# **Technical Design Specification: Stateful Coding Agent**

Version: 8.0

Date: 18 June 2025

Status: Proposed (Based on TRR v25.0)

1. Architectural Overview

This specification details the implementation of a stateful Coding Agent. This is achieved by refactoring the code\_execution\_node to use an in-memory file cache and adding a finalize\_coding\_report node to produce a structured JSON summary of the work performed.

2. State Management Modifications (agent\_core\_v8.py)

The GraphState TypedDict requires one new key to store the final coding report.

class GraphState(TypedDict):

# ... (all existing keys)

# NEW KEY for v8 implementation

coding\_execution\_report: Optional[Dict]

3. Artifact Specification: coding\_execution\_report.json

The Coding Agent will now produce this final artifact, which contains the complete proposed code.

{

"metadata": {

"report\_version": "8.0",

"generation\_date": "YYYY-MM-DD",

"source\_design\_spec": "<Version of the design spec>"

},

"summary": "Summary of changes implemented from the design spec.",

"final\_code\_artifacts": [

{

"file\_path": "src/supervisor\_v8.py",

"content": "<The complete, new content of the supervisor\_v8.py file>"

},

{

"file\_path": "src/agent\_core\_v8.py",

"content": "<The complete, new content of the agent\_core\_v8.py file>"

}

]

}

4. Graph Modifications (agent\_core\_v8.py)

The build\_coding\_agent\_graph will be updated:

1. code\_execution\_node (Refactored): This node will be modified to maintain an in-memory dictionary of file contents. It will read from this dictionary first before reading from the disk. After each "Think" step, it will update the dictionary with the newly generated code. This node will now also pass the in-memory cache to the next node via the agent state.
2. finalize\_coding\_report (New Node): This node is added after the code\_execution\_node. It takes the in-memory cache of file contents and uses a new prompt, create\_coding\_report\_prompt, to generate the coding\_execution\_report.json.
3. Graph Edges: The graph will now flow from generate\_coding\_plan -> code\_execution\_node -> finalize\_coding\_report.

5. New Prompt Specification (prompts\_v8.yaml)

A new prompt is required to generate the final JSON report.

# --- Prompt for Coding Agent Finalization ---

create\_coding\_report\_prompt: >

You are a technical writer summarizing a coding session. Based on the provided final file contents, create a structured JSON report.

\*\*CRITICAL:\*\* Your output must be a single, valid JSON object.

JSON Schema:

{{

"metadata": { ... },

"summary": "<A brief, one-sentence summary of the changes made.>",

"final\_code\_artifacts": [

{{

"file\_path": "<file\_path>",

"content": "<The full, escaped string content of the file>"

}}

]

}}

### FINAL FILE CONTENTS:

{final\_file\_contents\_json}

This updated design will produce a fully functional and auditable Coding Agent.