AI Multi-Agent Workflow Automation Platform: Project Proposal

Youssef Awad (ID: 8179)
Mohamed Morsy (ID: 8199)
Ahmed Samir (ID: 8120)
Youssef Ibrahim (ID: 8177)
Youssef Attia (ID: 8164)
Karim Mohamed (ID: 8237)

1 Team Members

- Youssef Awad (ID: 8179)
- Mohamed Morsy (ID: 8199)
- Ahmed Samir (ID: 8120)
- Youssef Ibrahim (ID: 8177)
- Youssef Attia (ID: 8164)
- Karim Mohamed (ID: 8237)

2 Project Overview

Modern professionals spend excessive time navigating multiple applications (e.g., email, calendar, task managers, and reporting tools). Existing workflow automation tools like Zapier and IFTTT rely on rigid, rule-based systems, limiting their adaptability.

This project proposes an AI-powered multi-agent platform where intelligent agents collaborate to automate digital tasks. Leveraging natural language processing (NLP), intent recognition, summarization, and reasoning, these agents dynamically adapt to user needs, surpassing traditional automation limitations.

Example Workflow: A client emails: "Can we meet Tuesday afternoon?" The Email Agent extracts the meeting intent, the Calendar Agent identifies available slots, the Task Agent creates a reminder, and the Report Agent logs the interaction for summaries.

3 Objectives

- Develop a scalable multi-agent orchestration framework for seamless agent collaboration.

- Build specialized AI agents:
  - **Email Agent**: NLP-driven intent recognition and named entity recognition (NER).
  - **Calendar Agent**: Intelligent schedule optimization.
  - **Task Agent**: Automated task creation and prioritization.
  - **Report Agent**: Summarization of tasks and events.
  - **Security & Spam Detection Agent**: Filtering spam and fraudulent emails.
- Create an intuitive drag-and-drop workflow builder for non-technical users.
- Provide a user-friendly dashboard to monitor workflows, agent performance, and spam detection analytics.
- Ensure scalability with a cloud-based backend and robust API integrations.

4 System Architecture

4.1 Core Components

4.1.1 Frontend (Web Application)

Technology: React with Tailwind CSS.
Features:

- Drag-and-drop workflow builder for connecting agents.
- Dashboard displaying active workflows, task statuses, and time-saving metrics.
- Reports page with exportable summaries.
- Spam & Fraud Monitoring Dashboard for spam statistics and alerts.

4.1.2 Backend (API + Orchestration)

Technology: FastAPI or Django REST framework.
Components:

- Agent orchestration engine (using LangChain, AutoGen, or CrewAI).
- Workflow execution engine for triggers and inter-agent data passing.
- Databases: PostgreSQL (structured data) and MongoDB (logs).
- Caching: Redis for performance optimization.

4.1.3 AI/ML Layer

- **Email Agent**: Fine-tuned BERT for intent classification; SpaCy for NER.

- **Calendar Agent**: Constraint satisfaction algorithms and reinforcement learning for scheduling.
- **Task Agent**: ML-based classification and regression for task categorization and prioritization.
- **Report Agent**: Summarization using Pegasus or T5 models; clustering for trend detection.
- **Security & Spam Detection Agent**: Transformer models (BERT, RoBERTa) for spam/phishing detection, with domain reputation checks and anomaly detection.

4.1.4 Integration Layer

APIs: Gmail, Outlook, Google Calendar, Microsoft To-Do, Trello, Slack.

5 AI Agents in Detail

5.1 Email Agent

Goal: Identify actionable items from emails.
Techniques:

- Intent classification (e.g., meeting request, task assignment) using fine-tuned BERT.
- Entity extraction (e.g., date, time, people) using SpaCy NER.

Output: Structured JSON with intent and entities, passed to other agents.

5.2 Calendar Agent

Goal: Optimize event scheduling.
Techniques:

- Constraint satisfaction for slot allocation.
- Reinforcement learning for user preference optimization.

Output: Suggested meeting slots and calendar API updates.

5.3 Task Agent

Goal: Transform unstructured text into structured tasks.
Techniques:

- Classification for task categories (e.g., work, personal, urgent).
- Regression for priority scoring.

Output: Task entries in integrated task management apps.

5.4 Report Agent

Goal: Generate concise productivity reports.
Techniques:

- Text summarization using Pegasus or T5 models.
- Clustering for detecting task/event trends.

Output: PDF or HTML reports (daily/weekly).

5.5 Security & Spam Detection Agent

Goal: Filter spam, phishing, and fraudulent emails before processing.
Techniques:

- Fine-tuned transformer models (BERT, RoBERTa) for classification.
- Domain reputation checks, blacklist/whitelist analysis.
- Anomaly detection for suspicious links and attachments.

Output:

- Classifies emails as 'Spam', 'Fraudulent', or 'Legitimate'.
- Legitimate emails proceed to the Email Agent; others are quarantined with user notifications.

6 Workflow Example

Scenario: An email arrives: "Hi, let's meet Thursday at 3 PM to discuss project deadlines."

1. **Security & Spam Detection Agent**: Scans and classifies the email as legitimate.
2. **Email Agent**: Detects meeting intent and extracts [Thursday, 3 PM].
3. **Calendar Agent**: Checks availability and books the slot.
4. **Task Agent**: Creates a task: "Prepare project deadline updates."
5. **Report Agent**: Logs the meeting and task in the weekly summary.

7 Frontend Features

- **Workflow Builder**: Drag-and-drop interface to connect agents (e.g., Email $\rightarrow$ Calendar $\rightarrow$ Task $\rightarrow$ Report).
- **Dashboard**:
  - Real-time view of active workflows and task statuses.
  - Metrics on time saved through automation.

- **Reports Page**: Auto-generated summaries with export options (PDF/HTML).
- **Spam & Fraud Monitoring Dashboard**:
  - Weekly statistics on detected spam/phishing emails.
  - Alerts for high-risk phishing attempts.
  - 'Review & Release' feature for managing false positives.

8 Tools & Technologies

- **AI/ML**: Hugging Face Transformers, SpaCy, PyTorch, Scikit-learn.
- **Backend**: FastAPI/Django, PostgreSQL, MongoDB, Redis.
- **Frontend**: React, Tailwind CSS, Drag-and-Drop libraries.
- **Orchestration**: LangChain, CrewAI, AutoGen.
- **Cloud**: AWS (Lambda, S3, EC2), Docker, Kubernetes.
- **Integrations**: Gmail API, Google Calendar API, Trello/Slack API.

9 Architecture Update: Spam/Fraud Detection

The workflow integrates the Security & Spam Detection Agent as the entry point:

1. New email arrives.
2. Security Agent scans and classifies the email.
3. Legitimate emails proceed to the Email Agent.
4. Spam/fraudulent emails are quarantined and logged in the Spam Report dashboard.

10 Expected Outcomes

- A fully functional multi-agent platform for workflow automation.
- Demonstrated capabilities in intent recognition, task generation, scheduling, summarization, and spam detection.
- An intuitive frontend platform accessible to non-technical users.
- Quantifiable evidence of time savings compared to manual workflows.