**Drone Monitoring for Smart Traffic Signal Optimization**

**Objective:**

As a member of a team working on an initiative to optimize traffic signal management in a busy city, your task is to design and implement a drone monitoring system that collects real-time traffic data from sensors and optimizes traffic signal timings dynamically based on current traffic conditions.

**Deliverables:**

1. **Data Flow Diagram:** Illustrate how real-time traffic data is collected from drones, analysed, and used to optimize traffic signal timings.
2. **Pseudocode and Implementation:** Provide detailed pseudocode and Java code for the algorithms used to optimize traffic signals and manage intersections.
3. **Documentation:** Explain the design decisions behind the algorithms, data structures used for efficient processing, assumptions made (e.g., drone reliability), and potential improvements for further optimization.
4. **User Interface:** Develop intuitive and informative interfaces for traffic managers and city officials to interact with the system, monitor traffic conditions, and manage signal timings.
5. **Testing:** Include comprehensive test cases to validate the functionality and effectiveness of the drone monitoring system under various traffic scenarios and conditions.

**Requirements:**

1. **Drone Monitoring:** Design a system that uses drones to collect real-time traffic data from sensors at various intersections across the city.
2. **Data Analysis:** Develop algorithms to analyse the collected data and optimize traffic signal timings dynamically based on current traffic conditions.
3. **Real-time Optimization:** Ensure the system can adjust signal timings in real-time to respond to changing traffic patterns and optimize flow.
4. **Visualization and Reporting:** Develop visualizations to monitor traffic conditions and signal timings in real-time, and generate reports on traffic flow improvements, average wait times, and overall congestion reduction achieved.
5. **User Interaction:** Design a user interface for traffic managers to monitor and manually adjust signal timings if needed, and provide a dashboard for city officials to view performance metrics and historical data.

**Grading Criteria:**

* Code Functionality (30%): Does the code meet all requirements, work correctly, and handle all edge cases and errors gracefully?
* Code Quality (20%): Is the code clean, well-organized, and follow proper naming conventions and coding standards?
* Documentation (20%): Is the documentation comprehensive, clear, and cover all aspects of the approach, pseudocode, code explanation, and assumptions?
* Testing (20%): Are the test cases comprehensive, demonstrating correct functionality and covering a wide range of inputs?
* User Interface (10%): Is the interface intuitive, user-friendly, and fully functional?

**Submission Guidelines:**

* Submit your assignment as a zip file containing all deliverables.
* Ensure your code is well-organized, commented, and follows proper naming conventions and coding standards.
* Include a README file with instructions on how to run and interact with the system.

**Late Submission Policy:**

* Late submissions will incur a penalty of 10% per day, up to a maximum of 2 days.
* Submissions beyond 2 days will not be accepted unless prior arrangements have been made.

**Plagiarism Policy:**

* All work must be original and completed individually.
* Collaboration on concepts is allowed, but each student must submit their own implementation.
* Plagiarism will result in a score of zero for the assignment and possible disciplinary action.