Danielle Chan

Modern Software Concepts in Python

9/28/25

Module 5 Assignment: Dependencies

The dependency graph for ‘run.py’ shows that it is the central controller of the application, coordinating between the web interface, database operations, and data processing modules. The primary dependencies are on Flask, which handles the web routes and template rendering, and psycopg, which provides secure database connections and query composition. It also depends on subprocess to run external scripts for scraping, cleaning, and LLM-based data standardization. Internally, ‘run.py’ relies on ‘append_data’ to insert new applicant records into the database and ‘sql_helpers’ to manage reusable SQL queries, which helps keep the codebase organized and consistent. Through these modules, it indirectly depends on json for reading standardized data files and on PostgreSQL as the backend database. The graph has a star-like structure, with run.py at the center connected to specialized modules and libraries, showing a clear separation of concerns. This architecture avoids circular dependencies and ensures that each component can be maintained or updated independently.