Arquitectura de Software

Capítulo 2:

Un sistema tiene impacto sobre la capacidad de este para alcanzar los objetivos de negocio.

Los objetivos de negocio del sistema pueden satisfacerse mediante comportamientos o características provistas

por este.

**Requerimiento:** Es una especificación que describe alguna funcionalidad,

atributo o factor de calidad de un sistema de *software*. Puede describir también

algún aspecto que restringe la forma en que se construye ese sistema.

Es posible que los requerimientos describan aspectos que restrinjan el proceso para

desarrollar un sistema de *software*. Por ejemplo, hay casos en que los sistemas deben implementarse bajo consideraciones fechas de inicio o terminación, presupuestos o usos de tecnologías específicos.

La ingeniería de requerimientos

es la disciplina que engloba las actividades relacionadas con la obtención, análisis, documentación y

validación de estos.

Capítulo 2.2 Requerimientos y niveles de abstracción

Recabarlos no solo consiste en escuchar al propietario o usuario final de un sistema, sino posteriormente

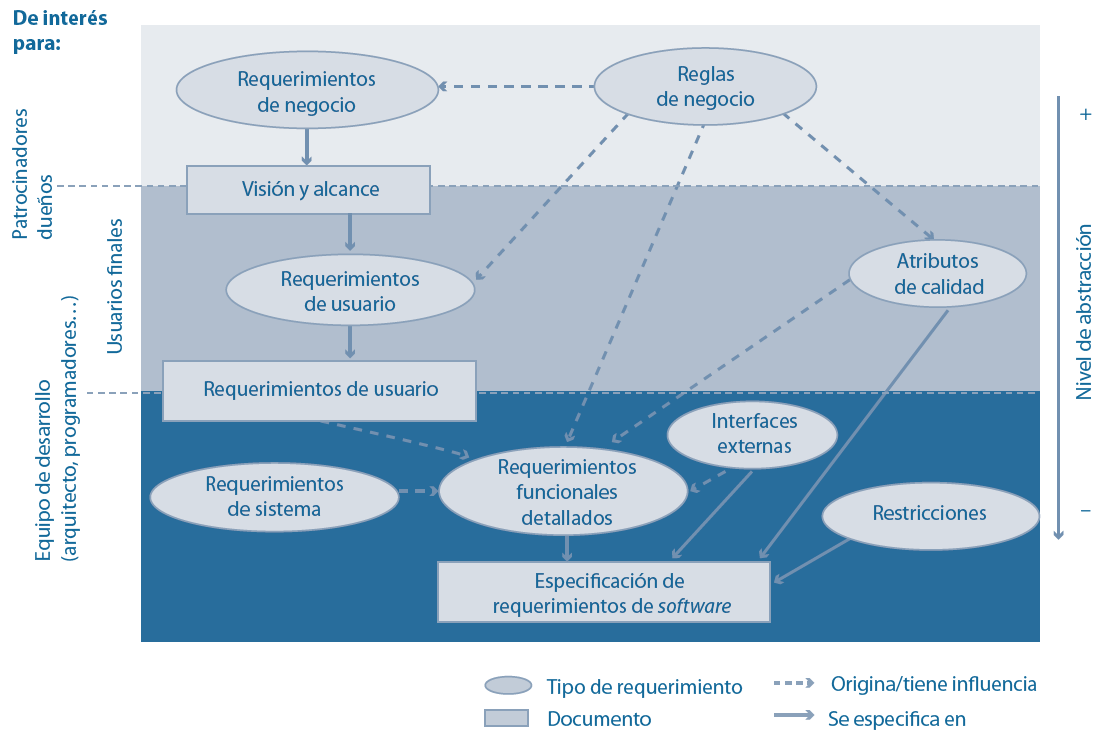
escribir en una lista lo que él quiere que este haga. Además del propietario o usuario de un sistema,

pueden existir otras personas interesadas en este; es importante reconocer correcta y oportunamente a todas

ellas.

Niveles de requerimientos:

* Nivel Superior (Requerimientos de negocio)
* Nivel Medio (Aspectos relacionados a la interacción con los usuarios)
* Nivel Inferior (Requerimientos que describen situaciones o elementos necesarios para el diseño del sistema)



Capítulo 2.2 Requerimientos de usuario y requerimientos funcionales

**Requerimientos de usuario:** especifican servicios que por lo

habitual dan soporte a procesos de negocio que los usuarios podrán llevar a cabo mediante el sistema, por ejemplo:

“Comprar boleto (de autobús) en línea”. (Se obtienen por medio de casos de uso o historias de usuario).

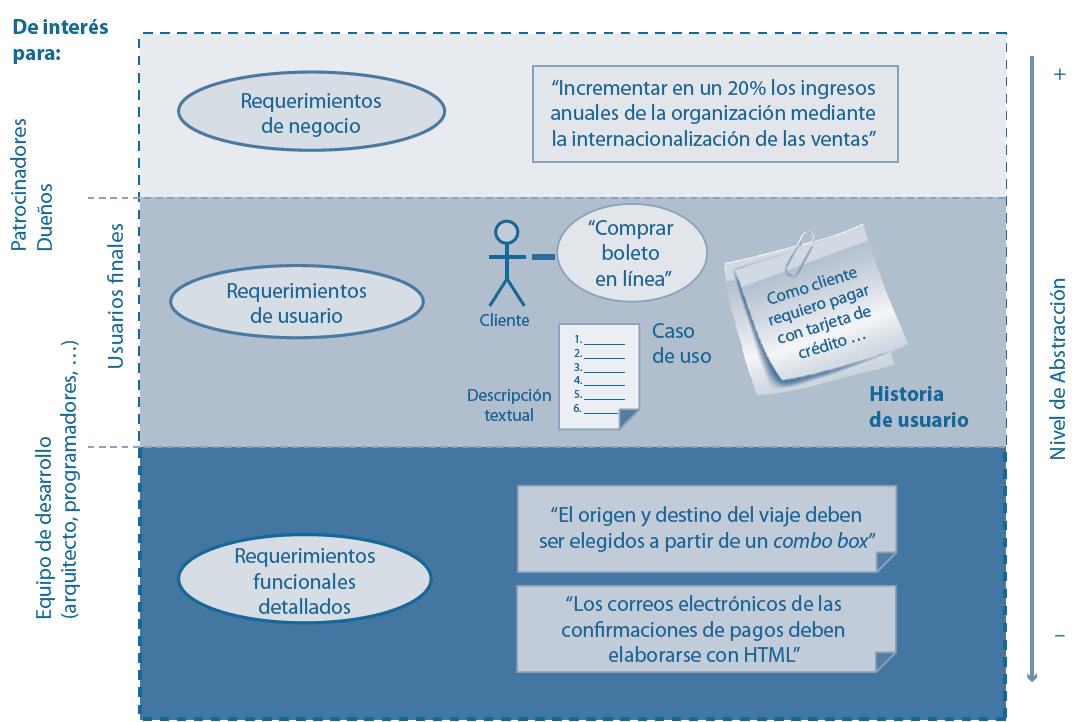
**Requerimientos funcionales:** describen detalles finos de diseño y/o implementación relacionados

a los requerimientos de usuario, por ejemplo: “El origen y des tino del viaje deben ser elegidos a partir de

un *combo box*”.

Los requerimientos de usuario se dividen en:

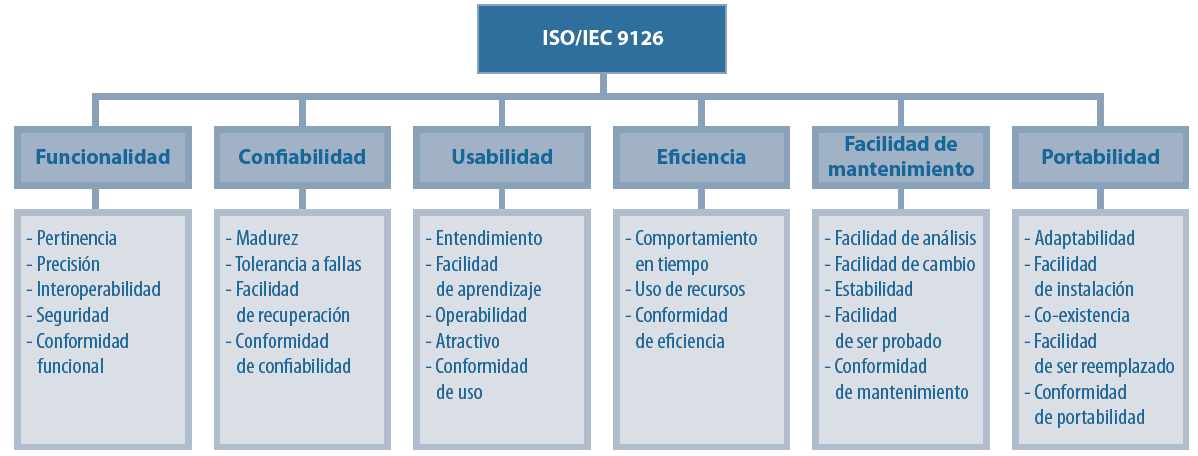
* Primario: El primario describe servicios fundamentales que los usuarios desean llevar a cabo por medio del sistema, por ejemplo: “Comprar boleto (de autobús) en línea”, soportan procesos clave de la organización.
* Secundario: describe acciones necesarias para dar soporte a la realización de los servicios especificados en los casos de uso o historias de usuario de los primarios. Para el ejemplo de caso de uso primario mencionado antes, casos de uso secundarios podrían ser: Acceder al sistema, Realizar alta o Enviar comprobante de compra.



Capítulo 2.2.2 Atributos de Calidad

Los atributos de calidad especifican características útiles para establecer criterios sobre la calidad del sistema. Actualmente no existe un “listado único” de atributos de calidad relevantes para diseñar la *arquitectura de software*.

Sin embargo, estándares para la evaluación de la calidad, como el ISO/IEC 9126 (*International Standard Organization*).



*1.* **Disponibilidad.** Indicador acerca de si el sistema se encuentra en una condición operable cuando

requiere ser utilizado.

*2.* **Seguridad.** Indicador del grado de protección ante usos o accesos inapropiados del sistema.

*3.* **Desempeño**. Indicador sobre la cantidad de trabajo realizado por el sistema considerando tiempo y

recursos.

*4.* **Facilidad de prueba.** Indicador acerca de la facilidad con la cual se elaboran pruebas efectivas para el

sistema.

*5.* **Modificabilidad.** Indicador referente al costo de realizar cambios en el sistema.

*6.* **Usabilidad**. Indicador sobre la facilidad con la cual el sistema puede ser utilizado por los usuarios.

*10.* **Interoperabilidad.** Indicador acerca de la facilidad del sistema para intercambiar información con

otros sistemas mediante interfaces.

Capítulo 2.2.3 Restricciones

Las restricciones describen aspectos que limitan el proceso de desarrollo del sistema.

Para facilitar su manejo se distinguen dos subclases:

I. Restricciones técnicas.

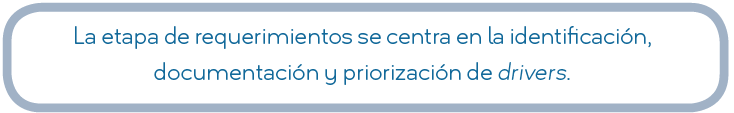
II. Restricciones administrativas. (Costos y tiempos de desarrollo)

2.2.4 Reglas de negocio e interfaces externas

Las reglas de negocio especifican políticas, estándares, prácticas o procedimientos organizacionales y/o gubernamentales que rigen o restringen la forma en que se realizan las actividades o procesos de una organización.

2.3 Drivers Arquitectónicos, requerimientos de arquitectura

Drivers arquitectónicos: Requerimientos de mayor influencia respecto de la forma que tomarán los elementos que componen las estructuras arquitectónicas.



Estos *drivers arquitectónicos* se clasifican en tres clases:

***1. Drivers* funcionales:** Son el subconjunto de los requerimientos funcionales, proveen información relevante para llevar a cabo la descomposición funcional del sistema y asignar estas funcionalidades a elementos específicos en la arquitectura.

Con ello, estos *drivers* se eligen considerando:

- Su relevancia en la satisfacción de los objetivos del negocio del sistema.

- La complejidad técnica que representa su implementación.

- El hecho de que representan algún escenario relevante para la arquitectura.

***2. Drivers* de atributos de calidad.** Generalmente son todos los requerimientos de atributos de calidad son drivers, pero para producir un diseño inicial es preferible considerar nada más un subconjunto de los requerimientos de atributos de calidad.

Se eligen considerando:

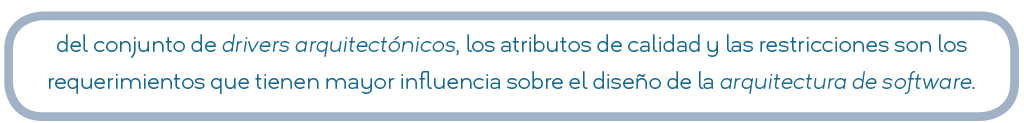
- Su relevancia en la satisfacción de los objetivos del negocio del sistema, es decir, su importancia para el

cliente.

- La complejidad técnica que representa su implementación, esta es su importancia para el arquitecto.

***3. Drivers* de restricciones. ESTOS NO TIENEN PRIORIDAD,** Por esta razón, todas las restricciones son *drivers* de la arquitectura.

2.3.5 Influencia de los drivers arquitectónicos en el diseño de la arquitectura



2.4 Fuentes de información para la extracción de drivers arquitectónicos

* Documento de visión y alcance
* Documento de requerimientos de usuario
* Documento de especificación de requerimientos

2.5 Métodos para identificar drivers arq

*1.* Taller de atributos de calidad (QAW, por sus siglas en inglés).

*2.* Método de diseño centrado en la arquitectura (ACDM, por sus siglas en inglés).

*3.* Funcionalidad, usabilidad, confiabilidad, desempeño, soporte (FURPS+).

As a < type of user >, I want < some goal > so that < some reason >.