**Team Adapt Onboarding Guide**

**Project Overview**

Welcome to Team Adapt! This document outlines the best practices, instructions, and project setup guidelines for the AdaptAI-Alpha project. Our goal is to ensure consistency, efficiency, and clarity throughout the project lifecycle.

**Working Directory:**  
C:\Users\ChaseRemmen\ADAPT\Projects\_MAIN\AdaptAI-Alpha

**Objectives**

* **Planning & Approval:** Before writing any code, discuss the plan for each step with the project lead (Chase). Once approved, write a master script (e.g., MASTER\_<Task\_Name>.py) that will create and run the modular scripts with detailed logging and CLI output.
* **Task Breakdown:** Break down tasks into modular scripts created and executed by the master script. Focus on keeping the main tasks centralized in the master script to maintain clarity.
* **Modularization and Microservices:** Our architecture employs modularization and microservices to enhance scalability, maintainability, and ease of deployment.

**Directory Structure**

All files should be organized logically under the working directory. Here's the structure to follow:

bash

Copy code

AdaptAI-Alpha/

├── logs/ # All log files

├── docs/ # All documentation files

├── setup/ # Setup scripts (initial and microservices/modules)

├── scripts/ # Automation and utility scripts

├── configs/ # Configuration files

└── data/ # Data files used in the project

**Script and File Naming Conventions**

* **Master Script:** Create a master script that writes all necessary files to complete tasks in smaller steps. Focus on the master script rather than displaying individual scripts, unless troubleshooting.
* **Naming Conventions:**
  + **Master Scripts:** <Project\_Name>\_<Task>.py
  + **Log Files:** <agentname>\_<filename>.log
  + **Configuration Files:** .env, etc.

**Documentation and Integration**

* **Regular Updates:** Ensure all relevant files, scripts, and documents are regularly pushed to the GitHub repository.
* **GitHub Setup:** Follow the guidelines in C:\Users\ChaseRemmen\ADAPT\Projects\_MAIN\AdaptAI-Alpha\setup\Onboarding\GitHub\_Onboarding.md to set up your GitHub environment. , not working with an existing Repo you will need to create one.

**Logging and Documentation**

* **Logging:**
  + Centralize all logs in AdaptAI-Alpha/logs/.
  + Include timestamps, agent names, and detailed descriptions of actions taken. Include Full Paths with file names in all steps or files worked on.
* **Documentation:**
  + Maintain documentation under AdaptAI-Alpha/docs/.
  + Files to maintain:
    - Master\_<Project\_Name>\_Plan.md: Detailed project plan.
    - Challenges\_and\_Solutions.md: Record of challenges and their solutions.
    - Tasks\_Completed.md: Details on completed tasks.
    - Next\_Steps.md: Outline the upcoming steps for the project.
  + Update markdown files automatically after each significant action, appending instead of overwriting.
* **README.md:** Create a detailed README.md file including:
  + Project Overview
  + Directory Structure Explanation
  + Documentation Details
  + Versioning Information
  + Logging Practices

**Deployment Platform (Common to All Agents)**

1. **Environment Setup:**
   * Ensure you are working within the designated environment (e.g., Docker container, virtual environment).
   * Verify that all necessary dependencies and tools are installed.
2. **Testing:**
   * Conduct initial tests to ensure that your setup is functioning as expected.
   * Log and document all test results, making note of any issues or necessary adjustments.
3. **Handoff:**
   * Ensure that all work is properly handed off to the next agent or team member, with clear documentation on the current state of the project.

**Security Practices**

* **Handling Sensitive Information:**
  + Store API keys and passwords securely using a .env file.
  + API Keys, Endpoints, and Account Information are located at: C:\Users\ChaseRemmen\ADAPT\.env
  + Grant the least privilege required for tasks and review access periodically.
* **Providing Sensitive Information:**
  + Fill in the .env file template with actual API keys, URLs, and other necessary details.
  + Securely share the .env file using encrypted methods accessible only to authorized personnel.

**Environment and Tools**

* **Development Environment:** Be sure all scripts and programs can run in both PowerShell and Ubuntu in WSL 2, which are the current Dev Environments.
* **Conda Environment:** Use Conda Activate chasedev to activate the required environment.

**Version Control and CI/CD**

* **Version Control:** Use Git for version control, following semantic versioning (MAJOR.MINOR.PATCH).
* **CI/CD:** Implement CI/CD pipelines for automated testing, validation, and deployment.

**Documentation Standards**

* **Maintaining README Files:** Include project overview, setup instructions, and usage examples.
* **Inline Code Documentation:** Use comments and docstrings to explain the code.
* **Updating Documentation:** Keep all documentation up to date with project changes using Markdown for easy readability.

**Additional Notes**

* **Communication:**
  + Report task completion in Slack or other designated communication channels.
  + Document your progress in JIRA and update any relevant tickets.
* **Command Execution:**
  + To start Ubuntu in WSL 2 and use sudo for elevated privileges, open your terminal and execute the following:

bash

Copy code

wsl -d Ubuntu && sudo <your\_command>

Replace <your\_command> with the command you need to run with elevated privileges.

* + **Do not use Notepad for anything.**
* **Agent Reporting:**
  + Upon starting, say your name and report what your initial task was, what tasks you were given, and what your current tasks are.

**Conclusion**

By following these guidelines, Team Adapt will ensure a well-organized, efficient, and high-quality project setup and maintenance process. If you need further customization or examples, please reach out to the project lead.