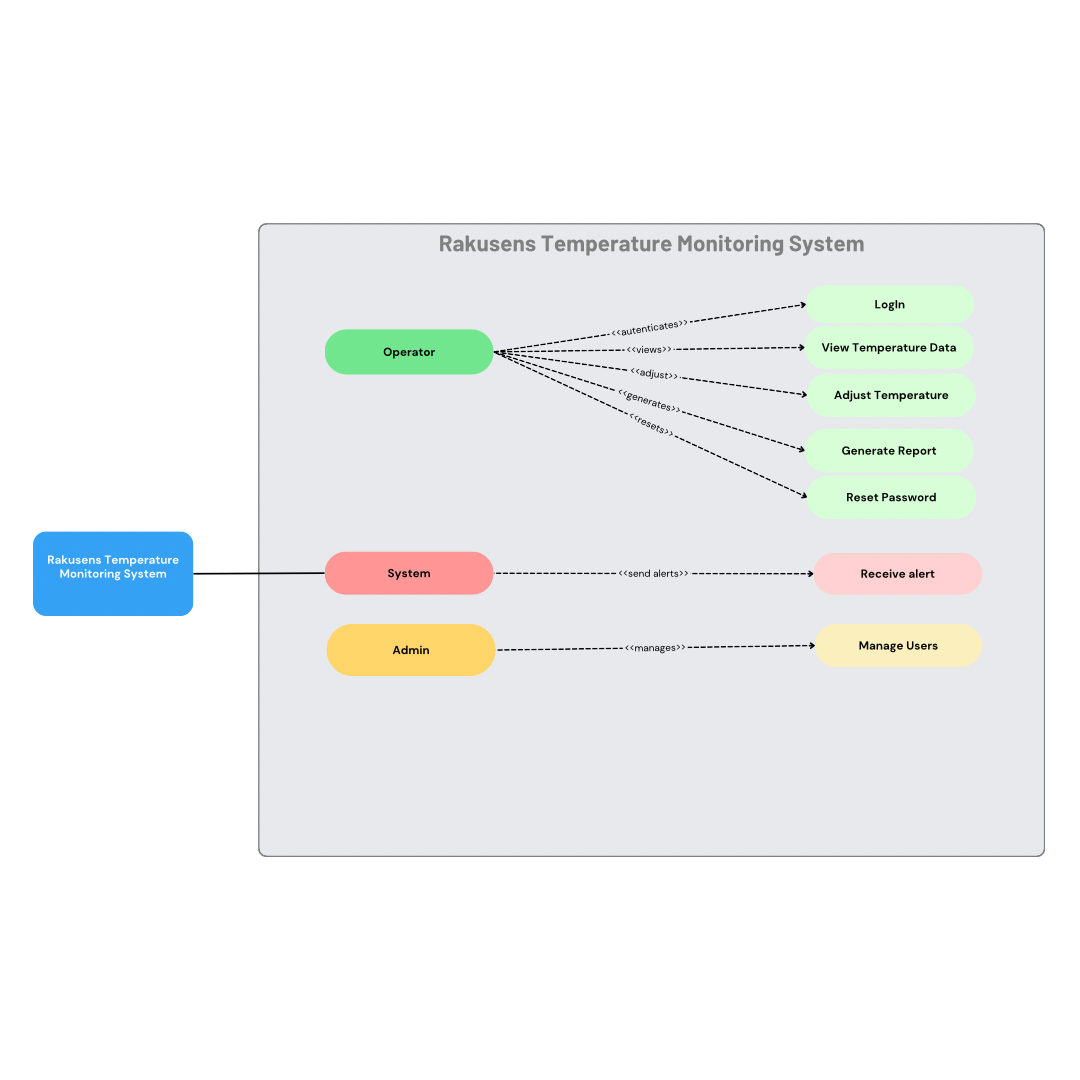
**P4. Interactive, real-time visualisation dashboard**

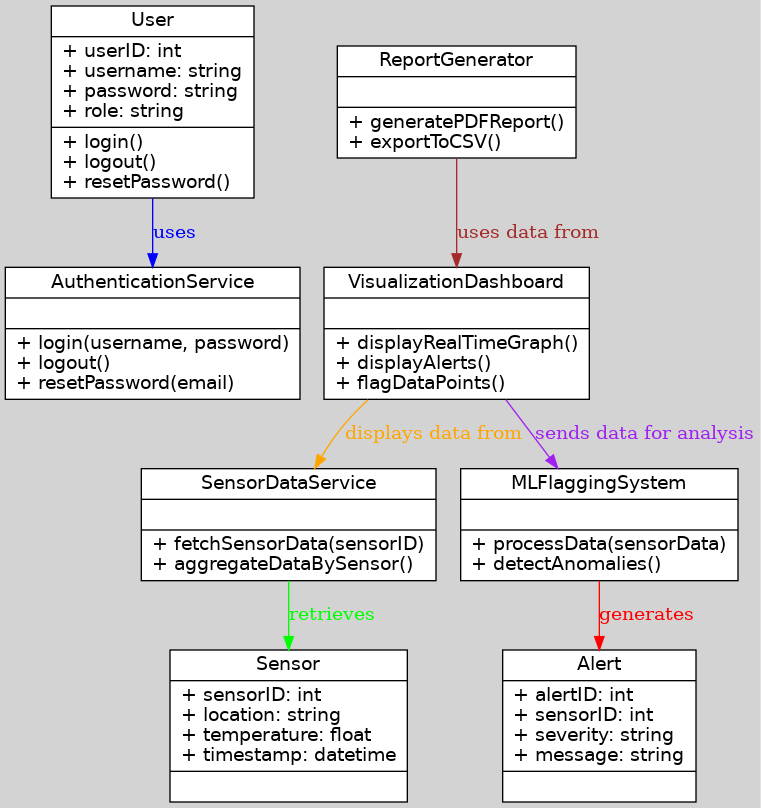
**UML Diagrams**

* **UML Case Diagram**



The Use Case Diagram shows how users will interact with the Rakusens Temperature Monitoring System and what they will do. Its main actors are Admin, Operator, and System. The Operator is responsible for a core set of actions, including allowing log-ins, viewing temperature data, changing temperature set points, generating reports, and resetting passwords when necessary. They will monitor and control the temperature data in real-time. The Admin manages User accounts in the system and ensures that will access the system functions. The use of alerts by the System whenever an anomaly is detected in the temperature data completes the role of the System and allows the Operator to be immediately notified. This diagram demonstrates the flow of actions, clearly defined responsibilities, and better operational efficiency and systems security.

* **UML Class Diagram**



The class diagram depicts the complete internal structure of the Rakusens Temperature Monitoring System by showing the classes, attributes, methods, and their relationships within the system. It begins with the User class, which stores users' information like ID, username, password, and role, with methods for logging in, logging out, and password reset. The AuthenticationService handles users' authentication, thereby ensuring secure access. The Sensor class records temperature-related data such as sensor ID, location, temperature values, and the respective timestamps. SensorDataService retrieves the data and provides an aggregate of information for further analysis. The VisualizationDashboard offers real-time graphs and alerts with which the operator can gain an overview of the whole system status. It works closely with the MLFlaggingSystem, which interacts with the sensor data and detects any anomalies. When these irregularities are discovered, the class Alert gets triggered by notifying the system and the operators of a potential problem. Furthermore, the ReportGenerator class renders formal reports in either PDF or CSV format for permanent record-keeping and analysis. The class diagram very nicely expresses the flow of data through the system; it shows the interaction effect between components to ensure effective monitoring, anomaly detection, and reporting.