

# DETAILED GRADE BREAKDOWN

## Assignment 3

Student ID: 63684

Repository: llm-turing-translation

Assessment Date: 2025-12-03

## FINAL SCORE: 40.0 / 100

Performance Tier: Below Standard

## Skills Summary

Skill	Score	Status
Testing Quality	0.0/10	<span style="color: red;">✗ Poor</span>
Ui Ux	7.0/10	<span style="color: green;">✓ Good</span>
Research Analysis	8.0/10	<span style="color: green;">✓ Excellent</span>
Project Planning	5.5/10	<span style="color: orange;">■ Fair</span>
Quality Standards	0.0/10	<span style="color: red;">✗ Poor</span>
Extensibility	3.0/10	<span style="color: red;">✗ Poor</span>
Version Management	4.0/10	<span style="color: red;">■ Fair</span>
Costs Pricing	0.5/10	<span style="color: red;">✗ Poor</span>
Config Security	2.0/10	<span style="color: red;">✗ Poor</span>
Code Documentation	10.0/10	<span style="color: green;">✓ Excellent</span>

# 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	<b>X</b>
Multiple test files (>3)	2.0	0.0	<b>X</b>
Test framework configured	2.0	0.0	<b>X</b>
Test functions (>10)	3.0	0.0	<b>X</b>
<b>TOTAL</b>	<b>10.0</b>	<b>0.0</b>	

## 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: Ui Ux

**Score: 7.0/10 points**

Status: ✓ Good

## Points Breakdown

Criterion	Max Points	Earned	Status
Screenshots/images (1+)	3.0	2.1	✗
Screenshots/images (5+)	3.0	2.1	✗
UI documentation	2.0	1.4	✗
User guide exists	2.0	1.4	✗
<b>TOTAL</b>	<b>10.0</b>	<b>7.0</b>	

## What Was Found

- Found 1 image(s)/screenshot(s)
- UI documentation found: requirements.txt

## What Was Missing

- All criteria met!

## How to Improve (+3.0 points)

1. Add more screenshots (currently 1, need 5+) (+3.0 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	✗
<b>TOTAL</b>	<b>10.0</b>	<b>8.0</b>	

## What Was Found

- Found 1 research document(s)
- Found 1 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: 5.5/10 points

Status: ■ Fair

## Points Breakdown

Criterion	Max Points	Earned	Status
PRD.md exists	2.0	2.0	✓
ARCHITECTURE.md exists	5.0	0.0	✗
Problem Statement	1.0	0.6	✗
Functional Requirements	1.5	0.8	✗
Success Metrics	0.5	0.3	✗
<b>TOTAL</b>	<b>10.0</b>	<b>5.5</b>	

## What Was Found

- PRD.md found
- Functional Requirements found in PRD
- Some architectural information found in documentation

## What Was Missing

- All criteria met!

## How to Improve (+4.5 points)

1. Add Problem Statement section to PRD.md (+1.0 point)
2. Add Success Metrics section to PRD.md (+0.5 points)
3. Document architecture in README or ARCHITECTURE.md (+3.0 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	<b>X</b>
CI/CD pipeline	3.0	0.0	<b>X</b>
Code style guide	2.0	0.0	<b>X</b>
Pre-commit hooks	2.0	0.0	<b>X</b>
Project setup file	1.0	0.0	<b>X</b>
<b>TOTAL</b>	<b>10.0</b>	<b>0.0</b>	

## 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: Extensibility

**Score: 3.0/10 points**

Status: **X** Poor

## Points Breakdown

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

## 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: Version Management

Score: 4.0/10 points

Status: ■ Fair

## Points Breakdown

Criterion	Max Points	Earned	Status
Git commits >10	2.0	0.8	✗
Meaningful commit messages	2.0	0.8	✗
PROMPT_BOOK.md exists	5.0	2.0	✗
Branching strategy	1.0	0.4	✗
<b>TOTAL</b>	<b>10.0</b>	<b>4.0</b>	

## What Was Found

## What Was Missing

- All criteria met!

## How to Improve (+6.0 points)

1. Make more commits to document project evolution (+1.0 point)
2. Create PROMPT\_BOOK.md to document AI interactions (+5.0 points)

# 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	<b>X</b>
Cost mentions in docs	3.0	0.2	<b>X</b>
Budget tracking	2.0	0.1	<b>X</b>
<b>TOTAL</b>	<b>10.0</b>	<b>0.5</b>	

## What Was Found

- API pricing analysis found (excellent for LLM projects)

## 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: ✕ Poor

## Points Breakdown

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

## 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: 10.0/10 points**

Status: ✓ Excellent

## Points Breakdown

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

## What Was Found

- README.md found (27145 bytes)
- Installation instructions found in README
- Usage examples found in README

## What Was Missing

- All criteria met!

# Overall Assessment Summary

## Key Strengths (8+ points)

- Research Analysis: 8.0/10
- Code Documentation: 10.0/10

## Critical Gaps (<5 points)

- Testing Quality: 0.0/10
- Quality Standards: 0.0/10
- Extensibility: 3.0/10
- Version Management: 4.0/10
- Costs Pricing: 0.5/10
- Config Security: 2.0/10

## Recommended Actions

### Immediate Priority:

- Address Testing Quality (0/10 points)
- Address Quality Standards (0/10 points)

### High Priority:

- Improve Extensibility (3.0/10 points)
- Improve Version Management (4.0/10 points)
- Improve Costs Pricing (0.5/10 points)
- Improve Config Security (2.0/10 points)