|  |
| --- |
| [ESCRIBIR EL NOMBRE DE LA COMPAÑÍA] |
| Especificación de requisitos de software |
| *Proyecto: Control de Tareas* |
|  |
| **Revisión*: [01.00]*** |
| **[26/08/2022]** |

|  |
| --- |
| Especificación de Requisitos según estándar de IEEE 830. |

**Contenido**

[Ficha del documento 4](#_heading=h.gjdgxs)

[1. Introducción 5](#_heading=h.30j0zll)

[1.1. Propósito 5](#_heading=h.1fob9te)

[1.2. Ámbito del Sistema 5](#_heading=h.3znysh7)

[1.3. Definiciones, Acrónimos y Abreviaturas 5](#_heading=h.2et92p0)

[1.4. Referencias 5](#_heading=h.tyjcwt)

[1.5. Visión General del Documento 5](#_heading=h.3dy6vkm)

[2. Descripción General 6](#_heading=h.1t3h5sf)

[2.1. Perspectiva del Producto 6](#_heading=h.4d34og8)

[2.2. Funciones del Producto 6](#_heading=h.2s8eyo1)

[2.3. Características de los Usuarios 6](#_heading=h.17dp8vu)

[2.4. Restricciones 6](#_heading=h.3rdcrjn)

[2.5. Suposiciones y Dependencias 7](#_heading=h.26in1rg)

[2.6. Requisitos Futuros 7](#_heading=h.lnxbz9)

[3. Requisitos Específicos 8](#_heading=h.35nkun2)

[3.1 Requisitos comunes de los interfaces 9](#_heading=h.1ksv4uv)

[3.1.1 Interfaces de usuario 9](#_heading=h.44sinio)

[3.1.2 Interfaces de hardware 9](#_heading=h.2jxsxqh)

[3.1.3 Interfaces de software 9](#_heading=h.z337ya)

[3.1.4 Interfaces de comunicación 9](#_heading=h.3j2qqm3)

[3.2 Requisitos funcionales 9](#_heading=h.1y810tw)

[3.3 Requisitos no funcionales 10](#_heading=h.4i7ojhp)

[3.3.1 Requisitos de rendimiento 10](#_heading=h.2xcytpi)

[3.3.2 Seguridad 10](#_heading=h.1ci93xb)

[3.3.3 Fiabilidad 10](#_heading=h.3whwml4)

[3.3.4 Disponibilidad 11](#_heading=h.2bn6wsx)

[3.3.5 Mantenibilidad 11](#_heading=h.qsh70q)

[3.3.6 Portabilidad 11](#_heading=h.3as4poj)

[3.4 Otros Requisitos 11](#_heading=h.1pxezwc)

# Ficha del documento

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Revisión** | **Autor** | **Modificación** |
| *26/08/2022* | *1.0* | *Joaquin Araya*  *Vicente Gonzalez*  *Raul Venegas* | *Desarrollo Informe ERS* |
|  |  |  |  |

Documento validado por las partes en fecha:

|  |  |  |
| --- | --- | --- |
| Por el cliente |  | Por la empresa suministradora |
| [Firma] |  | [Firma] |
| Sr./Sra. |  | Sr./Sra. |

# 1. Introducción

Este documento se presentarán los requisitos necesarios para el desarrollar el software que ayude en la gestión, control y administración de tareas. Este documento esta estructurado bajo las normas estándar IEEE 830

## 1.1. Propósito

En esta subsección se definirá el propósito del documento ERS y se especificará a quién va dirigido el documento

El propósito de este documento es proporcionar las especificaciones y pautas generales que se deberá seguir para el desarrollo de “Control de Tareas”, tomando en cuenta todas las etapas desde su inicio hasta el completo desarrollo del proyecto. Este documento está dirigido a todo aquel personal involucrado en el proceso de desarrollo, gerentes y usuarios.

## 1.2. Ámbito del Sistema

En esta subsección:

• El software tendrá el nombre de “Control de Tareas”.

• Se explicará lo que el sistema hará y lo que no hará. Gestión de procesos, ordenamiento y control de actividades, la cual permite definir perfiles personalizados, mejorar los tiempos de respuestas y con esta información mejorar la toma de decisiones que permita que la información esté disponible en todo momento para la toma de decisiones.

• Se describirán los beneficios, objetivos y metas que se espera alcanzar con el futuro sistema. Con el sistema a implementar se espera que los procesos tengan una mejora, tener procesos automatizados, lo cual haría que la empresa pueda competir con sus similares de igual o mejor manera, mejorando el servicio que hoy brindan y llegando a nuestros clientes.

## 1.3. Definiciones, Acrónimos y Abreviaturas

En esta subsección se definirán todos los términos, acrónimos y abreviaturas utilizadas en la ERS.

Términos:

* Interfaz: Medio que permite la comunicación entre el sistema y el usuario
* Sistema operativo: Conjunto de programas informáticos que permiten la administración eficaz de los recursos de una computadora
* Base de datos: Es un conjunto de datos almacenados sistemáticamente en un archivo digital para el posterior uso en el sistema
* Login: Nombre o alias que se da a una persona para permitirle el acceso al sistema siempre y cuando estén previamente registrado
* Password: Contraseña o clave para autentificar el ingreso a un lugar o sitio

Acrónimos:

Abreviaturas:

* ERS: Especificación de Requisitos Software. Este documento
* DB: Base de datos
* SO: Sistema operativo

## 1.4. Referencias

En esta subsección se mostrará una lista completa de todos los documentos referenciados en la ERS.

IEEE Recommended Practice for Software Requirements Specification. ANSI/IEEE std. 830, 1998.

Documento “CASO 5 PTY4613 2019”

## 1.5. Visión General del Documento

En esta subsección se describe brevemente los contenidos y la organización del resto de la ERS.

Este documento está compuesto por tres secciones, comenzando por la introducción a este y se hace referencia general a lo que se quiere construir.

En la segunda sección se mostrará una descripción general del software a desarrollar con el fin de mostrar las principales funciones, usuarios con sus respectivos roles, restricciones, suposiciones y dependencias, los requisitos futuros y los requisitos específicos.

En la tercera y última sección se definirán los requisitos detallados para poder lograr desarrollar el software solicitado.

# 2. Descripción General

En esta sección se describen todos aquellos factores que afectan al producto y a sus requisitos. No se describen los requisitos, sino su contexto. Esto permitirá definir con detalle los requisitos en la sección 3, haciendo que sean más fáciles de entender.

Normalmente, esta sección consta de las siguientes subsecciones: Perspectiva del producto, funciones del producto, características de los usuarios, restricciones, factores que se asumen y futuros requisitos.

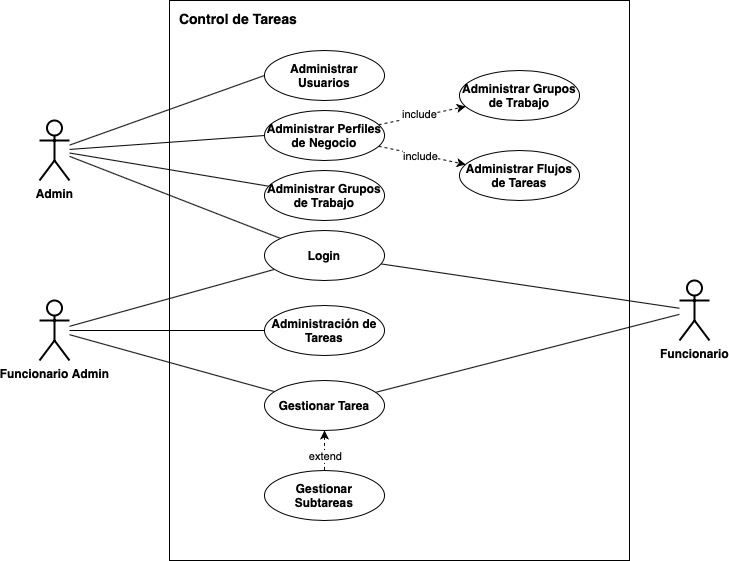
## 2.1. Perspectiva del Producto

Esta subsección debe relacionar el futuro sistema (producto software) con otros productos. Si el producto es totalmente independiente de otros productos, también debe especificarse aquí. Si la ERS define un producto que es parte de un sistema mayor, esta subsección relacionará los requisitos del sistema mayor con la funcionalidad del producto descrito en la ERS, y se identificarán las interfaces entre el producto mayor y el producto aquí descrito. Se recomienda utilizar diagramas de bloques.

El sistema de control de tareas será un producto diseñado para trabajar tanto en en un entorno web como de escritorio lo que permitirá tener un acceso rápido y accesible independiente de la ubicación del usuario. Este sistema será independiente.

## 2.2. Funciones del Producto

En esta subsección de la ERS se mostrará un resumen, a grandes rasgos, de las funciones del futuro sistema. Las funciones deberán mostrarse de forma organizada, y pueden utilizarse gráficos, siempre y cuando dichos gráficos reflejen las relaciones entre funciones y no el diseño del sistema.



## 2.3. Características de los Usuarios

Esta subsección describirá las características generales de los usuarios del producto, incluyendo nivel educacional, experiencia y experiencia técnica.

El software de gestión está pensado para usuarios con conocimiento básico de manejo computacional, manejo del negocio.

## 2.4. Restricciones

Esta subsección describirá aquellas limitaciones que se imponen sobre los desarrolladores del producto:

• Políticas de la empresa.

• Limitaciones del hardware.

• Interfaces con otras aplicaciones.

• Operaciones paralelas.

• Funciones de auditoría.

• Funciones de control.

• Lenguaje(s) de programación.

• Protocolos de comunicación.

• Requisitos de habilidad.

• Criticalidad de la aplicación.

• Consideraciones acerca de la seguridad.

* Lenguaje y tecnologías en uso: Java, ORACLE??????????????
* El sistema deberá tener un diseño simple de facil entendimiento independiente del lenguaje de programación o de la plataforma con que se ejecute.
* Tiempo: El proyecto se tendrá que realizar en un plazo no superior a los 4 meses, esto contando desde la fecha de inicio del 00/08/2022 al 00/12/2022.
* Funciones de control de roles: El sistema debe ser capaz de controlar los permisos de visualización y ejecución dependiendo del rol asignado al usuario, de manera que solo tenga una vista de la información proporcionada para ese rol.

## 2.5. Suposiciones y Dependencias

Esta subsección de la ERS describirá aquellos factores que, si cambian, pueden afectar a los requisitos. Por ejemplo, los requisitos pueden presuponer una cierta organización de ciertas unidades de la empresa, o pueden presuponer que el sistema correrá sobre cierto sistema operativo. Si cambian dichos detalles en la organización de la empresa, o si cambian ciertos detalles técnicos, como el sistema operativo, puede ser necesario revisar y cambiar los requisitos.

El sistema a desarrollar en su aplicación de escritorio correrá sobre plataforma windows 7 o superior, por el lado de la aplicación web funcionará en las distintas aplicaciones web que hay en el mercado actualmente.

El sistema tendrá la capacidad de generar reportes en formato PDF.

## 2.6. Requisitos Futuros

Esta subsección esbozará futuras mejoras al sistema, que podrán analizarse e implementarse en un futuro.

El sistema debe tener la capacidad que en el futuro de que si el cliente necesita generar cambios se puedan implementar sin mayor problema.

# 3. Requisitos Específicos

Esta sección contiene los requisitos a un nivel de detalle suficiente como para permitir a los diseñadores diseñar un sistema que satisfaga estos requisitos, y que permita al equipo de pruebas planificar y realizar las pruebas que demuestren si el sistema satisface, o no, los requisitos. Todo requisito aquí especificado describirá comportamientos externos del sistema, perceptibles por parte de los usuarios, operadores y otros sistemas. Esta es la sección más larga e importante de la ERS. Deberán aplicarse los siguientes principios:

• El documento debería ser perfectamente legible por personas de muy distintas formaciones e intereses.

• Deberán referenciarse aquellos documentos relevantes que poseen alguna influencia sobre los requisitos.

• Todo requisito deberá ser unívocamente identificable mediante algún código o sistema de numeración adecuado.

• Lo ideal, aunque en la práctica no siempre realizable, es que los requisitos posean las siguientes características:

* **Corrección**: La ERS es correcta si y sólo si todo requisito que figura aquí (y que será implementado en el sistema) refleja alguna necesidad real. La corrección de la ERS implica que el sistema implementado será el sistema deseado.
* **No ambiguos**: Cada requisito tiene una sola interpretación. Para eliminar la ambigüedad inherente a los requisitos expresados en lenguaje natural, se deberán utilizar gráficos o notaciones formales. En el caso de utilizar términos que, habitualmente, poseen más de una interpretación, se definirán con precisión en el glosario.
* **Completos**: Todos los requisitos relevantes han sido incluidos en la ERS. Conviene incluir todas las posibles respuestas del sistema a los datos de entrada, tanto válidos como no válidos.
* **Consistentes**: Los requisitos no pueden ser contradictorios. Un conjunto de requisitos contradictorio no es implementable.
* **Clasificados**: Normalmente, no todos los requisitos son igual de importantes. Los requisitos pueden clasificarse por importancia (esenciales, condicionales u opcionales) o por estabilidad (cambios que se espera que afecten al requisito). Esto sirve, ante todo, para no emplear excesivos recursos en implementar requisitos no esenciales.
* **Verificables**: La ERS es verificable si y sólo si todos sus requisitos son verificables. Un requisito es verificable (testeable) si existe un proceso finito y no costoso para demostrar que el sistema cumple con el requisito. Un requisito ambiguo no es, en general, verificable. Expresiones como a veces, bien, adecuado, etc. Introducen ambigüedad en los requisitos. Requisitos como “en caso de accidente la nube tóxica no se extenderá más allá de 25Km" no es verificable por el alto costo que conlleva.
* **Modificables**: La ERS es modificable si y sólo si se encuentra estructurada de forma que los cambios a los requisitos pueden realizarse de forma fácil, completa y consistente. La utilización de herramientas automáticas de gestión de requisitos facilitan enormemente esta tarea.
* **Trazables**: La ERS es trazable si se conoce el origen de cada requisito y se facilita la referencia de cada requisito a los componentes del diseño y de la implementación. La trazabilidad hacia atrás indica el origen (documento, persona, etc.) de cada requisito. La trazabilidad hacia delante de un requisito R indica que componentes del sistema son los que realizan el requisito R.

## 3.1 Requisitos comunes de los interfaces

Descripción detallada de todas las entradas y salidas del sistema de software.

### 3.1.1 Interfaces de usuario

Describir los requisitos del interfaz de usuario para el producto. Esto puede estar en la forma de descripciones del texto o pantallas del interfaz. Por ejemplo posiblemente el cliente ha especificado el estilo y los colores del producto. Describa exacto cómo el producto aparecerá a su usuario previsto.

La interfaz de usuario consistirá en un conjunto de ventanas, botones, listas desplegables, campos de textos, mensajes de error y gráficas. Estas deberán ser diseñadas para el sistema propuesto, tanto para el modulo web como para la aplicación de escritorio.

### 3.1.2 Interfaces de hardware

Especificar las características lógicas para cada interfaz entre el producto y los componentes de hardware del sistema. Se incluirán características de configuración.

Los equipos para poder interactuar con el sistema deberá tener como mínimo las siguientes especificaciones :

* Monitor: Con el fin de que el usuario pueda visualizar el sistema y sus actividades
* Mouse: El sistema debe ser capaz de reaccionar a el movimiento del mouse y los botones.
* Teclado: El sistema debe ser capaz de reaccionar a las pulsaciones del teclado, este es el hardware encargado en la entrada de datos al sistema
* Smartphone o Tablet: El sistema deberá ser capaz de proyectarse en dispositivos móviles inteligentes

### 3.1.3 Interfaces de software

Indicar si hay que integrar el producto con otros productos de software.

* Para cada producto de software debe especificarse lo siguiente:
* Descripción del producto software utilizado
* Propósito del interfaz
* Definición del interfaz: contiendo y formato
* SO: Windows 7 o superior
* Exploradores: Mozilla, Edge , Google Chrome, Zafari (estos en cualquiera de sus versiones)

### 3.1.4 Interfaces de comunicación

Describir los requisitos del interfaces de comunicación si hay comunicaciones con otros sistemas y cuales son las protocolos de comunicación.

Los servidores, cliente y aplicación se comunicaran entre si, mediante protocolos estándares en internet.

## 3.2 Requisitos funcionales

Definición de acciones fundamentales que debe realizar el software al recibir información, procesarla y producir resultados.

En ellas se incluye:

* Comprobación de validez de las entradas
* Secuencia exacta de operaciones
* Respuesta a situaciones anormales (desbordamientos, comunicaciones, recuperación de errores)
* Parámetros
* Generación de salidas
* Relaciones entre entradas y salidas (secuencias de entradas y salidas, formulas para la conversión de información)
* Especificación de los requisitos lógicos para la información que será almacenada en base de datos (tipo de información, requerido)

Las requisitos funcionales pueden ser divididos en sub-secciones.

3.2.1 Requisito funcional 1

Creación de perfiles:

* El sistema de información debe considerar el rol de administrador, diseñador de procesos y funcionario.

3.2.2 Requisito funcional 2

Rol de usuario Administrador:

* El administrador podrá crear en el sistema todos los recursos que el sistema necesita para su operación (usuarios, roles, unidades y su jerarquía, etc.).

3.2.3 Requisito funcional 3

Rol usuario funcionario:

* Cada funcionario de acuerdo a su rol podrá crear tareas, asignando un responsable y un plazo, adjuntando toda la información necesaria para su desarrollo. Además podrá desagregar las tareas recibidas en tareas subordinadas bajo el mismo concepto anterior. También podrá relacionar tareas, pudiendo crear dependencias entre ellas.

3.2.4 Requisito funcional 4

Rol usuario diseñador

* Cada diseñador de procesos, podrá crear flujos de trabajo “tipo”, los cuales podrán ser instanciados y asignados en cualquier momento. Esto tiene sentido para procesos repetitivos, los cuales se ejecutan en diferentes momentos del año. El objetivo es generar el flujo de trabajo una vez, y ejecutarlo varias veces, en diferente o mismo tiempo, con diferentes o mismos responsables.

3.2.5 Requisito funcional 5

Aceptación o Rechazo de tareas

* Cada funcionario podrá aceptar o rechazar la tarea asignada, ingresando una justificación. Este acto tendrá como consecuencia un aviso a quien asignó la tarea para que resuelva el problema, replanteando la tarea, los plazos o asignando el responsable.

3.2.6 Requisito funcional 6

Cumplimiento de tareas:

* Cada funcionario tendrá acceso a un panel de control, el cual muestra todas las tareas que tiene asignadas, su porcentaje de cumplimiento de acuerdo al plazo. El indicador debe ser mostrado como un semáforo, el cual indique en verde las tareas en desarrollo que tengan un plazo de entrega mayor a 1 semana, en amarillo las tareas que estén con plazo de entrega menor a una semana y en rojo las tareas atrasadas.

3.2.7 Requisito funcional 7

Control de tareas:

* Cada funcionario además podrá ver el estado de las tareas que asignó y las subordinadas a ella.

3.2.8 Requisito funcional 8

Avance tarea:

* El cálculo de porcentaje de avance se hará considerando el tiempo transcurrido en relación al plazo.

3.2.9 Requisito funcional 9

Avance de tareas globales:

* El porcentaje global se calculará ponderando las tareas de acuerdo a su duración. Así mismo se calculará el avance de una tarea de acuerdo a las tareas subordinadas.

3.2.10 Requisito funcional 10

Cumplimiento de tarea:

* Cada funcionario al ejecutar una tarea, o una tarea de un flujo podrá marcarla como terminada en cualquier momento, quedando la tarea con un 100%.

3.2.11 Requisito funcional 11

Mensaje de alerta:

* Cada funcionario podrá reportar un problema en la ejecución de la tarea, ingresando un mensaje, el cual será alertado a quien asignó la tarea para su revisión, solución y posibles cambios (asignación, plazo, etc).

3.2.12 Requisito funcional 12

Revisión carga:

* Cada funcionario de acuerdo a su rol podrá ver la carga de trabajo de cada uno de los funcionarios subordinados.

3.2.13 Requisito funcional 13

Rol maestro:

* El rol con mayor jerarquía tendrá acceso a un tablero global de control, el cual mostrará un resumen de tareas por unidad interna, pudiendo mostrar el detalle de estas. Además incluirá los porcentajes de cumplimiento globales y los detallados, para apoyar la toma de decisión.

## 3.3 Requisitos no funcionales

### 3.3.1 Requisitos de rendimiento

Especificación de los requisitos relacionados con la carga que se espera tenga que soportar el sistema. Por ejemplo, el número de terminales, el número esperado de usuarios simultáneamente conectados, número de transacciones por segundo que deberá soportar el sistema, etc.

Todos estos requisitos deben ser mesurables. 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”.

### 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 requisitos 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.
* El sistema debe incluir medidas de seguridad tales como enmascarar clave y control de sesiones.
* La autenticación de usuarios debe considerar las medidas de seguridad respectivas, tales como manejo de sesiones y acceso con usuario-clave-perfil a modo de acceder a las funcionalidades de acuerdo al perfil o rol que posee el usuario.

### 3.3.3 Fiabilidad

Especificación de los factores de fiabilidad necesaria del sistema. Esto se expresa generalmente como el tiempo entre los incidentes permisibles, o el total de incidentes permisible.

### 3.3.4 Disponibilidad

Especificación de los factores de disponibilidad final exigidos al sistema. Normalmente expresados en % de tiempo en los que el software tiene que mostrar disponibilidad.

### 3.3.5 Mantenibilidad

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.

### 3.3.6 Portabilidad

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

* Porcentaje de componentes dependientes del servidor.
* Porcentaje de 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.

## 3.4 Otros Requisitos

Cualquier otro requisito que no encaje en otra sección.