

Coding & Solutioning

Debugging and Traceability

Date	13 MAY 2023
Team ID	NM2023TMID09640
Project Name	IoT Based Weather Adaptive Street Lighting System

Introduction:

Debugging and traceability are crucial aspects of software development that facilitate the identification and resolution of issues within a codebase. In the context of the weather adaptive street lighting system, effective debugging techniques and traceability mechanisms are essential to ensure the system operates reliably. This section discusses the debugging strategies and traceability measures employed in the provided code snippet.

Debugging Techniques:

The code snippet incorporates several debugging techniques to assist in identifying and resolving potential issues. These techniques include:

a. Serial Communication: The code utilizes the Serial communication interface to output relevant information during runtime. By printing debug messages to the Serial monitor, developers can monitor the program flow, variable values, and execution status, aiding in the identification of errors or unexpected behavior.

b. Commenting and Uncommenting: The code includes commented-out sections that can be selectively uncommented to enable or disable specific functionalities or debug statements. This technique allows developers to isolate and test specific code segments, facilitating the identification of potential bugs.

c. Conditional Debugging: The code implements conditional statements that selectively execute certain code blocks based on specific conditions. By strategically placing these conditions, developers can enable or disable debugging code, allowing for targeted investigation of particular scenarios.

d. Error Handling: The code employs error handling mechanisms, such as printing error messages or failure notifications, to provide valuable insights into potential issues. By catching and reporting errors, developers can pinpoint the root causes and take appropriate corrective actions.

Traceability Measures:

The code snippet incorporates traceability measures to ensure that code changes, revisions, and updates can be tracked effectively. These measures include:

a. Version Control: The code can be managed using a version control system, such as Git, to track changes over time. By utilizing version control, developers can maintain a chronological record of modifications, allowing for easy identification of introduced issues and the ability to roll back changes if necessary.

b. Documentation: The code is accompanied by comprehensive documentation that outlines the purpose, functionality, and usage of each component. This documentation serves as a reference for developers, facilitating understanding and traceability of the codebase.

c. Code Comments: Inline comments are used throughout the code to explain the purpose of variables, functions, and code blocks. By providing detailed comments, developers can better understand the code's intent, aiding in traceability and troubleshooting.

d. Clear Naming Conventions: The code adopts clear and consistent naming conventions for variables, functions, and other components. By using meaningful and descriptive names, developers can easily navigate and trace the code, improving overall traceability.

Conclusion:

The code snippet incorporates effective debugging techniques and traceability measures to facilitate the identification and resolution of issues. Through the utilization of techniques such as serial communication, commenting/uncommenting, conditional debugging, and error handling, developers can efficiently debug the code and identify potential bugs or unexpected behavior. Additionally, traceability measures such as version control, documentation, code comments, and clear naming conventions ensure that code changes are well-documented and easily traceable. These practices contribute to the overall reliability and maintainability of the weather adaptive street lighting system.