CrewAI Agent-Based Model Design - Instructions

LEAD 352 – Check In 2

Table of contents

0.1 Team Information

Team Name: United Front

Team Members:

1. Jacqueline Chesnowitz (GitHub Username: jackiechesnowitz)
2. Alexa Hall (GitHub Username: alexa809)
3. Joshua Faughnan (GitHub Username: jfaughn2)
4. Joseph Valli (GitHub Username: Jvalli1)
5. Molly Singer (GitHub Username: _____)
6. Abigail Uhrin (GitHub Username: abbyuhrin)

GitHub Repository URL: https://github.com/alexa809/Teamcohesion.git

0.2 Project Overview

In this assignment, you will design a complete CrewAI agent-based simulation. You'll write the actual implementation code for the key components that could be executed in the future.

Describe your overall simulation concept (what problem is your agent-based model addressing?):

Research Question: How does team cohesion affect team performance?

Task Design: In our simulation, one team (control team) will perform their regular tasks at work and then continue to live their personal lives. Essentially, the control team will only engage in work-related activities while the second team (variable team) will perform their tasks as well but then will congregate for team bonding afterward. Meanwhile, the variable team will be doing activities such as getting dinner together, having happy hours, and going to sporting events.

Team Composition Variables: Each team will have 6 members, which is ideal for effectiveness and what teams prefer. These agents will have different expertise areas, a mix of personalities, and various cultural backgrounds. Doing so will cultivate a team environment similar to that found in the corporate workforce.

0.3 Setup and Imports

LEAVE THIS SECTION AS IT IS other than team name (you will use this in your code to start):

```
#!/usr/bin/env python3
"""
CrewAI Agent-Based Model Design

Team: United Front
"""

# Import necessary libraries
from crewai import Agent, Task, Crew, Process
from langchain_community.llms import Ollama
# Add any other imports you need

# Define the language model that will power your agents
llm = Ollama(model="llama2")
```

0.4 Agent Definitions

Define your agents with distinct roles, goals, and backstories. You need at least 3 agents, but can create more if needed.

0.4.1 Agent 1

```
# Define your first agent
Optimist = Agent(
    role="'Optimist'",  # The agent's function or position in the team
```

```
    goal="Achieve the best possible outcome and maintain high morale. Complete the project ahead of schedule with maximum team satisfaction. Always sees the bright side, even in failure. Might focus on motivating others or improving team dynamics",  # What the agent aims to accomplish
    backstory="You maintain a positive attitude and boost team morale.', llm=custom_llm",  # Background information that shapes the agent's approach
    verbose=True,
    llm=llm
)
```

0.4.2 Agent 2

```
# Define your second agent
Learner = Agent(
    role="'Learner'",
    goal="Gain knowledge or experience, even if it means slower progress. Prioritizes understanding over efficiency. May try new strategies or seek feedback",
    backstory="You are constantly seeking knowledge and asking questions.', llm=custom_llm",
    verbose=True,
    llm=llm
)
```

0.4.3 Agent 3

```
# Define your third agent
Procrastinator = Agent(
    role="Procrastinator",
    goal="'Delay tasks until the last possible moment, while still trying to meet deadlines. Focused on minimizing effort upfront. May be driven by urgency or last-minute pressure.",
    backstory="'You put off tasks until the last minute, causing delays.', llm=custom_llm",
    verbose=True,
    llm=llm
)
```

0.4.4 Additional Agents (Optional)

```
# Define additional agents as needed
Complainer = Agent(
    role="Complainer",
    goal="Point out flaws, express dissatisfaction, or find support for their grievances. May be motivated by attention, change, or simply venting. Could help identify system weaknesses (if channeled productively)",
    backstory="You constantly find problems but rarely offers solutions.', llm=custom_llm",
    verbose=True,
    llm=llm
)
```

0.4.5 Additional Agents (Optional)

```
# Define additional agents as needed
Collaborator = Agent(
    role="Collaborator",
    goal="Ensure teamwork, harmony, and shared success. Prioritizes group achievement over individual gains. Seeks alignment, delegation, and cooperation",
    backstory="you work well with others and mediate conflicts', llm=custom_llm",
    verbose=True,
    llm=llm
)
```

0.4.6 Additional Agents (Optional)

```
# Define additional agents as needed
Foodie = Agent(
```

```
    role="'Foodie'",
    goal="'Seek comfort, reward, or motivation through food-related experiences. Might use food as an incentive, distraction, or coping mechanism. Could influence group morale with treats or breaks.",
    backstory="'You bring the best snacks and always know the best lunch spots.', llm=custom_llm",
    verbose=True,
    llm=llm
)
# Explain this agent's personality traits and other characteristics:
```

0.5 Task Definitions

Define the tasks that your agents will perform. Create at least one task for each agent.

0.5.1 Task 1

```
# Define your first task
optimist_task = Task(
    description="Encourage and motivate teammates throughout all stages of the Der Eisendrache Easter Egg, maintaining high morale even during difficult or frustrating challenges.",  # Clear statement of what the task entails
    expected_output="'Consistently high team morale, motivated agents, and a positive atmosphere that contributes to smoother teamwork and potentially faster task completion.",  # What the task should produce when complete
    agent=optimist  # Which agent will perform this task
)
```

0.5.2 Task 2

```
# Define your second task
learner_task = Task(
    description="'Explore and implement various strategies or alternative methods for completing the Der Eisendrache Easter Egg, documenting insights and lessons learned during the process.",
    expected_output="A detailed summary of strategies tested, clearly outlining what worked best, and valuable learning points that contribute to the team's overall strategy",
    agent=learner,
    context=""  # Optional: Provide additional context for the task
)
```

0.5.3 Task 3

```
# Define your third task
procrastinator_task = Task(
    description="Manage personal responsibilities within the Easter Egg completion, intentionally delaying certain tasks but ensuring they are completed effectively under the pressure of approaching deadlines",
    expected_output="

    expected_output='Tasks successfully completed, though with minimal lead time, potentially revealing efficiencies gained or lost from time pressure",
    agent=procrastinator
)
```

0.5.4 Additional Tasks (Optional)

```
# Define additional tasks as needed
complainer_task = Task(
    description="Critically identify flaws, weaknesses, or inefficiencies in the current strategy and teamwork dynamics during the Easter Egg completion process, clearly voicing areas for improvement",
    expected_output="A clear list of identified problems or potential pitfalls with suggestions (if possible), aiding the team in preemptively addressing these concerns",
    agent=complainer
)
```

0.5.5 Additional Tasks (Optional)

```
# Define additional tasks as needed
collaborator_task = Task(
    description="Facilitate teamwork, mediate conflicts, delegate responsibilities effectively, and ensure all agents work harmoniously towards completing the Easter Egg efficiently",
    expected_output="Strong team cooperation, effective conflict resolution, and efficient role delegation, leading to smooth progression through Easter Egg steps",
    agent=collaborator
)
```

0.5.6 Additional Tasks (Optional)

```
# Define additional tasks as needed
foodie_task = Task(
    description="Enhance team morale and maintain energy levels by coordinating timely snack breaks, providing treats, and recommending meals or breaks strategically during critical or exhausting moments of the Easter Egg",
    expected_output="Consistently high team satisfaction and maintained energy levels throughout the task, boosting morale and potentially increasing performance",
    agent=foodie
)
```

0.6 Process Flow Design

0.6.1 Crew Setup

```
# Set up your crew with the defined agents and tasks
crew = Crew(
    agents=[optimist, learner, procrastinator, complainer, collaborator, foodie],
    tasks=[optimist_task, learner_task, procrastinator_task, complainer_task, collaborator_task, foodie_task ],

    verbose= True # Prints detailed logs of agent interactions
    process= 'sequential', # Agents work one after another
)

# Explain why you chose this process type:

# Start the collaboration and execute the tasks

result = crew.kickoff()

# Print the final output

print("Final Output:", result)
```

0.6.2 Workflow Diagram

Sketch or describe the workflow of your simulation (how tasks and agents interact):

Work

The team must successfully complete the complex Der Eisendrache Easter Egg. This involves progressing through multiple elaborate steps, including activating all dragon heads, obtaining upgraded bows (Wrath of the Ancients), completing bow upgrade quests, performing ritual steps, time-travel sequences, and ultimately defeating the final boss. Agents will need to coordinate closely, communicate effectively, manage resources wisely, and navigate unexpected challenges to accomplish this goal efficiently and successfully.

Work subtask

To successfully complete the Der Eisendrache Easter Egg, the team must first collaborate to activate the power and fully open the map, providing the essential foundation for subsequent steps. Next, they'll work together to feed all three dragon heads by strategically defeating zombies, ultimately earning the powerful Wrath of the Ancients bow. After this, each agent will undertake the intricate quest of upgrading one of the four elemental bows—Storm, Void, Wolf, or Fire—with roles tailored to their unique personalities: the Optimist offering encouragement, the Learner exploring new methods, the Collaborator facilitating teamwork, the Procrastinator pushing through tight deadlines, the Complainer highlighting challenges, and the Foodie providing snacks to boost morale.

Once the bows are upgraded, the team must carefully navigate the wisps and time travel sequences, accurately complete the Simon Says computer puzzles, and defend the Keeper through multiple charging rituals around the castle. After thorough preparation and strategic resource management, the agents will face the challenging final boss battle against the Gatekeeper, demanding precise coordination, role clarity, and unwavering teamwork. Upon defeating the boss and securing the summoning key, the agents can reflect on their experiences, assessing how effectively their cohesion and bonding influenced overall performance and satisfaction during this demanding task.

Fun

The fun task will be going out to happy hour or dinner after working on the task for the day as a means to destress and regroup.

0.7 Implementation Considerations

0.7.1 Challenges

What challenges do you anticipate in implementing this simulation?

Role conflicts – The Procrastinator may delay, frustrating the Optimist.

Miscommunication – The Collaborator must keep everyone aligned.

Execution errors – The Learner may struggle with complex steps.

Time pressure – The Procrastinator may slow progress on tight steps.

Negativity – The Complainer might lower team morale.

Focus issues – The Foodie may distract with snack breaks.

Resource mismanagement – The Learner may waste ammo or points.

Frustration levels – The Complainer may amplify mistakes.

Overcomplication – The Collaborator may over-explain strategies.

Fatigue – The entire team may burn out before happy hour.

0.7.2 Future Enhancements

Describe potential future enhancements to your simulation:

To further enhance our simulation, we may outline roles and responsibilities to be more closely correlated to the tasks found within the easter egg. For the time being, our roles and tasks outlined are more so in terms of team cohesion and positively affecting the group in terms of maintaining team morale. If our team is having difficulty with logistics of the Easter Egg in itself, we will have to rework the task outline.

0.8 Submission Checklist (confirm these are completed)

- Completed all required agent definitions (minimum 3)
- Created at least one task for each agent
- Designed the process flow and team structure
- Considered implementation challenges and future enhancements