AI-Assisted Development Strategy for EM-Tools

# 1. Overview

This document outlines a strategy for using Cursor and CodiumAI in tandem to develop, test, and maintain EM-Tools. The strategy reflects the modular, milestone-based nature of EM-Tools, and introduces a phased workflow that leverages Cursor for logic creation and refactoring, and CodiumAI for automated test generation and validation.

# 2. Role of Each Tool

## 2.1 Cursor

Cursor is an AI-native IDE that supports code generation, editing, and refactoring. It is best used during the initial development and ongoing refinement of EM-Tools logic modules.

* Use Cursor to:
* • Scaffold new modules and milestone scripts (e.g., generate\_financials.py, normalize\_inputs.py)
* • Refactor long or repetitive code into reusable functions or modules
* • Write and edit inline documentation, comments, and docstrings
* • Debug modules using conversational AI

## 2.2 CodiumAI

CodiumAI (Qodana) is an AI tool specialized in generating test cases and improving QA coverage. It should be used after logic development to harden and validate the code.

* Use CodiumAI to:
* • Generate unit and integration tests for newly created or refactored modules
* • Improve confidence in milestone completion before packaging or syncing
* • Detect edge cases and validation gaps early

# 3. Phased Workflow for EM-Tools Development

1. Phase 1: Drafting and Logic Development (Cursor)

* • Create new modules or rewrite legacy ones using Cursor.
* • Refactor and document the logic iteratively with AI assistance.

1. Phase 2: Testing and Validation (CodiumAI)

* • Use CodiumAI to generate tests for each major function or script.
* • Validate using run\_all\_tests.py or custom test runners.

1. Phase 3: Packaging and Milestone Completion

* • Once tests pass, bundle files into the appropriate milestone folder (e.g., v0.5/).
* • Finalize documentation and prepare for GitHub/Dropbox sync.

# 4. Suggested Handoff Process

For collaborative development, the following handoff process ensures smooth transitions between Cursor-based development and CodiumAI-based testing:

1. 1. Developer completes initial version using Cursor.
2. 2. Developer saves and stages file in shared folder (e.g., EM-Tools/Current\_Work/).
3. 3. Test engineer or validator opens the file in VS Code with CodiumAI.
4. 4. Generated tests are reviewed, finalized, and saved to `tests/`.
5. 5. A `run\_all\_tests.py` summary is generated to confirm passing state.
6. 6. Milestone is packaged and archived in versioned folder (e.g., v0.5/).