

Common Quality Attributes

- Availability - Need to be there when required i.e can always get a call through
- Performance - Control resource demand tactics i.e robot voice doesn't start mid call
- Modifiability - Can change an update system i.e want to add video calls
- Reliability - System does not fail i.e call does not drop
- Security - System is not vulnerable i.e. people can't listen in
- Testability - Possible to test in any way required

Common Stakeholders

- User
- Service Provider
- Analysis
- Developer

Common Design Patterns

- Layered - distinct layers, can only interact with adjacent layers
- Model-View-Controller (MVC) - UI and application isolation, user interface changes often
- Blackboard - think /r/place
- N-tier Architecture - See Layered

Types of Structures

- Modular - static structures, focus on how functionality is divided up
- Component & Connector - Runtime structures focused on interactions
- Allocation - Mapping to Environments

Lifecycles

- V Model
 - “Verification & Validation” model
 - Extension of waterfall model
 - Testing planned in parallel with development
 - Coding phase joins these two sides
- RUP
 - “Rational Unified Process”
 - Kind of like a gantt chart
- Spiral Model
 - Risk driven process model
 - Iterative design model
- Agile SCRUM
 - 2-4 week sprint
 - Daily scrum meetings
 - Most important issues tackled first as they are on top of backlog