Nova AI Coordinator System Refactor Summary

Overview

This document summarizes the full refactor of the Nova AI Coordinator system. The goal was to clean up redundant code, implement a unified verbosity toggle, improve session context propagation across agents, and refine task classification logic using LLMs.
The system now fully supports verbose toggling, context-passing across agents, modular logging, and fallback logic for general queries.

main.py

- Added `verbose_mode = True/False` toggle to control system-wide verbosity.
- Injected session context before agent and sub-agent calls.
- Added `summarize_context()` to only show useful context (e.g. model, location).
- Suppressed all warnings with `warnings.filterwarnings("ignore")` when verbose=False.
- Improved final summary output and session file confirmation.

nova.py

- Added `self.verbose` in constructor.
- All print/logging operations now wrapped with `if self.verbose:`.
- Intent splitting enhanced using regex if LLM fails (e.g., on commas, periods).
- General question fallback categorization improved:
  - If LLM classifies an intent as Uncategorized or similar, and the intent contains "what is", "capital", "how many", etc., it is routed to `answer_general_question`.

emil.py, lola.py, ivan.py

- All agents were refactored to support `verbose` in constructor.
- Logging and task debug printing guarded with `if self.verbose:`.
- Session context is correctly accessed and used to retrieve shared data.
- Emil pushes model metadata to context.
- Lola consumes context to generate reports.

csv_function_mapper.py

- Added `verbose=True` flag to constructor.
- All warning and success `print()` statements wrapped with `if self.verbose:`.
- Controlled verbosity via `FunctionMapLoader(verbose=verbose_mode)` in `main.py`.

function_logger.py

- Updated decorator `log_function_call()` to only print `FUNCTION CALL:` when `self.verbose=True`.
- Automatically ignores top-level function calls or static methods where no `self` exists.

functions_registery.py

- Removed hardcoded `print()` statements for PLEXOS library loading.
- Optional fallback to environment variable `VERBOSE_MODE` to control prints.

Nova AI Coordinator System Refactor Summary

Flow Mapping: Prompt to Task Chain

Prompt -> Nova.create_task_list_from_prompt_async(prompt)

- Split intents via LLM or punctuation fallback
- Categorize each intent (e.g. 'answer_general_question', 'process_emil_request')
- Assign to agents: Nova, Emil, Lola, Ivan
  - Inject session_context with KB values
  - Agent processes and appends result
- Final report saved and session archived.