

## **5-2 Milestone Four: Enhancement Three: Databases**

Kylee Middleton

Southern New Hampshire University

CS-499 12001: Computer Science Capstone

Jeff Phillips

February 8, 2026

The artifact that I chose for this milestone is my Android-based Inventory Management application that I had originally created for my CS- 360 Mobile Architecture and Programming class. This application allows users to create an account, log in securely, and manage a list of inventory items stored in a local SQLite database. Its core functionality is to allow users to add new items, edit item quantities, delete items, and view inventory details in a grid-based user interface. For Milestone Four of CS-499, I enhanced the artifact focusing on databases and security.

I selected this artifact because it demonstrates practical database skills in a real application. It shows that I can design and implement database-backed features that improve data portability, resilience, and security and go beyond basic CRUD. This project also supports my career goal of managing STEM projects because it reflects my ability to translate requirements into working solutions and communicate technical decisions clearly.

For the database enhancement, I improved the application beyond basic CRUD by implementing an inventory import or export using a structured JSON file format. Users can now export their inventory data to a JSON file and later import that file to restore the inventory record. This requires designing a clear, readable data format and implementing a workflow that is safe and user-friendly through Android's Storage Access Framework, so users can choose where backups are stored. On the database side, the import process uses transactions to ensure data integrity and prevent partial restores if an import fails. Unlike earlier versions of this application, the database is no longer simply a storage layer but now supports an end-to-end backup and restore feature that makes the application more user friendly and aligned with industry expectations.

I also strengthened how sensitive user data is handled in the database. For credentials, the application uses secure, one-way password storage instead of storing plaintext passwords. For sensitive settings like SMS alerts, the application encrypts the saved phone number before storing it in SQLite and decrypts it only when it needs to be displayed or used. These changes help to reduce unnecessary exposure of sensitive data and demonstrate a security first mindset that aligns with industry standards and best practices.

In my proposal, I had planned for my database enhancement to expand beyond basic CRUD by adding export and import functionality and protecting any sensitive user data that is stored in the database. This milestone meets that plan by implementing JSON export and import for inventory items and by ensuring that sensitive values are not stored in plaintext. These enhancements also show progress in many of the course outcomes including using well founded and innovative techniques, skills, and tools in computing practices, developing a security mindset that anticipates vulnerabilities and protects data, and designing solutions using appropriate standards and practices while managing trade-offs.

Enhancing this artifact has pushed me to think of the database as an evolving component rather than a one-time setup. Adding the backup and restore functionality required careful planning and decisions about what data would be included, how to structure it, and how to ensure imports do not corrupt or partially overwrite existing data. The biggest challenge that I faced throughout my work on this enhancement was balancing usability and security. For example, the application supports SMS related features that require a phone number but storing that number in plaintext would create a security risk that could be avoided. Solving that problem required integrating encryption into the database workflow while also keeping the settings screen simple and functional for users.

Overall, working on this artifact helped me to think more like a security-conscious developer. I treated user data as something that should be protected by design rather than as an afterthought. This artifact also helped me to build confidence in designing and implementing database features that are common in real software products, such as backup and restore workflows, transaction-safe imports, and the safe handling of sensitive data.