring) for continuous fairness monitoring.

Attacker attempts RCE through prompt injection, tool schema manipulation, or MCP envelope abuse:
Attackers may attempt to execute arbitrary code on MCP servers. Detection relies on logs showing shell-like commands or disallowed egress. The killswitch disables write/execute tools and isolates the server. Containment patches schemas, enforces envelope signatures, and uses certificate pinning. Recovery progressively reinstates tools with progressive testing. This scenario references Article 15 (Robustness & Cybersecurity) to defend against RCE, Article 62 (Reporting of Serious Incidents) for prompt reporting of security breaches, and Article 72 (Post-Market Monitoring)to continuously track potential vulnerabilities.

Section IX: System Audit

MAP

AstralAssist is a multi-agent personal assistant designed to manage emails, retrieve and modify Google Drive documents, process expenses, interact with HR-type data, and perform multi-step administrative tasks. Its core consists of a large language model (LLM), multi-agent orchestration, tool APIs, a retrieval-augmented generation (RAG) memory, a FastAPI backend, and OAuth/service account authentication credentials. Users interact through a web UI, submitting requests that the orchestrator parses to determine which agents and tools are required. Agents, such as email, document, or finance agents, execute tasks based on LLM-generated plans, calling external APIs as needed. All actions, results, and errors are logged in memory, allowing traceability, and responses are returned to the user in a formatted output.

The system runs in a cloud-hosted environment with high privileges and deep integration with Google Workspace and a custom HR backend. It uses long-lived OAuth tokens, multi-agent orchestration, persistent memory, and external APIs. Key stakeholders include employees, whose personal information may be exposed; HR professionals, who depend on secure and accurate data; system administrators responsible for credentials and infrastructure security; and end-users, who could be affected by malicious agent behavior or prompt injection attacks. Assets at risk include employee data, emails, financial receipts, Drive documents, memory logs, authentication tokens, and API keys. Persistent memory introduces risks such as cross-agent contamination and “shadow memory” retention, which can complicate privacy and auditability. Human-in-the-loop (HITL) threats arise when tasks like email sending or HR approvals occur without mandatory human intervention. Authentication and identity threats involve OAuth tokens and delegated identities, where LLMs could leak credentials or agents could exceed intended privileges. Agency and reasoning threats occur because autonomous planning pipelines can propagate errors through flawed reasoning or misinterpretation. Multi-agent coordination introduces emergent failures when inter-agent communication is misaligned. Finally, tool and

execution threats arise from LLM-generated plans invoking external APIs unsafely, potentially causing unauthorized emails, file modifications, or HR actions.

MEASURE

To assess these threats, AstralAssist employs rigorous assurance testing and monitoring metrics. Memory-based risks are tested through adversarial prompts and ingestion scenarios to ensure personally identifiable information (PII) is masked or skipped, such as submitting sample CSVs or binary files containing sensitive data. HITL controls are validated by confirming that high-risk operations, like email sends or HR approvals, cannot execute without explicit human confirmation. Authentication threats are measured by verifying that secret tokens, OAuth credentials, and access permissions cannot be exposed or bypassed.

Agency and reasoning threats are evaluated through malformed or unsafe commands to confirm the system refuses unsafe actions, while multi-agent threats are assessed by sending conflicting or ambiguous prompts to check for unintended cross-agent contamination or duplicated operations. Tool and execution threats are tested by attempting disallowed attachments, paths, or IDs to verify safe failure without side effects. Pass/fail criteria include detection of unauthorized actions, unsafe reasoning, unsafe API execution, or leakage of PII. Failures trigger mitigation steps such as credential rotation, stricter input validation, or updated filters. Continuous monitoring tracks system behavior and flags anomalies in real time, allowing teams to intervene before threats escalate.

MANAGE

Management strategies combine technical, procedural, and operational controls to mitigate threats. Technical controls include input validation, sandboxing, identity enforcement, and credential management. Procedural controls enforce HITL workflows, change control, and escalation protocols to prevent unsupervised operations. Access control and privileging models maintain least-privilege principles, while data lifecycle policies govern retention, deletion, and redaction. Monitoring systems detect anomalies, enforce operational boundaries, and track agent-to-agent communication to reduce the risk of emergent failures.

Incident response plans define containment, rollback, and remediation strategies, with clear ownership and timelines for mitigation tasks. Operational success is measured by preventing PII leakage, enforcing HITL checkpoints, and ensuring multi-agent coordination functions reliably. Security dashboards and automated alerts allow administrators to act quickly, and audit logs provide a complete record of system behavior. Management also includes updating training data, retraining agents, and refining reasoning logic to reduce errors while improving compliance with organizational policies. These combined measures create a proactive defense posture, ensuring the system remains safe and reliable under normal and adversarial conditions.

GOVERN

Governance frameworks support accountability, oversight, and regulatory compliance. Agency and reasoning threats are governed through separation of agent goals, mandatory HITL approvals for high-risk operations, and adherence to regulatory requirements such as the EU AI Act and GDPR. Memory-based threats are addressed via encryption, controlled retention, and audit logs to ensure traceability of sensitive data. Tool execution is governed with sandboxing, API security, and rate limiting, alongside policies that enforce ethical and regulatory standards. Authentication policies include privileged access management, OAuth security, and token protection to prevent misuse or leakage.

HITL threats are managed with mandatory human oversight for critical operations, and multi-agent coordination is controlled via authorization boundaries, defined communication protocols, and ongoing monitoring. Oversight is strengthened through red-teaming exercises, monthly review board meetings, continuous logging, and maintenance of an updated risk register and incident response plan. Documentation clearly defines user guidance, escalation procedures, and monitoring responsibilities. These governance measures align operational, technical, and regulatory objectives, ensuring the system remains compliant, accountable, and resilient against emerging threats.

Section X: Team Specific Activity

As discussed in class, this will be addressed in the final presentation.