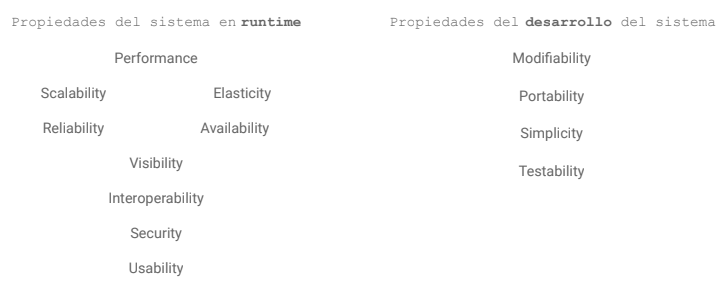
## Atributos de calidad

* Disponibilidad
* Rendimiento
  + Network performance
  + User-Perceived Performance
  + Efficiency
* Escalabilidad
* Visibilidad
* Testeabilidad
* Portabilidad
* Interoperabilidad
* Usabilidad
* Manejabilidad
* Fiabilidad
* Seguridad
* Simplicidad
* Modificabilidad



## 

## Tácticas

### Tácticas para disponibilidad (confiabilidad+recuperación)

* Ping/Echo
* Heartbeat/Watchdog
* Voting
* Redundancia
  + Active redundancy
  + Passive redundancy
  + Spares
* Removal from service
* Transactions
* Process monitor

### Tácticas para rendimiento

* Reduce events number
* Reduce resources consumption
* Concurrencia
* Multiple copies
* More resources available
* More copies of computation
* More copies of data
* Scheduling
* Priorizar

### Tácticas para escalabilidad

* Stateless
* Async design
* Sharding

### Tácticas para modificabilidad

* Reduce module size (dividirlo, por ejemplo)
* Mejorar la cohesión
* Disminuir el acoplamiento

### Tácticas para seguridad

* Detect attacks
* Resist
* Limit access
* Identify
* React (lock access)
* Recover (restore)

### Tácticas para testeabilidad

* Control state
* Record interface
* Assert
* Limit non-determinism

### Tácticas para usabilidad

* Cancel
* Undo
* Pause
* Autocorrector

### Tácticas para interoperabilidad

* Use interfaces

## 

## Estilos arquitectónicos

### Replication styles

* Replicated repository
* Cache

### Hierarchical styles

* Client-server
* Layered systems
* Layered client-server
* Client-stateless-server
* Client-cache-stateless-server
* Layered-client-cache-stateless-server
* Remote session
* Remote data access

### Mobile code

* Virtual machine
* Remote evaluation
* Code on demand
* Latered-code-on-demand-client-cache-stateless-server
* Mobile agent

### Peer to peer styles

* Event-based integration
* Distributed objects
* Brokered distributed objects

### Data flow styles

* Pipe & Filter
* Uniform Pipe & Filter
* Batch sequential

### Data centered Architectures

* Database
* Blackboard

## Web Architecture & REST

x

## Preguntas

Qué es la arquitectura del software  
Nombrar X atributos

REST y qué es hypermedia

Qué es visibilidad

Qué es throughput

Service Level Agreement

Explicar estilos

Explicar Client-Stateless-Server

Explicar Remote Evaluation

Explicar Code On Demand

Por qué las bases de datos relacionales no son escalables  
NoSQL. Diferencia entre replicación y sharding  
ACID

Qué es sharding

Consistencia eventual

Enterprise Service Bus

Algún cloud pattern  
Auto-scaling

Priority Queue

Enterprise Applications