Assignment 1 - Detailed Feedback Report

Student ID: 38966

Assessment Overview

Your submission demonstrates a foundation that shows potential but requires substantial development. While there are fundamental elements in place, there are multiple critical areas requiring attention. You have substantial potential to improve, and the recommendations below outline a clear path forward to strengthen your software engineering practices and meet the assignment requirements.

Areas of Excellence

Your submission demonstrates particular strength in the following areas:

Research Analysis

• Found 1 research document(s)

• Research documentation includes visualizations

• Research documentation provides analysis methodology

Testing Quality

• Found 1 test file(s)

• Good test coverage with 1 test file(s)

• Excellent test count: 15 test functions

Code Documentation

• README.md found (11929 bytes)

- Installation instructions found in README

- Usage examples found in README

Areas for Improvement and Development

The following areas require focused attention and development. Addressing these will significantly strengthen your software engineering practices and overall project quality:

Project Planning

Recommended Actions:

- Create PRD.md with Problem Statement, Functional Requirements, and Success Metrics

- Document architecture in README or ARCHITECTURE.md

Quality Standards

Current Status: Linting configured: pyproject.toml

Recommended Actions:

- Set up CI/CD pipeline (GitHub Actions, GitLab CI, etc.)

- Create dedicated style guide or CONTRIBUTING.md

- Set up pre-commit hooks for automatic quality checks

- Add more quality tools

Extensibility

Current Status: Large files detected (avg 379 lines/file)

Recommended Actions:

- Create plugin/extension system for modularity

- Refactor into modular structure with smaller files

- Document how to extend/customize the system

Costs Pricing

Recommended Actions:

• Create cost analysis document (e.g., COSTS.md, budget.xlsx)

• Document costs, pricing, and budget considerations

• Create budget tracking system/document

Config Security

Current Status: No hardcoded secrets found in production code

Recommended Actions:

• Create .env.example file to document required environment variables

• Create .gitignore file and add .env to it

• Improve .gitignore coverage for better security

Version Management

Current Status: Limited commit history: 2 commits

Recommended Actions:

• Make more commits (target: >10)

• Create PROMPT_BOOK.md to document AI interactions

Priority Action Items

Immediate Focus Areas:

These items require your immediate attention as they represent critical gaps in the current submission:

• Address Project Planning

• Address Costs Pricing

High Priority Enhancements:

These areas would significantly strengthen your submission and should be addressed after the immediate focus items:

• Improve Quality Standards

• Improve Extensibility

• Improve Config Security

• Improve Version Management

Conclusion

Your submission demonstrates potential, but requires substantial development in multiple critical areas. Focus first on the immediate priority items, which address fundamental requirements. Work systematically through each area, seeking examples and best practices. Remember that software engineering is a skill that develops through practice and iteration. Use this feedback as a roadmap for improvement.

Self-Assessment Report

Ollama Chat Application
LLMs and Multi-Agent Systems Course

Group Code:	OmerAndYogever
Students:	Omer Caplan, ID: 208753665 Yogev Cuperman, ID: 207540550
Project Name:	Ollama Chat Application
Repository:	https://github.com/OmerCaplan/AgentsOrchestration_assignment_1
Assessment Date:	November 10, 2025
Self-Grade:	79/100

Category-by-Category Assessment

Category	Weight	My Score	Final Score
Project Documentation	20%	16/20	16
README & Code Documentation	15%	14/15	14
Project Structure & Code Quality	15%	13/15	13
Configuration & Security	10%	9.5/10	9.5
Testing & QA	15%	12.5/15	12.5
Research & Analysis	15%	0/15	0
UI/UX & Extensibility	10%	8.5/10	8.5
TOTAL	**100%**		**79**

Grade Level: 70-79 (■■■ - Good)

With a self-grade of 79, we expect **reasonable and balanced** scrutiny. The evaluators will check adherence to main criteria but will allow room for minor imperfections, as this is a first assignment in the course.

Justification for Self-Assessment

Strengths

The project demonstrates strong software engineering practices with excellent documentation across multiple dimensions. The formal PRD provides clear project requirements and objectives, while the comprehensive README offers detailed installation, configuration, and troubleshooting guidance. The README documentation is particularly strong with step-by-step instructions, multiple platform support, comprehensive troubleshooting section, and clear usage examples, earning it 14/15 points. Code quality is excellent with proper documentation through docstrings and comments, descriptive naming conventions, and clean, maintainable structure with good modular organization earning 13/15 points.

Security and configuration management are exemplary with proper use of .env files, no hardcoded credentials, and clear documentation of all configuration parameters. The testing suite is comprehensive, covering 16 different test scenarios with a 100% success rate.

The project includes three distinct UI implementations - Basic Tkinter, Modern CustomTkinter, and Web Interface. This demonstrates versatility and user-centric design, though the UI/UX score reflects that while multiple implementations exist, there may be room for additional polish in areas such as explicit extensibility mechanisms, accessibility features, or more advanced UI/UX patterns.

The documentation of the AI interaction process adds educational value and transparency to the development methodology, showing the iterative development approach and actual prompts used.

Weaknesses

The most significant deficiency is the complete absence of research and analysis components, worth 15% of the grade. The project does not include experiments comparing different models, parameter sensitivity analysis, performance benchmarking, Jupyter notebooks with data analysis, visualizations of model behavior, or academic references. This represents the primary factor limiting the grade.

Additional gaps include the lack of formal architecture diagrams such as C4 models or UML diagrams, absence of formal test coverage reports, and potential areas for improvement in UI/UX such as explicit plugin systems, advanced accessibility features, or interactive dashboards. The project structure, while good and modular, may not strictly follow all recommended folder organization patterns.

Investment

Significant effort was invested in creating three complete UI implementations, writing comprehensive documentation including a formal PRD, developing a robust test suite with 16 test cases, and properly documenting the AI-assisted development process. The work includes Windows-specific instructions, troubleshooting guides, and multiple installation pathways. This represents substantial work beyond a minimal implementation and demonstrates commitment to delivering a polished, user-ready application.

Innovation

Providing three UI options demonstrates creative thinking and understanding that different users have different preferences and use cases. The inclusion of a modern CustomTkinter interface shows awareness of contemporary UI trends, while the web-based option demonstrates versatility in implementation approaches. The comprehensive documentation of the AI interaction process provides educational value beyond the technical implementation.

Learning

The project demonstrates learning in multiple areas: GUI development with different frameworks, API integration with local LLM services, environment management using UV package manager, security best practices, comprehensive testing methodologies, and professional documentation standards. The progression from basic Tkinter to modern CustomTkinter to web-based interfaces shows skill development throughout the project.

Academic Integrity Declaration

We hereby declare that:

• Our self-assessment is honest and truthful
• We checked our work against all criteria before determining the grade
• We are aware that a high self-grade will lead to more rigorous review
• We accept that the final grade may differ from our self-assessment
• This work is the product of our work (of the team) and we are responsible for all software

Student 1: Omer Caplan (ID: 208753665)

Signature: Omer   **Date:**   November 10, 2025

Student 2: Yogev Cuperman (ID: 207540550)

Signature: Yogev   **Date:**   November 10, 2025