# Section 5: 4+1 Architectural Views - Scenarios

See Slides and Paper “Architectural Blueprints – The 4+1 View Model of Software Architecture” in Supplemental Materials

## Question 1

Describe each of Kruchten’s Four Views of Architecture.

### Answer

*Grader: Please review the four views presented in the paper mentioned in the question.*

1. Logical View: Presents the system service classes described by the problem domain and scenarios. Describes services as components and classes. Describes the relationships between services in terms of the responsibilities each has been assigned and their dependency on other services to fulfil those responsibilities.
2. Process View: 1) Describes the assignment of services to processes / executable programs. 2) Describes the interaction between processes to implement the given scenarios; usually via UML sequence diagrams. 3. Describes the optional use of tasks (threads) to describe scheduled activities needed to implement scenarios.
3. Development View: Presents system component from the implementation components i.e. packages, libraries, and other implementation artifacts. Can also presents a work breakdown (work-tasks), scheduling dependencies between work-tasks, and assignment of work-tasks to developers.
4. Physical View: Presents the assignment of components or subsystems (processes) onto processors (servers), network communication between processor, subnets, firewalls, physical locations, and other aspects of the system’s deployment.

## Question 2

Describe the purpose of each of these sections from 4+1 Scenario template.

1. Scenario ID
2. System Fault / Failure
3. System State
4. System Response
5. Response Metric

Why do we specify multiple Responses and Metrics for a single scenario?

### Answer

1. Scenario ID: A name that uniquely identifies each scenario.
2. System Fault / Failure: A description of the issue addressed by the scenario.
3. System State: The state of the system when the issue occurs.
4. System Response: The response of the system to the fault’s occurrence. This is the design tactic the system will apply to address the fault i.e. keep the fault from becoming a system failure.
5. Response Metric: A measurable system quality that is used to evaluate the effectiveness of the response and to compare with other responses in the same fault scenario.

There can be multiple possible responses to the fault defined by a scenario. Each response will have its positive and negative effects on a system quality i.e. performance, robustness, security, etc. The metric is given to provide some quantifiable measurement used to evaluate and aid in the selection of the response to be used in the system’s design.