**PROMPT:**  
@UglyComplexMethod.cs

Analyse the source code properply

- Provide 3 specific improvement suggestions

METRICS:

- Calculate complexity score for the function

- Identify functions over 20 lines

- Find duplicate code patterns

- Count nested conditions

ISSUES:

- Security vulnerabilities

- Performance bottlenecks

- Maintainability problems

AI Driven Development

- Testing gaps

REPORT:

- Prioritize issues by impact

- Suggest improvement roadmap

- Estimate effort for each fix

**AI Suggestion:**

Key Improvements:

Reduced Complexity:

Original: 44 lines with multiple responsibilities

Refactored: Main method is now only 5 lines with clear separation of concerns

Single Responsibility Principle:

GetRelatedEntities(): Handles entity retrieval

GetShortenedDescription(): Manages description shortening logic

GetFormattedDescriptionByType(): Routes to specific formatters

Individual formatter methods for each type (Crew, Expense, Equipment)

Improved Readability:

Clear method names that describe their purpose

Eliminated nested if-else chains

Better variable naming (crewSizeSuffix vs maybeCrewSize)

Performance Optimizations:

Used string.Equals() with StringComparison.OrdinalIgnoreCase instead of .ToLower()

Early return for empty parent descriptions

More efficient string operations

Maintainability:

Each method is focused and testable

RelatedEntities class encapsulates related data

Clear separation between data retrieval, processing, and formatting

Code Quality:

Consistent formatting and spacing

Better parameter naming

Eliminated magic strings and improved string handling

Benefits:

Testability: Each method can be unit tested independently

Maintainability: Changes to one formatting type don't affect others

Readability: The main method clearly shows the high-level flow

Performance: More efficient string comparisons and operations

Extensibility: Easy to add new entity types or modify existing ones

The refactored code maintains the exact same functionality while being much more maintainable and easier to understand.