**Artifact 3: Database Management**

The Grazioso Salvare Dashboard is a web-based application designed to interact with a MongoDB database to display animal rescue data. The original implementation used basic MongoDB operations for data retrieval, which were sufficient for small datasets but became inefficient as the dataset grew. Additionally, the system lacked robust security measures and did not have a proper backup and recovery strategy. This artifact was selected to showcase my proficiency in database management by enhancing the dashboard's interaction with MongoDB to improve data retrieval efficiency, security, and integrity.

**Original Implementation:** The dashboard's original implementation relied on simple MongoDB queries for data retrieval. While functional, this approach became increasingly inefficient as the dataset expanded. The original code also did not implement security features such as role-based access control (RBAC) or encryption, nor did it have a backup and recovery strategy, making the system vulnerable to data loss and unauthorized access.

**Enhancements Made:** To address these issues, I implemented several key enhancements focused on database management:

1. **Advanced Query Techniques:** I utilized MongoDB’s aggregation framework to perform complex data manipulations directly within the database. This reduced the application's processing load and improved the efficiency of data retrieval operations.
2. **Database Indexing:** I created indexes on frequently queried fields such as rescue\_type and location. This significantly sped up data retrieval times, especially for large datasets.
3. **Data Security Enhancements:** I added role-based access control (RBAC) to manage user permissions and implemented encryption for sensitive data fields, ensuring only authorized users could access certain information.
4. **Backup and Recovery:** I set up automated backup and recovery procedures using MongoDB’s mongodump and mongorestore utilities. This ensures data integrity and availability in case of failures or data corruption.

**Technical Explanation:** The advanced query techniques were implemented using MongoDB's aggregation framework, allowing for efficient data processing within the database. Indexes were created on key fields by modifying the MongoDB schema, significantly improving query performance. Security enhancements, including RBAC and encryption, were configured through MongoDB Compass and the server's configuration files. Backup and recovery processes were automated using scripts that regularly execute mongodump and mongorestore commands to create and restore backups.

**Outcomes Achieved:** These enhancements align with the course outcomes related to database management. By optimizing the interaction between the dashboard and MongoDB, I demonstrated my ability to implement advanced database techniques to improve a software application's efficiency, security, and reliability. The improvements showcase my proficiency in managing databases to ensure data integrity, security, and availability.

The database management enhancements made to the Grazioso Salvare Dashboard have significantly improved its efficiency, security, and reliability. The dashboard can now handle larger datasets with faster retrieval times while ensuring that data is secure and protected against unauthorized access and potential data loss. This artifact is a strong example of my ability to practically apply advanced database management techniques, demonstrating my skills in optimizing software for better performance and security.