1. **Purpose**: Understand the purpose of the functionality and how it fits into the overall system or project.
2. **Features**: Get a detailed list of features and functionalities provided by the system.
3. **Architecture**: Learn about the architecture and design principles underlying the functionality.
4. **Dependencies**: Identify any dependencies on other systems, components, or services.
5. **Data Flow**: Understand the flow of data within the functionality and how information is processed.
6. **Use Cases**: Discuss typical use cases and scenarios where the functionality is utilized.
7. **Error Handling**: Inquire about error handling mechanisms and how issues are detected and resolved.
8. **Performance**: Ask about performance considerations and any key performance indicators relevant to the functionality.
9. **Security**: Understand security measures implemented within the functionality to protect data and ensure compliance.
10. **Monitoring**: Learn how the functionality is monitored, any logging mechanisms in place, and how performance is tracked.