

DETAILED GRADE BREAKDOWN

Assignment 3

Student ID: 63707

Repository: HW3

Assessment Date: 2025-12-04

FINAL SCORE: 17.5 / 100

Performance Tier: Below Standard

Skills Summary

Skill	Score	Status
Ui Ux	3.0/10	✗ Poor
Research Analysis	8.0/10	✓ Excellent
Project Planning	0.0/10	✗ Poor
Testing Quality	0.0/10	✗ Poor
Extensibility	3.0/10	✗ Poor
Quality Standards	0.0/10	✗ Poor
Costs Pricing	0.5/10	✗ Poor
Config Security	2.0/10	✗ Poor
Code Documentation	0.0/10	✗ Poor
Version Management	1.0/10	✗ Poor

Skill: Ui Ux

Score: 3.0/10 points

Status: X Poor

Points Breakdown

Criterion	Max Points	Earned	Status
Screenshots/images (1+)	3.0	0.9	X
Screenshots/images (5+)	3.0	0.9	X
UI documentation	2.0	0.6	X
User guide exists	2.0	0.6	X
TOTAL	10.0	3.0	

What Was Found

- Found 2 image(s)/screenshot(s)

What Was Missing

- All criteria met!

How to Improve (+7.0 points)

1. Add more screenshots (currently 2, need 5+) (+3.0 points)
2. Add UI screenshots to README.md (+2.0 points)
3. Create user guide or UI documentation (+2.0 points)

Skill: Research Analysis

Score: 8.0/10 points

Status: ✓ Excellent

Points Breakdown

Criterion	Max Points	Earned	Status
Jupyter notebooks exist	4.0	3.2	✗
Multiple notebooks (>2)	2.0	1.6	✗
Has visualizations/plots	2.0	1.6	✗
Analysis documentation	2.0	1.6	✗
TOTAL	10.0	8.0	

What Was Found

- Found 1 research document(s)
- Found 6 data file(s) for analysis

What Was Missing

- All criteria met!

How to Improve (+2.0 points)

1. Create more research documentation (currently 1, need >2) (+2.0 points)

Skill: Project Planning

Score: 0.0/10 points

Status: **X** Poor

Points Breakdown

Criterion	Max Points	Earned	Status
PRD.md exists	2.0	0.0	X
ARCHITECTURE.md exists	5.0	0.0	X
Problem Statement	1.0	0.0	X
Functional Requirements	1.5	0.0	X
Success Metrics	0.5	0.0	X
TOTAL	10.0	0.0	

What Was Missing

- All criteria met!

How to Improve (+10.0 points)

1. Create PRD.md with Problem Statement, Functional Requirements, and Success Metrics (+4.0 points)
2. Document architecture in README or ARCHITECTURE.md (+5.0 points)

Skill: Testing Quality

Score: 0.0/10 points

Status: **X** Poor

Points Breakdown

Criterion	Max Points	Earned	Status
Test files exist	3.0	0.0	X
Multiple test files (>3)	2.0	0.0	X
Test framework configured	2.0	0.0	X
Test functions (>10)	3.0	0.0	X
TOTAL	10.0	0.0	

What Was Missing

- All criteria met!

How to Improve (+10.0 points)

1. Create test files for your code (+3.0 points)
2. Create test files for your code (+2.0 points)
3. Configure a test framework (pytest, jest, etc.) (+2.0 points)
4. Write test functions (target: >10) (+3.0 points)

Skill: Extensibility

Score: 3.0/10 points

Status: **X** Poor

Points Breakdown

Criterion	Max Points	Earned	Status
Plugin/extension system	3.0	0.9	X
Modular structure (3+ dirs)	3.0	0.9	X
Interfaces/APIs	2.0	0.6	X
Extension documentation	2.0	0.6	X
TOTAL	10.0	3.0	

What Was Found

What Was Missing

- All criteria met!

How to Improve (+7.0 points)

1. Create plugin/extension system for modularity (+3.0 points)
2. Define interfaces/APIs for extensibility (+2.0 points)
3. Document extension/customization approach (+2.0 points)

Skill: Quality Standards

Score: 0.0/10 points

Status: **X** Poor

Points Breakdown

Criterion	Max Points	Earned	Status
Linting configuration	2.0	0.0	X
CI/CD pipeline	3.0	0.0	X
Code style guide	2.0	0.0	X
Pre-commit hooks	2.0	0.0	X
Project setup file	1.0	0.0	X
TOTAL	10.0	0.0	

What Was Missing

- All criteria met!

How to Improve (+10.0 points)

1. Configure linting tools (pylint, eslint, etc.) (+2.0 points)
2. Set up CI/CD pipeline (GitHub Actions, GitLab CI, etc.) (+3.0 points)
3. Create code style guide or CONTRIBUTING.md (+3.0 points)
4. Set up pre-commit hooks for automatic quality checks (+1.0 point)
5. Configure quality assurance tools (+1.0 point)

Skill: Costs Pricing

Score: 0.5/10 points

Status: **X** Poor

Points Breakdown

Criterion	Max Points	Earned	Status
Cost analysis document	5.0	0.2	X
Cost mentions in docs	3.0	0.2	X
Budget tracking	2.0	0.1	X
TOTAL	10.0	0.5	

What Was Found

What Was Missing

- All criteria met!

How to Improve (+9.5 points)

1. Create cost analysis document (e.g., COSTS.md, budget.xlsx) (+5.0 points)
2. Add detailed cost analysis (+2.5 points)
3. Create budget tracking system/document (+2.0 points)

Skill: Config Security

Score: 2.0/10 points

Status: X Poor

Points Breakdown

Criterion	Max Points	Earned	Status
No hardcoded secrets (CRITICAL)	5.0	1.0	X
.env.example exists	2.0	0.4	X
.gitignore exists	1.0	0.2	X
Uses environment variables	2.0	0.4	X
TOTAL	10.0	2.0	

What Was Found

- No hardcoded secrets found in production code

What Was Missing

- No hardcoded secrets found in production code

How to Improve (+8.0 points)

1. Create .env.example file to document required environment variables (+2.0 points)
2. Create .gitignore file and add .env to it (+2.0 points)
3. Use environment variables for configuration (+2.0 points)
4. Improve .gitignore coverage for better security (+2.0 points)

Skill: Code Documentation

Score: 0.0/10 points

Status: **X** Poor

Points Breakdown

Criterion	Max Points	Earned	Status
README.md >1KB	3.0	0.0	X
Installation instructions	1.0	0.0	X
Usage examples	1.0	0.0	X
Code structure documented	2.0	0.0	X
Python docstrings (>50%)	3.0	0.0	X
TOTAL	10.0	0.0	

What Was Found

What Was Missing

- All criteria met!

How to Improve (+10.0 points)

1. Create README.md with installation and usage instructions (+5.0 points)
2. Document code structure in README or create docs/ directory (+2.0 points)
3. Add docstrings to Python functions and classes (+3.0 points)

Skill: Version Management

Score: 1.0/10 points

Status: ✕ Poor

Points Breakdown

Criterion	Max Points	Earned	Status
Git commits >10	2.0	0.2	✗
Meaningful commit messages	2.0	0.2	✗
PROMPT_BOOK.md exists	5.0	0.5	✗
Branching strategy	1.0	0.1	✗
TOTAL	10.0	1.0	

What Was Found

What Was Missing

- All criteria met!

How to Improve (+9.0 points)

1. Initialize git repository and make regular commits (+2.0 points)
2. Write meaningful commit messages explaining changes (+2.0 points)
3. Create PROMPT_BOOK.md to document AI interactions (+5.0 points)

Overall Assessment Summary

Key Strengths (8+ points)

- Research Analysis: 8.0/10

Critical Gaps (<5 points)

- Ui Ux: 3.0/10
- Project Planning: 0.0/10
- Testing Quality: 0.0/10
- Extensibility: 3.0/10
- Quality Standards: 0.0/10
- Costs Pricing: 0.5/10
- Config Security: 2.0/10
- Code Documentation: 0.0/10
- Version Management: 1.0/10

Recommended Actions

Immediate Priority:

- Address Project Planning (0/10 points)
- Address Testing Quality (0/10 points)
- Address Quality Standards (0/10 points)
- Address Code Documentation (0/10 points)

High Priority:

- Improve Ui Ux (3.0/10 points)
- Improve Extensibility (3.0/10 points)
- Improve Costs Pricing (0.5/10 points)
- Improve Config Security (2.0/10 points)
- Improve Version Management (1.0/10 points)