Evaluation Criteria

Object-Oriented Design (UML Diagram) (4 marks):

Inheritance: Proper use of inheritance hierarchy to model relationships between classes.

Dynamic Method Dispatch: Demonstrated usage of dynamic method dispatch to invoke overridden methods at runtime.

Aggregation and/or Composition: Clear demonstration of aggregating or composing objects to form more complex structures.

Overriding Methods from Object Class: Proper implementation of equals(Object obj) and toString() methods in at least one class for meaningful object comparison and string representation.

Custom Database System (4 marks):

Database Creation: Proper implementation of a custom database system within the application.

Data Storage: Efficient storage and retrieval of relevant information using the custom database.

Data Manipulation: Ability to perform CRUD (Create, Read, Update, Delete) operations on the data stored in the custom database.

Integration with Application: Seamless integration of the custom database system with the existing application functionalities.

Error Handling: Proper handling of errors and edge cases related to database operations to ensure the robustness of the system.

Graphical User Interface (GUI) with Event Handling (4 marks):

GUI Design: Well-designed graphical user interface that offers intuitive interaction with the system's functionalities.

Event Handling: Effective handling of user interactions through event-driven programming paradigms.

Integration with Existing System: Seamless integration of GUI components with the existing system functionalities.

Code Quality and Documentation (2 marks):

Readability and Maintainability: Clear and well-structured code with appropriate comments to aid understanding and future modifications.

Consistency: Consistent naming conventions, coding style, and formatting throughout the project.

Documentation: Comprehensive documentation including inline comments within the code and a separate report detailing the system's design, usage, and sample outputs.

Functionality and Correctness (2 marks):

System Functionality: Full implementation of the desired functionalities as specified in the project requirements.

Error Handling: Proper handling of edge cases and error conditions to ensure the robustness of the system.

Accuracy: Correctness of the system's behavior in different scenarios, including input validation and output generation.

Creativity and Innovation (2 marks):

Originality: Implementation of unique features or creative approaches to solving specific problems within the chosen application domain.

Efficiency: Consideration of performance optimizations and resource utilization to enhance the system's efficiency where applicable.

Testing and Validation (2 marks):

Test Coverage: Adequate test coverage to ensure the correctness of the implemented functionalities.

Test Cases: Creation of meaningful test cases to validate different aspects of the system's behavior.

Validation: Demonstrated validation of the system through sample inputs and outputs, including edge cases.

Final Deliverables:

- Completeness: Submission of all required deliverables including source code, documentation, sample outputs, and any additional files or resources.
- Organization: Proper organization of files and documentation in a clear and understandable manner.
- Execution: Successful execution of the provided main class to demonstrate the usage of the system.

*** Total Marks: 20**

*** Bonus for early submissions (mid-MAY or before)!**

*** Penalties for late submissions!**