

# Resumen Ingeniería de Software 2: clase 1 (parte 2) – SRS

## Requerimientos SRS

### *Requerimientos*

- Va por etapas:
  - Solicitud.
  - Definición.
  - Análisis y especificación.
- Requerimientos funcionales:
  - Definen el comportamiento del sistema.
  - Describen las tareas que el sistema debe realizar.
  - Al definir un requerimiento funcional hay que mantener un equilibrio entre la excesiva generalidad y el exceso de detalle con descripciones innecesarias o redundantes.
- No funcionales
  - Definen aspectos que son deseables desde el punto de vista del usuario. (No implican funcionalidades). Se pueden ver como restricciones, tiempos de respuesta, características de usabilidad, facilidad de mantenimiento, etc.

### *Especificación de requerimientos*

- Mediante el documento ConOps normalizado en el estándar IEE Std. 1362-1998.
- Es un documento dirigido a los usuarios, el mismo describe las características de un sistema propuesto desde el punto de vista del usuario.
- La descripción del sistema es el medio de comunicación que recoge la visión general, cualitativa y cuantitativa de las características del sistema; compartido por la parte del cliente y desarrolladora.

### *IEEE 1362 → descripción del sistema*

- Ofrece un formato y contenidos específicos para confeccionar descripciones de sistema en los desarrollos y modificaciones del mismo.
- No especifica una técnica exacta, sino que otorga líneas generales que deben ser respetadas. Guía de referencia.
- El estándar identifica los elementos que al menos debe incluir una **Descripción del sistema**. El usuario puede incorporar otros elementos agregando cláusulas y sub-cláusulas.

### *Requerimientos del Software – IEEE 830 - SRS*

- Documento denominado SRS (ERS) y normalizado en el estándar IEEE Std. 830-1998.

## Resumen Ingeniería de Software 2: clase 1 (parte 2) – SRS

- Un documento SRS es la especificación de las funciones que realiza un determinado producto de software, programa o conjunto de programas en un determinado entorno.
- El documento de especificación de requisitos puede desarrollarlo personal que represente a la parte desarrolladora, o de la parte cliente; se aconseja la intervención de ambas partes.

### *IEEE 830 – SRS*

#### ➤ **Alcance del documento**

- Brindar una colección de buenas prácticas para escribir especificaciones de requerimientos de software (SRS). Se describen los contenidos y las cualidades de una buena especificación de requerimientos.

#### ➤ **Naturaleza del documento**

- El SRS es una especificación para un producto de software particular. El SRS es escrito por uno o más representantes del equipo de desarrollo y uno o más representantes de la parte cliente.

#### ➤ **Ambiente del SRS**

- El software desarrollado puede contener toda la funcionalidad del proyecto o puede ser parte de un sistema más grande. En el último caso habrá un SRS que declarará las interfaces entre el sistema y el software desarrollado, y pondrá que función externa y requerimientos de funcionalidad tiene con el software desarrollado.

### *Características de un buen documento SRS*

#### ➤ **Correcto**

- Un SRS es correcto sí y sólo sí cada requisito declarado se encuentra en el software.

#### ➤ **No ambiguo**

- Un SRS es inequívoco sí, y sólo sí cada requisito declarado tiene sólo una interpretación.

#### ➤ **Completo**

- Un SRS está completo sí, y sólo sí se reconoce cualquier requisito externo impuesto por una especificación del sistema.

#### ➤ **Consistente**

- La consistencia se refiere a consistencia interior. Si un SRS no está de acuerdo con algún documento de nivel superior, como una especificación de requerimientos de sistema, entonces no es consistente.

## Resumen Ingeniería de Software 2: clase 1 (parte 2) – SRS

- **Priorizado**
  - Un SRS es priorizado por la importancia de sus requerimientos particulares.
- **Comprobable**
  - Un SRS es comprobable si, y sólo si cada requisito declarado es comprobable. Un requisito es comprobable si y sólo si existe algún proceso con que una persona o máquina pueda verificar que el producto de software reúne el requisito. En general los requisitos ambiguos no son comprobables.
- **Modificable**
  - Un SRS es modificable sí, y sólo si, su estructura y estilo son tales que puede hacerse cualquier cambio a los requerimientos fácilmente, completamente y de forma consistente mientras conserva la estructura y estilo.
- **Trazabilidad**
  - Claridad del origen de cada requerimiento y su trazabilidad hacia los requerimientos futuros que serán desarrollados. Hacia adelante y hacia atrás

### *Consideraciones para un buen SRS*

- **Preparación conjunta del SRS**
    - El SRS se debe preparar en conjunto con las partes que intervienen, esto para lograr un buen acuerdo entre las partes.
  - **Evolución de SRS**
    - El SRS debe evolucionar en conjunto con el software, para registrar cambios, los responsables de estos y la aceptación de los mismos.
  - **Prototipos**
    - El uso de prototipos se utiliza frecuentemente para la definición de requerimientos.
  - **Diseño incorporado en el SRS**
    - El SRS puede incorporar los atributos o funciones externos al sistema, en particular las que describen el diseño para interactuar entre los subsistemas.
  - **Requerimientos incorporados en el SRS**
    - Los detalles particulares de los requerimientos son anexados como documentos externos (CU, plan de proyecto, plan de aseguramiento de calidad, etc).
- 
-

### Partes de un SRS

---

#### FICHA DEL DOCUMENTO

#### CONTENIDO

<b>1</b>	<b><u>INTRODUCCIÓN</u></b>
1.1	Propósito
1.2	Alcance
1.3	Referencias
<b>2</b>	<b><u>DESCRIPCIÓN GENERAL</u></b>
2.1	Perspectiva del producto
2.2	Funcionalidad del producto
2.3	Características de los usuarios
2.4	Evolución previsible del sistema
<b>3</b>	<b><u>REQUISITOS NO FUNCIONALES</u></b>
3.1	Requisitos de rendimiento
3.2	Seguridad
3.3	Portabilidad
<b>4</b>	<b><u>MANTENIMIENTO</u></b>
<b>5</b>	<b><u>APÉNDICES</u></b>

### Sección 1 del SRS

#### Introducción

---

##### ❖ 1.1 Propósito

Se define el propósito del documento y se especifica a quién va dirigido el documento

##### ❖ 1.2 Alcance o ámbito del sistema

Se da un nombre al futuro sistema . Se explica lo que el sistema hará y lo que no hará.  
Se describen los *beneficios, objetivos y metas* que se espera alcanzar con el futuro sistema

##### ❖ 1.3 Referencias

Se presenta una lista completa de todas las referencias de los documentos mencionados o utilizados para escribir el SRS.

Identificar cada documento por el título, número de reporte, fecha y publicación. Y las fuentes de las referencias de donde se obtuvieron.

### Sección 2 del SRS

#### Descripción General

---

❖ Esta sección del SRS debe describir los factores generales que afectan el producto y sus requerimientos. No declara los requerimientos específicos. Los que se definen en detalle en Sección 3 del SRS.

❖ Esta sección normalmente consiste en:

**Perspectiva del producto**

**Funcionalidades del producto**

**Características de los usuarios**

**Evoluciones previsibles del sistema**

#### ❖ 2.1. Perspectiva del producto

Si el producto es independiente y totalmente autónomo, debe declararse que así es.

Si el SRS define un producto que es un componente de un sistema más grande entonces se debe relacionar los requerimientos de ese sistema más grande a la funcionalidad del software y debe identificar las interfaces entre ese sistema y el software.

#### ❖ 2.2. Funciones del sistema

Se debe presentar un resumen de las funciones del futuro sistema.

Las funciones deberán mostrarse de forma organizada, y pueden utilizarse gráficos, siempre que reflejen las relaciones entre funciones y no el diseño del sistema.

## Resumen Ingeniería de Software 2: clase 1 (parte 2) – SRS

### ❖ 2.3. Características del Usuario

Se deben describir las características generales de los usuarios intencionales del producto que incluye nivel educativo, experiencia, y la especialización técnica.

### ❖ 2.4. Evoluciones previsibles del sistema

Se identifican requerimientos que serán implementados en futuras versiones

## Sección 3 del SRS

### Requerimientos no funcionales

- ❖ Debe contener todos los requerimientos no funcionales del software a un nivel de detalle para permitirles a los diseñadores diseñar el sistema, y a los auditores probar que el sistema satisface esos requerimientos.

#### ❖ 3.3.1 Requerimientos de rendimiento

Requerimientos relacionados con la carga que se espera tenga que soportar el sistema. Por ejemplo, nro de terminales, nro esperado de usuarios simultáneamente conectados, etc.

Todos estos requerimientos deben ser mensurables. Por ejemplo, indicando “el 95% de las transacciones deben realizarse en menos de 1 segundo”, en lugar de “los operadores no deben esperar a que se complete la transacción”.

## Resumen Ingeniería de Software 2: clase 1 (parte 2) – SRS

### ❖3.3.2 Seguridad

Especificación de elementos que protegerán al software de accesos, usos y sabotajes maliciosos, así como de modificaciones o destrucciones maliciosas o accidentales. Los requerimientos pueden especificar:

Empleo de técnicas criptográficas, Registro de ficheros con “logs” de actividad, Asignación de determinadas funcionalidades a determinados módulos, Restricciones de comunicación entre determinados módulos, Comprobaciones de integridad de información crítica.

### ❖3.3.3Portabilidad

Especificación de atributos que debe presentar el software para facilitar su traslado a otras plataformas u entornos.

Pueden incluirse:

% componentes dependientes del servidor. % código dependiente del servidor. Uso de un determinado lenguaje por su portabilidad. Uso de un determinado compilador o plataforma de desarrollo. Uso de un determinado sistema operativo.

## Sección 4 del SRS

### Mantenimiento

---

- ❖ Identificación del tipo de mantenimiento necesario del sistema.
- ❖ Especificación de quien debe realizar las tareas de mantenimiento, por ejemplo usuarios, o un desarrollador.
- ❖ Especificación de cuando debe realizarse las tareas de mantenimiento. Por ejemplo, generación de estadísticas de acceso semanales y mensuales.

### Sección 5 del SRS

#### Apéndices

---

Pueden contener todo tipo de información relevante para la SRS pero que, propiamente, no forme parte de la SRS.

Por ejemplo:

*Casos de Uso*

*Historias de usuario*