Final Report:

1. Approach and Meeting Requirements

My partner and I approached this project by thoroughly analyzing the requirements and systematically tackling each aspect. Here’s how we managed the project:  
  
1. Planning:  
- Created a timeline to allocate specific tasks between the two of us.  
- Broke the project into smaller modules: frontend, backend, and database integration.  
  
2. Division of Responsibilities:  
- Tyler was responsible for most backend elements, including setting up the database and connecting it to the frontend.  
- Vinny focused on incorporating the frontend elements and refining the user interface to ensure usability and aesthetic appeal.  
  
3. Iterative Development:  
- Implemented the project incrementally, testing each feature before moving to the next.  
- Followed an agile approach where we reviewed our progress weekly and adjusted tasks as needed.  
  
4. Meeting Requirements:  
- We ensured that the project met all specified requirements, including:  
 - A fully functional user interface (UI) that allows users to interact with the system.  
 - Proper database integration for storing and retrieving PC part details.  
 - Core functionalities like filtering, sorting, and saving custom configurations.  
  
By maintaining effective communication and efficiently dividing tasks, we successfully completed the project on time.

2. Tools and Languages Used

We used the following tools and technologies to develop the project:  
  
Languages:  
- Java: For the backend logic and creating the user interface using JavaFX.  
- SQL: For writing database queries and integrating with MySQL.  
  
Frameworks & Libraries:  
- JavaFX: For designing and building the graphical user interface.  
- MySQL Connector: For connecting the Java application to the database.

3. Successes, Pitfalls, Lessons Learned, and Improvements

Successes:  
- Successfully developed a functional application that met all project requirements.  
- Integrated the database seamlessly and implemented filtering functionality without major issues.  
- Maintained consistent progress through effective communication and teamwork.  
  
Pitfalls:  
- Faced initial challenges in aligning the frontend design with backend functionality.  
- Encountered rendering issues with JavaFX on macOS, which required additional debugging time.  
- Managing time was occasionally challenging due to our conflicting schedules.  
  
Lessons Learned:  
- Regular testing throughout the development process helps avoid last-minute debugging.  
- Clear communication and proper task division are vital in collaborative projects.  
- Understanding the limitations and nuances of the tools and platforms used (e.g., JavaFX on macOS) is crucial to avoid delays.  
  
What to Do Differently Next Time:  
- Spend more time planning and designing the user interface before diving into development.  
- Allocate more time for testing and debugging on different systems to ensure cross-platform compatibility.  
- Use a shared repository with clear and consistent commit messages to better track changes and contributions.