A Design Proposal for Object-Oriented Software Supporting Humanoid Robot Operations in Educational Settings

Student Name: Lauren Pechey

Module: Object Oriented Programming

Degree: Masters of Computer Science, University of Essex Online

Module Tutor: Stelios Sotiriadis

Date: 14/09/2025

Introduction and Aim

This report proposes an object-oriented software system to support the operation of a humanoid robot in a school classroom. The software enables the robot to perform three operations: delivering materials, monitoring classroom temperature, and greeting/supporting students with special needs. Teachers interact with the system through a command-line interface (CLI), while the back-end simulates classroom sensor data and manages robot state transitions for safe and realistic operation. The design utilises object-oriented programming (OOP) concepts, data structures (lists, stacks, queues), and UML diagrams (activity, class, sequence, and state) to illustrate behaviour and interactions.

Background Research

Humanoid robots are designed to resemble and mimic human behaviour in social, physical, and environmental contexts (Ackerman, 2023; Chalmers et al., 2022). They are increasingly deployed in education to augment teaching, engage learners, and provide targeted support for students with special needs (Chalmers et al., 2022). Research on human-robot interaction (HRI) and collaborative robots highlights their ability to perform physical tasks, monitor environments, and interact socially in ways that complement teaching (Mishra et al., 2021). Classrooms provide structured, socially rich environments where humanoid robots can influence learning outcomes, reduce repetitive teacher workload, and offer consistent, individualised support to all students (Ekström & Pareto, 2022).

Keywords such as “robot”, “human-robot interaction”, “collaborative robot”, and “educational robotics” guided the research, ensuring both operational capabilities and potential student support were considered (Mulko, 2023; Mukherjee et al., 2022).

Proposed Robot Operations

The software supports three main operations:

1. **Deliver Materials:** The robot carries small objects, such as books or worksheets, to students, improving workflow efficiency (Ekström & Pareto, 2022).

2. **Monitor Classroom Conditions:** The robot tracks temperature and reports anomalies to ensure a comfortable learning environment, especially for sensitive students (Papakostas et al., 2021).

3. **Greet/Support Students:** The robot provides predictable social interaction and encouragement, aiding engagement and motivation for students with autism, ADHD, or social anxiety, while logging interactions for teacher review (Qidwai et al., 2020).

Objected-Oriented Design

The system applies OOP principles to ensure modularity, scalability, and maintainability (Rumbaugh et al., 1999). The central class, *RobotController*, coordinates operations—material delivery, temperature monitoring, and student interactions—through supporting classes: *TaskManager* (manages delivery queue), *SensorModule* (simulates temperature), and *InteractionModule* (handles communication and logs interactions), each with distinct attributes and methods to maintain clear responsibility separation.

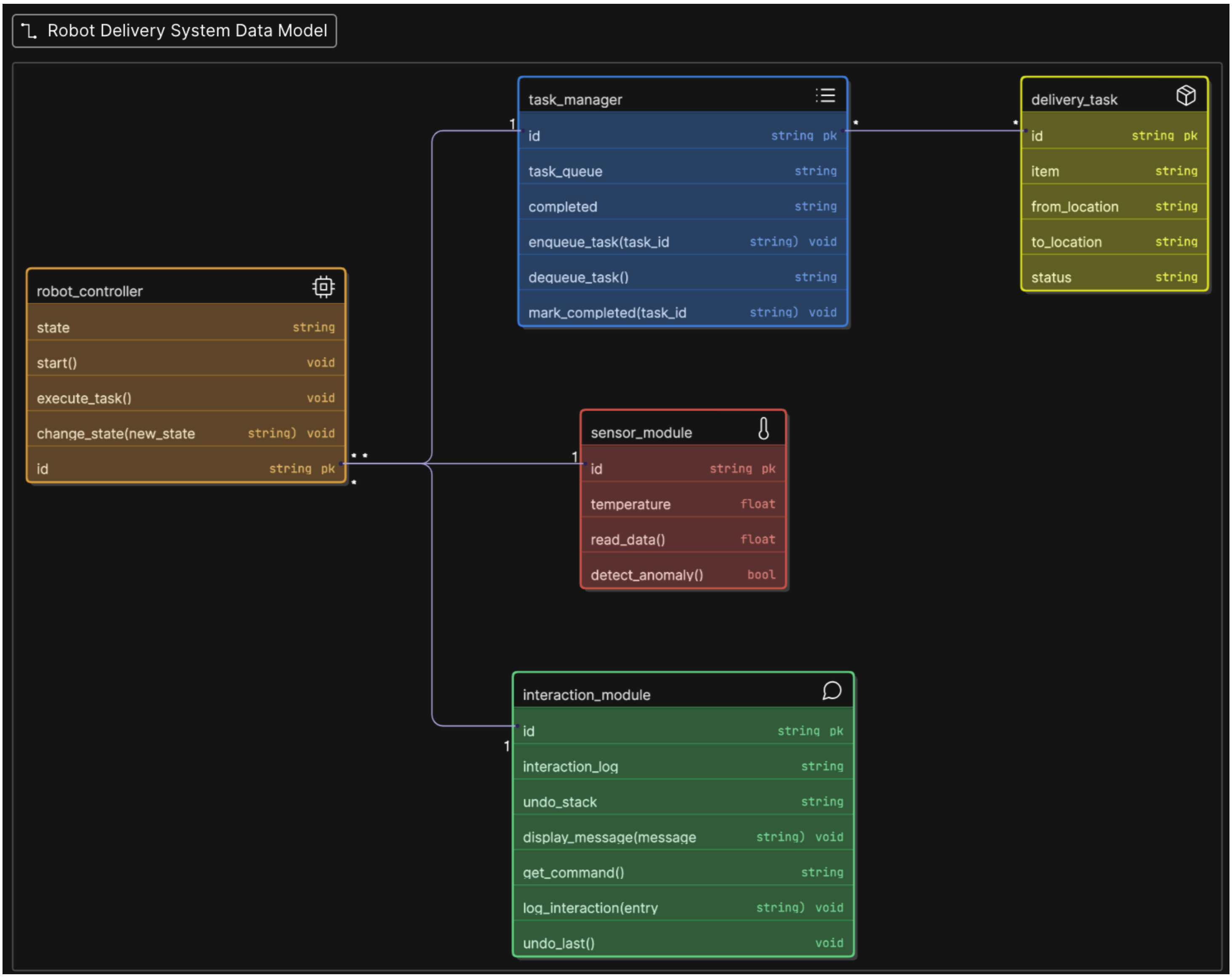
Encapsulation protects class data, e.g., *TemperatureSensor* retains its temperature attribute and methods (Bennett et al., 2021). Abstraction exposes high-level methods such as *monitor_environment*() and *deliver_material*() for teacher interaction (Singh et al., 2021). Inheritance allows a generic *Sensor* class to be extended by *TemperatureSensor*, and polymorphism enables methods like *report_issue*() to behave differently per sensor type while greetings and reminders share a common interface.

Class relationships reinforce modularity: *RobotController* associates with *TaskManager*, aggregates *SensorModule* data, and composes *InteractionModule*, supporting maintainability and future classroom extensions.

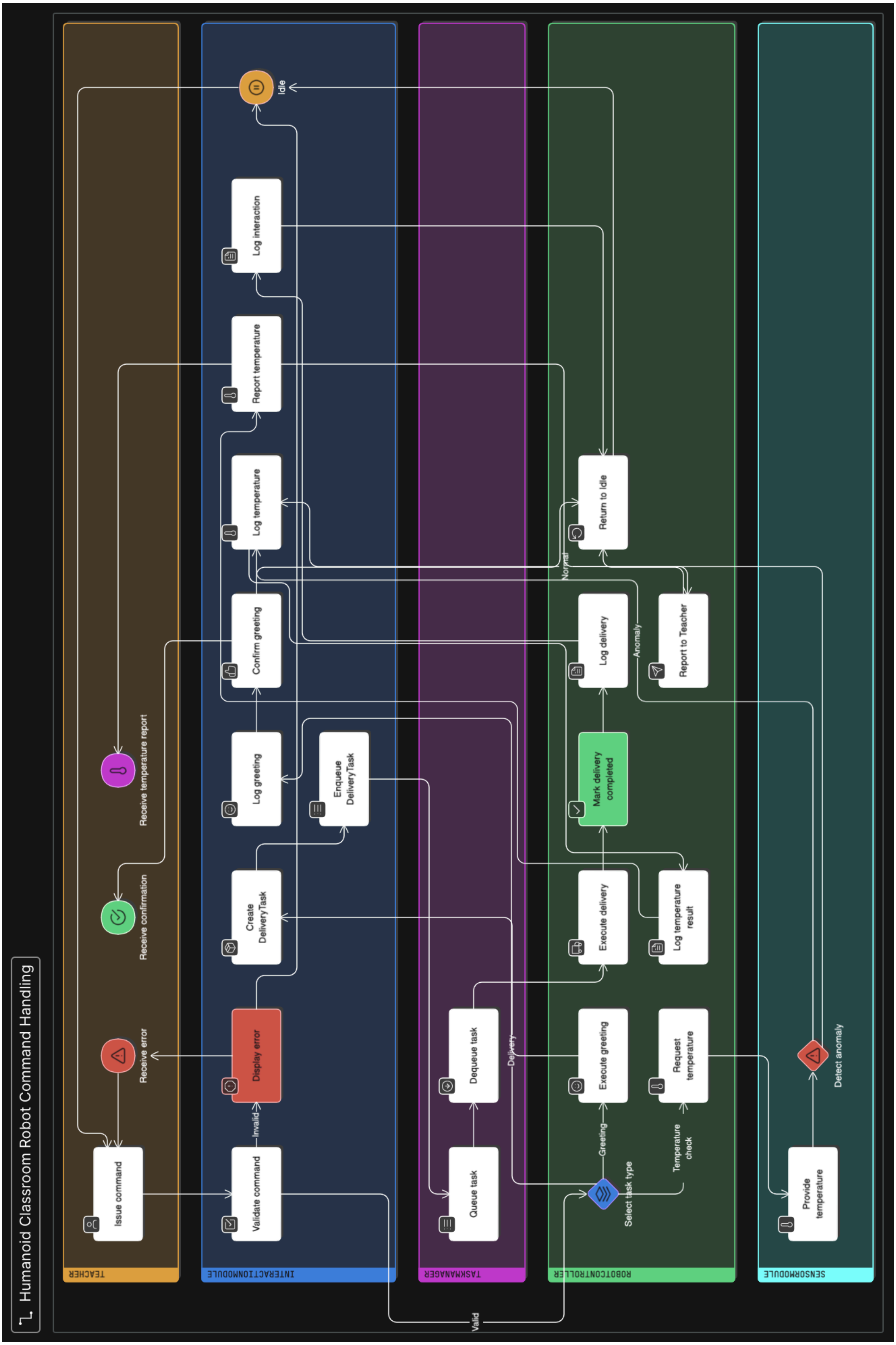
UML Diagrams

Four UML diagrams, each on a separate page, illustrate the humanoid robot system's structure, behaviour, and interactions, supporting clear design and communication.

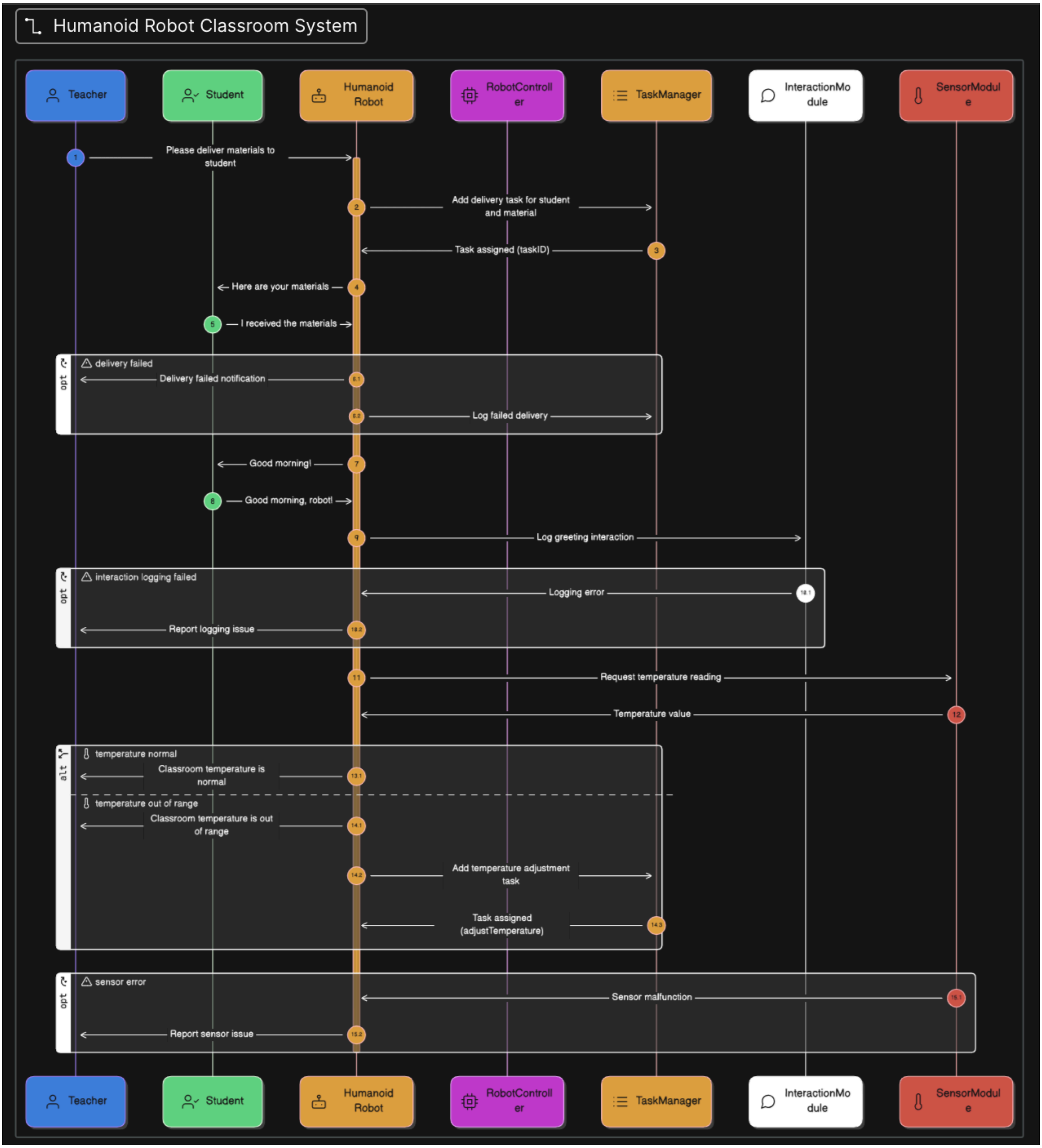
Class Diagram – Illustrates classes, their attributes, methods, and relationships.



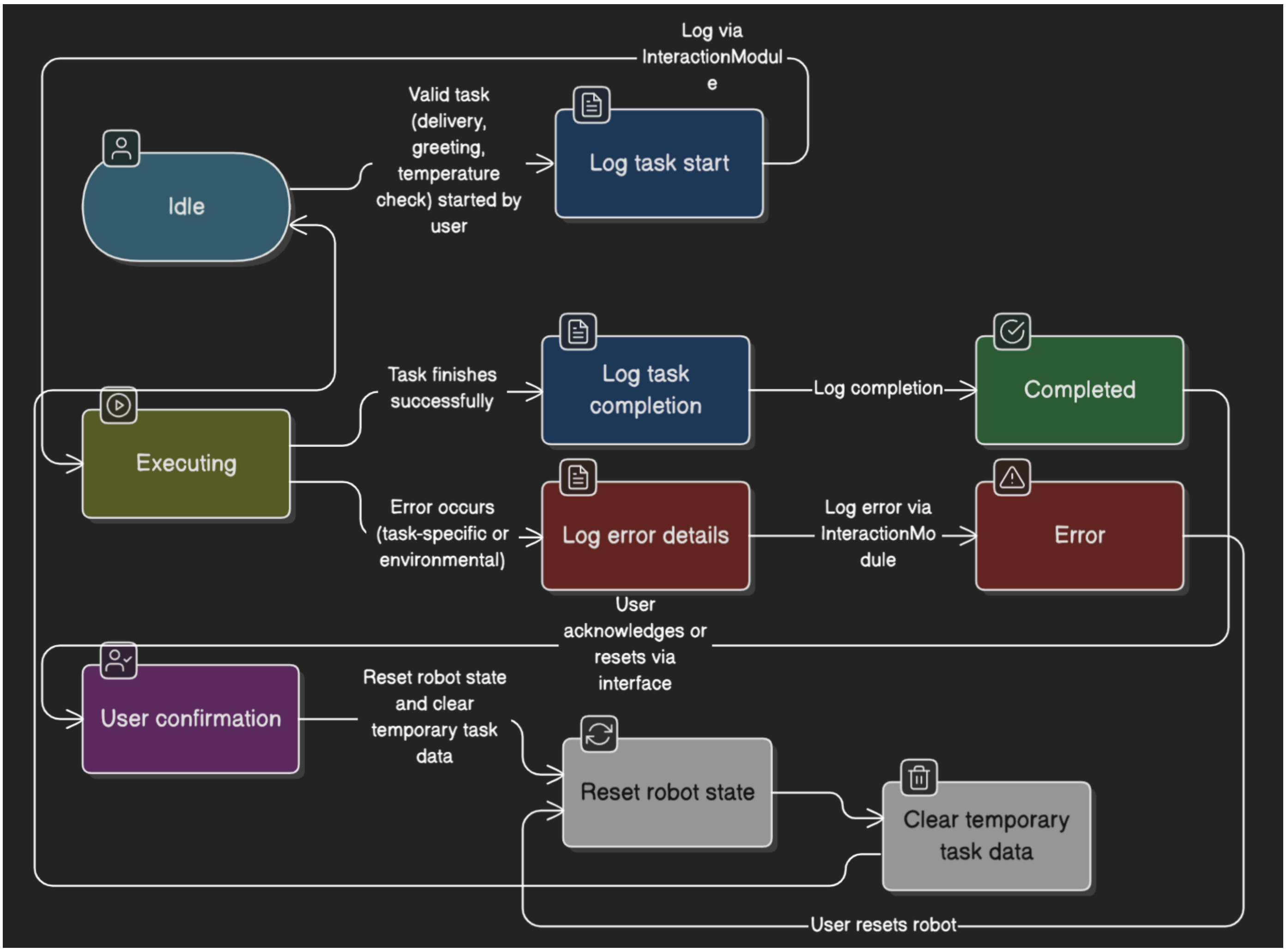
Activity Diagram – Illustrates the workflow of the humanoid robot performing tasks.



Sequence Diagram – Shows how the teacher issues a command, the system interprets it, and the robot executes the operation.



State Transition Diagram – Models robot states such as Idle → Executing → Completed → Idle, with transitions triggered by events.



Data Structures and Search Algorithms

The system uses key data structures to support efficient task management and information retention (Lott & Phillips, 2021). A list stores historical operations and temperature readings, enabling teachers to review past logs and identify patterns among students with special needs. A stack tracks recent actions, allowing undo functionality for the last greeting or reminder to maintain supportive interactions. A queue organizes pending delivery tasks in the order assigned, ensuring fair and efficient material distribution (Bennett et al., 2021).

Search algorithms enhance data processing: linear search scans environmental readings sequentially to detect anomalies in real-time datasets, while binary search efficiently locates specific values in sorted historical logs. Together, these structures and algorithms enable accurate monitoring, effective task management, and reliable, responsive support for all students (Li et al., 2022).

Rationale & Conclusion

The rationale explains the reasoning behind the key design decisions in this humanoid robot system, highlighting both technical and educational considerations.

Choice of Operations – Firstly, the three operations (delivering materials, monitoring classroom temperature, and greeting/supporting students) were selected to address essential classroom functions: physical, environmental, and social (Mukherjee et al., 2022). Delivering materials supports efficient workflow, reducing repetitive teacher tasks and ensuring all students receive necessary resources (Mishra et al., 2021). Monitoring temperature maintains a comfortable and safe environment, which is particularly important for students sensitive to environmental changes (Qidwai et al., 2020). Greeting and supporting students provides consistent, predictable social interaction, fostering inclusivity, engagement, and motivation for all learners, particularly those with autism, ADHD, or social anxiety (Papakostas et al., 2021). Each operation was chosen to balance pedagogical benefits with practical classroom needs.

Object-Oriented Design – The system's OOP-based structure ensures modularity, scalability, and maintainability (Rumbaugh et al., 1999). Separating responsibilities into distinct classes allows future extensions, such as adding new sensors or interaction types, without disrupting existing functionality (Singh et al., 2021). Encapsulation protects internal data, abstraction provides simple interfaces for teacher interaction, and relationships between classes ensure efficient communication (Matus & Cano, 2025). This approach guarantees a robust, flexible architecture capable of supporting both current operations and future classroom scenarios.

Human-Robot Interaction via CLI – The CLI provides a controlled, straightforward interface for teachers to issue commands, check status, and interact with the robot (see below example). By mapping commands directly to system methods, the interface simplifies operation, reduces potential errors, and allows teachers to focus on instructional activities rather than system complexity (Matus & Cano, 2025).

Pseudocode example for CLI

 # Command-line interface (CLI) interaction examples

> deliver_material('worksheet1', 'StudentA')

RobotController: TaskManager queued delivery of worksheet1 to StudentA

RobotController: Delivery completed successfully

> monitor_temperature()

RobotController: Current classroom temperature is 22°C

RobotController: All readings within safe range

> greet_student('StudentB')

InteractionModule: Hello, StudentB! Keep up the great work today.

InteractionModule: Greeting logged for teacher review

Simulation and Data Management – The back-end simulation generates classroom data to represent real-world variability, with retained historical records enabling trend monitoring and anomaly detection (Lott & Phillips, 2021). Using lists, stacks, and queues ensures that operations, interactions, and task management are efficiently organised and easily retrievable (Lott & Phillips, 2021). These structures, combined with search algorithms, support reliable decision-making and monitoring while ensuring that all students are effectively supported.

Support for Students – The robot's greeting and reminder functionality provides structured, low-pressure interaction and positive reinforcement. Retaining logged interactions allows teachers to review patterns, tailor support, and ensure inclusive engagement (Matus & Cano, 2025). This data-driven approach enhances both educational outcomes and student wellbeing.

Limitations and Future Improvements – The current system focuses on simulation and CLI interaction, which does not fully replicate natural human communication or integrate with hardware. Future enhancements could include graphical or voice-based interfaces, adaptive-learning capabilities, and real-time sensor integration. These improvements would increase realism, engagement, and responsiveness while maintaining the core modular and maintainable OOP design.

References:

Ackerman, E. (2023) Humanoid robots are getting to work. *IEEE Spectrum*. Available at: https://spectrum.ieee.org/humanoid-robots (Accessed: 13 September 2025)

Bennett, S., McRobb, S., & Farmer, R. (2021) *Object-Oriented Systems Analysis and Design Using UML.* London: McGraw-Hill Higher Education.

Chalmers, C., Keane, T., Boden, M., & Williams, M. (2022) Humanoid robots go to school. *Education and Information Technologies 27(1):* 7563–7581. DOI: https://doi.org/10.1007/s10639-022-10913-z

Ekström, S., & Pareto, L. (2022) The dual role of humanoid robots in education: As didactic tools and social actors. *Education and Information Technologies* 27(1):12609–12644. DOI: https://doi.org/10.1007/s10639-022-11132-2

Li, S., Zheng, P., Fan, J., & Wang, L. (2022). Toward proactive human–robot collaborative assembly: A multimodal transfer-learning-enabled action prediction approach. *IEEE Transactions on Industrial Electronics, 69*(8), 8579-8588. DOI: https://doi.org/10.1109/TIE.2021.3105977

Lott, S., & Phillips, D. (2021) *Python Object-Oriented Programming: Build Robust and Maintainable Object-Oriented Python Applications and Libraries.* 4th ed. Birmingham, UK: Packt Publishing.

Matus, S., & Cano, S. (2025). Human-robot interaction in higher education: A literature review. In A. Coman & S. Vasilache (Eds.), *Social Computing and Social Media. HCII 2025* (Lecture Notes in Computer Science, vol. 15786). Cham: Springer. DOI: https://doi.org/10.1007/978-3-031-93539-8_16

Mishra, D., Parish, K., Lugo, R., & Wang, H. (2021) A framework for using humanoid robots in the school learning environment. *Electronics* 10(6): 1-12. DOI: https://doi.org/10.3390/electronics10060756

Mukherjee, D., Gupta, K., Chang, L. H., & Najjaran, H. (2022) A survey of robot learning strategies for human-robot collaboration in industrial settings. *Robotics and Computer-Integrated Manufacturing, 73*, 102231. DOI: https://doi.org/10.1016/j.rcim.2021.102231

Mulko, M. (2023) 5 of the world's most realistic humanoid robots ever. *Interesting Engineering*. Available at: https://interestingengineering.com/lists/5-worlds-realistic-humanoid-robots (Accessed: 13 September 2025).

Papakostas, G., Sidiropoulos, G., Papadopoulou, C., Vrochidou, E., Kaburlasos, V., Papadopoulou, M., Holeva, V., Nikopoulou, V., & Dalivigkas, N. (2021) Social robots in special education: A systematic review. *Electronics, 10*(12): 1398. DOI: https://doi.org/10.3390/electronics10121398

Qidwai, U., Kashem, S., & Conor, O. (2020) Humanoid robot as a teacher's assistant: Helping children with autism to learn social and academic skills. *Journal of Intelligence & Robot Systems* 98(1): 759–770. DOI: https://doi.org/10.1007/s10846-019-01075-1

Rumbaugh, J., Jacobson, I., & Booch, G. (1999) *The Unified Modeling Language Reference Manual*. 2nd ed. Addison-Wesley.

Singh, N., Chouhan, S. S., & Verma, K. (2021). Object oriented programming: Concepts, limitations and application trends. *2021 5th International Conference on Information Systems and Computer Networks (ISCON)*, Mathura, India, 1-4. DOI: https://doi.org/10.1109/ISCON52037.2021.9702463