**Documento de Especificación de Requerimientos de Software para B+**

**Versión 0.1  
  
5 de Junio del 2021  
  
  
  
Preparado por:**

**Pedro Eduardo Cruz de la Fuente**

**Alexandra Guadalupe Adelfa Oreza Mendicuti**

**Noé Alejandro González Bautista**

**Suemi Andrea Castillo González**

**Alejandro Torre Reyes**

**CONTROL DE DOCUMENTACIÓN**

**Control de Configuración**

|  |  |
| --- | --- |
| Título: | Especificación de Requerimientos de Software |
| Referencia: | IEEE Recommended Practice for Software Requirements Specification. ANSI/IEEE std. 830, 1998. |
| Autor: | Pedro Eduardo Cruz de la Fuente |
| Fecha: | 5 de Junio del 2021 |

**Histórico de versiones**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Versión** | **Fecha** | **Estado** | **Responsable** | **Nombre de archivo** |
| 0.1 | 05/Jun/2021 | B | Noé González | ERS.doc |
|  |  |  |  |  |

Estado: (B)orrador, (R)evisión, (A)probado

**Histórico de cambios**

|  |  |  |
| --- | --- | --- |
| **Versión** | **Fecha** | **Cambios** |
| 0.2 | 05/Jun/2021 | Se realizó un primer acercamiento a las secciones del punto, introducción. |
| 0.2.1 | 13/Jun/2021 | Se empezó a trabajar con el punto 2, se realizó el punto 2.1 y se detallaron los puntos del 2.1.1 al 2.1.5 |
|  |  |  |

**Contenido**

[**1 INTRODUCCIÓN 5**](#_heading=h.37m2jsg)

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

[1.2 Audiencia. 5](#_heading=h.3znysh7)

[1.3 Alcance. 5](#_heading=h.2et92p0)

[*1.3.1 Producto a elaborar. 5*](#_heading=h.tyjcwt)

[*1.3.2 Objetivos. 5*](#_heading=h.3dy6vkm)

[*1.3.3 Fecha deseada de inicio y finalización del desarrollo. 5*](#_heading=h.1t3h5sf)

[1.4 Definiciones, acrónimos y abreviaturas. 5](#_heading=h.4d34og8)

[1.5 Referencias. 5](#_heading=h.2s8eyo1)

[1.6 Panorama general. 6](#_heading=h.17dp8vu)

[**2 DESCRIPCIÓN GENERAL 7**](#_heading=h.3rdcrjn)

[2.1 Perspectiva del producto. 7](#_heading=h.26in1rg)

[*2.1.1 Interfases del sistema. 7*](#_heading=h.lnxbz9)

[*2.1.2 Interfases de usuario. 7*](#_heading=h.35nkun2)

[*2.1.3 Interfases de hardware. 7*](#_heading=h.1ksv4uv)

[*2.1.4 Interfases de software. 8*](#_heading=h.44sinio)

[*2.1.5 Interfases de comunicación. 8*](#_heading=h.2jxsxqh)

[*2.1.6 Restricciones de memoria. 8*](#_heading=h.z337ya)

[*2.1.7 Operaciones. 8*](#_heading=h.3j2qqm3)

[*2.1.8 Requerimientos de adaptación del sitio. 8*](#_heading=h.1y810tw)

[2.2 Funciones del producto. 9](#_heading=h.4i7ojhp)

[2.3 Usuarios. 9](#_heading=h.2xcytpi)

[*2.3.1 Características Generales. 9*](#_heading=h.1ci93xb)

[*2.3.2 Clases. 9*](#_heading=h.3whwml4)

[2.4 Restricciones generales. 9](#_heading=h.2bn6wsx)

[2.5 Supuestos y dependencias. 10](#_heading=h.qsh70q)

[2.6 Asignación de requerimientos. 10](#_heading=h.3as4poj)

[**3 REQUERIMIENTOS ESPECÍFICOS 11**](#_heading=h.1pxezwc)

[3.1 Interfases externas. 11](#_heading=h.49x2ik5)

[3.2 Funciones (requerimientos funcionales). 12](#_heading=h.2p2csry)

[3.3 Desempeño de requerimientos (requerimientos no funcionales). 12](#_heading=h.147n2zr)

[3.4 Requerimientos de bases de datos lógicas. 12](#_heading=h.3o7alnk)

[3.5 Restricciones de diseño. 13](#_heading=h.23ckvvd)

[*3.5.1 Estándares de conformidad. 13*](#_heading=h.ihv636)

[3.6 Documentación de usuario 13](#_heading=h.32hioqz)

[3.7 Atributos del sistema de software. 13](#_heading=h.1hmsyys)

[*3.7.1 Confiabilidad. 13*](#_heading=h.41mghml)

[*3.7.2 Disponibilidad. 13*](#_heading=h.2grqrue)

[*3.7.3 Seguridad. 13*](#_heading=h.vx1227)

[*3.7.4 Mantenimiento. 14*](#_heading=h.3fwokq0)

[*3.7.5 Portabilidad. 14*](#_heading=h.1v1yuxt)

[3.8 Organización de requerimientos específicos. 14](#_heading=h.4f1mdlm)

[**4 DEPENDENCIAS Y CONFLICTOS ENTRE REQUERIMIENTOS 15**](#_heading=h.2u6wntf)

[**5 GRADO DE SATISFACCIÓN E INSTALACIÓN DE LOS USUARIOS RESPECTO A LOS REQUERIMIENTOS 16**](#_heading=h.19c6y18)

[**6 TEMAS O DETALLES POR RESOLVER 17**](#_heading=h.3tbugp1)

[**7 GLOSARIO 18**](#_heading=h.28h4qwu)

[**8 APÉNDICES 19**](#_heading=h.nmf14n)

1. **Introducción**
   1. **Propósito.**

El presente documento tiene como propósito definir de manera clara y precisa todas las funcionalidades y restricciones del sistema que se desea construir. A través de este listado, se obtendrá una comprensión completa y específica de las características, necesidades y operatividad que requiere el sistema, lo que permitirá al equipo de trabajo involucrado en el proyecto avanzar a través de las siguientes fases del desarrollo, sentando la guía de las tareas que deberá cumplir el sistema. De igual manera, el listado servirá al cliente para comprender y ratificar que los requisitos establecidos en el documento cubran por completo las necesidades y funciones que ha considerado.

En la ERS deben recogerse tanto las necesidades de clientes y usuarios como los requisitos que debe cumplir el sistema software a desarrollar para satisfacer dichas necesidades.

Esta especificación está sujeta a revisiones por el equipo de desarrollo, que se recogerán por medio de sucesivas versiones del documento hasta alcanzar su aprobación. Una vez aprobado servirá de base al equipo de desarrollo para la construcción del nuevo sistema.

* 1. **Audiencia.**

El documento se encuentra dirigido al siguiente listado de personas:

* El equipo de desarrollo
* El cliente
  1. **Alcance.**
     1. **Producto a elaborar.**

El proyecto *B+* consiste en el desarrollo de una plataforma que permita facilitar el contacto entre solicitantes que se encuentren en necesidad de recibir donaciones de sangre en algún hospital o centro médico, y aquellas personas que tengan la intención de donar altruistamente, de manera que, para cada posible solicitante, se cuente con una red de donadores compatibles en términos de tipo de sangre y ubicación, creando una conexión personalizada y mayores oportunidades de cubrir las necesidades del solicitante.

A través de un sistema de campañas, *B+* se propone brindar los datos necesarios, entre los cuales se incluyen dirección de hospital, nombre y tipo de sangre de quien recibe y contactos, para que las personas interesadas en donar brinden su apoyo.

El solicitante podrá generar campañas dentro de la plataforma, a través de las cuales compartir su caso, sus necesidades y el resto de la información necesaria para su contacto. En estas mismas, se encontrará información para realizar el proceso de donación en el centro médico u hospital donde este resida, de tal manera que los donadores interesados en la campaña puedan acceder a estos datos. Los solicitantes podrán también dar por terminada una campaña, así como renovar una campaña en caso de que esta alcance su fecha de expiración sin haber sido completada.

Cualquier persona que tenga el deseo o intención de donar, podrá registrarse con los datos necesarios al respecto tipo de sangre, ubicación y demás. Pasará a formar parte de una red de donadores, quienes podrán realizar búsquedas o recibir notificaciones sobre campañas de manera personalizada, con respecto a la información de cada usuario.

Los donadores podrán acceder a estas campañas, usar la información de contacto para obtener cualquier otro dato deseado, y conocer el proceso que debe seguir para realizar la donación de la forma más sencilla posible.

* + 1. **Objetivos.**

Con la propuesta de proyecto se propone:

1. Ofrecer ayuda a aquellas personas en necesidad de recibir donaciones de sangre.
2. Brindar un canal de comunicación entre solicitante de sangre, donador.
3. Crear una plataforma recurrente para personas interesadas en realizar donaciones de sangre.
4. Brindar a los usuarios donadores la información de los hospitales más cercanos en su área en donde pueda donar sangre.
   * 1. **Fecha deseada de inicio y finalización del desarrollo.**

*<Dar las fechas deseables por parte del cliente para el inicio y finalización del desarrollo del sistema de software.>*

* 1. **Definiciones, acrónimos y abreviaturas.**

|  |  |
| --- | --- |
| Nombre | Descripción |
| **B+** | Es el nombre del proyecto, surge de un juego de palabras en inglés por su pronunciación "*Be positive*" y también es parte de uno de los 4 grupos sanguíneos. |
| **Campaña** | Es la publicación que se muestra en la plataforma a los posibles donadores. Se presenta la información necesaria para la donación incluyendo tipo de sangre e información del hospital a donar. |
| **Solicitante** | Toda persona que realiza una solicitud de donación en el sistema. Es quien proporciona los detalles necesarios para la creación de la campaña, incluidos el tipo de sangre requerido, medios de contacto, datos generales y hospital donde se encuentra el donatario. |
| **Donador/donante** | Toda persona que proporciona su sangre o componentes sanguíneos, para uso terapéutico sin esperar nada a cambio.Toda persona que proporciona su sangre o componentes sanguíneos, para uso terapéutico sin esperar nada a cambio. |
| **Donatario** | Es la persona que recibe la donación, no es necesariamente quien realiza la solicitud pero es en quien se utiliza la donación. |
| **ERS** | Es el documento de especificación de requerimientos de software. |

* 1. **Referencias.**

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

* 1. **Panorama general.**

|  |  |
| --- | --- |
| **Sección 1** | En esta primera sección se da una introducción del proyecto, su alcance y objetivos. |
| **Sección 2** | Se da un entendimiento de las principales funciones que tendrá que realizar, factores de limitación, supuestos y dependencias que afectan el sistema. |
| **Sección 3** | Se da una descripción detallada de los requerimientos que el sistema debe de satisfacer. |
| **Sección 4** | Se definen los conflictos y dependencias de cada uno de los requisitos, así como una descripción del por qué. |
| **Sección 5** | Se representan los grados de satisfacción e insatisfacción de cada uno de los requerimientos que han sido implementados |
| **Sección 6** | Se tiene un listado de aquellos temas o detalles que quedaron pendientes, que no han sido solucionados o no han quedado claros. |
| **Sección 7** | Se contará con un glosario del documento |
| **Sección 8** | Contiene ejemplos de formatos, referencias de información, descripciones de problemas a ser resueltos por el software y empaquetamientos especiales del código y los medios para encontrar la seguridad, exportación, carga inicial u otros requerimientos. |

1. **Descripción general**
   1. **Perspectiva del producto.**

*<Se pone al producto en perspectiva con otros productos o proyectos. Si el producto es independiente y totalmente autocontenido, si es una nueva versión y/o se va a sustituir algún otro, aquí debe establecerse. Si se define un producto que es componente de uno más grande entonces se deben describir los componentes del sistema o proyecto mayor e identificar las interfases entre dicho sistema y el software que se está especificando.*

*Un diagrama de bloques mostrando los componentes más importantes del sistema más grande, sus interconexiones y las interfaces externas puede ser de mucha utilidad.*

*Identificar las principales interfaces externas del producto de software que se está especificando (la descripción no debe ser detallada) y describir el hardware de computadoras y el equipo periférico a ser usado (esta es una descripción global).*

*Se deben comentar las razones por las que ciertas restricciones de diseño serán especificadas más adelante en la ERS.>*

El sistema *B+* es un proyecto independiente, autocontenido y de inversión social en la forma de una plataforma web la cual tiene como objetivo crear un medio de comunicación entre pacientes los cuales solicitan una donación de sangre y un donador altruista el cual esté dispuesto a realizar la donación. Además, el sistema ofrece la oportunidad de mostrar una lista de hospitales cercanos al donador altruista en donde se le es permitido realizar una donación sin necesidad de algún donatario.

En esta primera versión, se plantea que el sistema funcione como un canal de comunicación entre el solicitante y el donador, donde la plataforma le facilitará al solicitante difundir su campaña. En esta versión el área de operación será reducida, concentrándose en los hospitales localizados en la ciudad de Mérida.

* + 1. **Interfaces del sistema.**
* El solicitante interactúa con el sistema para crear una campaña o removerla, así como para aplazar su expiración.
* El administrador interactúa con el sistema para dar de baja a solicitudes.
* El API de servicios de hospitales interactúa con el sistema para brindar la información de los hospitales en México.
* El donador interactúa con el sistema para buscar solicitudes de donación, y reportar solicitudes falsas
* El sistema interactúa con el solicitante para notificarle de que su campaña está próxima a expirar, confirmar la creación de su campaña, proveer un enlace con el cual podrá eliminar sus campañas.
* El sistema interactúa con el administrador para enviar los reportes de los solicitantes.
* El sistema interactúa con el donador para enviar información de contacto del hospital, y notificar de una solicitud cercana.
  + 1. **Interfaces de usuario.**

*<Se debe especificar lo siguiente:*

*a)Las características lógicas de cada interfase entre el producto de software y sus usuarios. Esto incluye aquellas características de configuración (por ejemplo, formatos de pantalla, esquemas de páginas o ventanas, contenido de algún reporte o menú, formatos de mensajes, etc.), necesarias para obtener los requerimientos de software.*

*b)Todos los aspectos de optimización de la interfase con la persona que usa el sistema. Esto es, incluir una lista de qué hace y qué no hace el sistema desde la perspectiva del usuario.>*

El sistema a desarrollar contará con un conjunto de ventanas el cual le permitirá al usuario tener una facilidad de uso, estas ventanas constaran de:

* Formularios para la creación de campañas, registro de cuenta e inicio de sesión
* Botones
* Mensajes de error
* Mensajes de éxito
* Menú de navegación
* Buscadores
* Cuadros de diálogo

* + 1. **Interfaces de hardware.**

Para este sistema, la interacción con un equipo de hardware no es requerida.

* + 1. **Interfaces de software.**

*Descripción y especificación de interacción de dbms postgresql*

|  |  |
| --- | --- |
| **Nombre** | PostgreSQL |
| **Mneumónico** | // |
| **Número de especificación** | // |
| **Número de versión** | 14.0 |
| **Fuente** | https://www.postgresql.org/ |

El sistema B+ interactuará con el sistema de base de datos, PostgreSQL, al registrar y remover las campañas de los solicitantes.

|  |  |
| --- | --- |
| **Nombre** | Hospitales Privados |
| **Mneumónico** | // |
| **Número de especificación** | // |
| **Número de versión** | 1.0 |
| **Fuente** | https://covid19.ciga.unam.mx/datasets/hospitales-privados/api |

*El sistema B+ interactuará con el API, Hospitales Privados, en el momento que sea requerida la información de hospitales privados.*

|  |  |
| --- | --- |
| **Nombre** | Hospitales Públicos |
| **Mneumónico** | // |
| **Número de especificación** | // |
| **Número de versión** | 1.0 |
| **Fuente** | https://covid19.ciga.unam.mx/datasets/hospitales-publicos/api |

*El sistema B+ interactuará con el API, Hospitales Privados, en el momento que sea requerida la información de hospitales públicos.*

*<Se especifica el uso de otros productos de software requeridos (por ejemplo sistemas manejadores de datos, sistema operativo o alguna utilería específica) y las interfases con otras aplicaciones desarrolladas (por ejemplo con el sistema de contabilidad). Para cada producto de software hay que especificar:*

* *Nombre.*
* *Mnemónico.*
* *Número de especificación.*
* *Número de versión.*
* *Fuente (quien lo provee, lugar donde está el código).*

*Para cada interfase hay que:*

* *Explicar el tipo de interacción del software relacionado con este producto de software.*
* *Definición de la interfase en términos del contenido y formato del mensaje. No es necesario detallar alguna interfase bien documentada, pero sí hacer referencia al documento que define la interfaz requerida.>*
  + 1. **Interfaces de comunicación.**

Para este sistema, no es utilizado un protocolo de comunicación específico crucial para el sistema.

* + 1. **Restricciones de memoria.**

*<Se debe especificar alguna característica aplicable y los límites de la memoria primaria y secundaria.>*

* + 1. **Operaciones.**

*<Se especifican las operaciones normales y especiales requeridas por el usuario tales como:*

* *Las distintas formas o modos de operación en la organización del usuario.*
* *Períodos de operación interactiva y períodos de operaciones desatendidas.*
* *Funciones que soporten procesamiento de datos.*
* *Operaciones de respaldo y recuperación.*

*Nota: Esto es especificado algunas veces como parte de la sección Interfaces de Usuario.>*

* + 1. **Requerimientos de adaptación del sitio.**

*<Se debe definir:*

*Los requerimientos para algún dato o inicialización de secuencia de operación que son específicos de un sitio determinado, misión, modos de operación (por ejemplo, límites de seguridad, etc.).*

*Especificar el sitio y las características de la misión que deben ser modificadas para adaptar el software a instalaciones específicas.>*

* 1. **Funciones del producto.**

*<Hacer un resumen de las funciones más importantes que el software debe ejecutar, sin mencionar los detalles de éstas funciones. Algunas veces, el resumen de funciones que es necesaria para esta parte puede ser tomado directamente de la sección de especificación de alto nivel -si existe una- que contiene funciones particulares del producto de software.*

*Las funciones deben estar organizadas de manera que se pueda hacer una lista de funciones entendibles para el cliente o para alguien que lea el documento por primera vez. Se deben usar métodos gráficos (diagramas) y de texto (cuadros sinópticos) para mostrar las diferentes funciones y las relaciones entre ellas.>*

* 1. **Usuarios.**
     1. **Características generales.**

*<Se describen las características generales del usuario deseado del producto, se incluye su nivel de estudios, experiencia y su profesionalismo técnico. No se debe utilizar para enunciar requerimientos específicos, pero se deben dar razones del por qué ciertos requerimientos son especificados más adelante (sección 3).*

* + 1. **Clases.**

*<Identificar las diversas clases de usuarios que usarán el producto. Las clases pueden ser diferenciadas por frecuencia de uso, subconjunto de funciones utilizadas, experiencia técnica, niveles de privilegios o seguridad, nivel de educación o experiencia. Distinguir los más importantes de los menos.>*

|  |  |
| --- | --- |
| Clase | Descripción |
| Donador |  |
| Donatario |  |

* 1. **Restricciones generales.**

*<Provee una descripción general de otros productos que limiten las opciones de los desarrolladores. Estas son:*

* *Políticas regulatorias.*
* *Limitaciones de hardware.*
* *Interfases con otras aplicaciones.*
* *Operaciones paralelas.*
* *Funciones de auditoría.*
* *Funciones de control.*
* *Requerimientos de lenguajes de alto nivel.*
* *Protocolos de comunicación.*
* *Lo crítico de la aplicación.*
* *Consideraciones de seguridad.>*
  1. **Supuestos y dependencias.**

*<Se listan cada uno de los factores que afecten los requerimientos. Estos factores no imponen restricciones de diseño al software, sino que los cambios a ellos (factores), pueden afectar los requerimientos. Los supuestos son contrarios a los hechos, por ejemplo, el uso de componentes de terceros, componentes comerciales que se planeen usar, temas relacionados al ambiente de desarrollo (equipo, SO, etc).>*

* 1. **Asignación de requerimientos.**

*<Se deben identificar los requerimientos que se implementarán en esta y futuras versiones del sistema. Los usuarios deben entender que no pueden obtener todas las características en la versión 1.0 del producto de software (si quieren que trabajen todas). Hay que acordar cuáles se implementarán primero, cuáles en segundo lugar y así. La priorización ayudará a resolver conflictos y planear las fases de entrega (utilizar punto 5).>*

1. **Requerimientos específicos**

*<Debe contener todos los requerimientos de software a un nivel suficiente de detalle que permita, a los desarrolladores, diseñar sistemas que los satisfagan, y que las personas encargadas de la verificación, puedan probar el sistema usándolos.*

*A través de esta sección, cada requerimiento especificado deberá ser externamente percibible por usuarios, operadores y otros sistemas externos. Estos requerimientos deben incluir una descripción mínima de cada entrada al sistema, cada salida del sistema, y todas las funciones que se realicen en el sistema como respuesta a una entrada o soporte de una salida. Como esta es la parte mas grande e importante de una ERS, se deben aplicar los siguientes principios:*

* *Los requerimientos específicos se deben enunciar de acuerdo a las características siguientes: correctos, no ambiguos, completos, consistentes, ordenados por importancia y/o estabilidad, verificables, modificables y rastreables.*
* *Los requerimientos específicos deben tener referencia cruzada con documentos relacionados.*
* *Todos los requerimientos deben ser identificados de manera única.*
* *Poner atención cuidadosa a la organización de los requerimientos para obtener una máxima legibilidad.*
* *Se debe mencionar al dueño del requerimiento.*
* *Los requerimientos deben ser de carácter cuantitativo y no cualitativo y capaces de ser probados. Cada requerimiento debe ser presentado con su criterio de aceptación.*
* *Se debe de justificar el requerimiento, diciendo el por qué del mismo.*

*No es necesario que exista texto entre este punto y el siguiente.>*

* 1. **Interfases externas.**

*<Se describen todas las entradas y todas las salidas del software. Se complementan las descripciones de interfases descritas en la sección 2 cuidando que no se repita la información. El contenido y formato es de la siguiente manera:*

*a)Nombre del producto.*

*b)Propósito de la descripción.*

*c)Fuente de entrada y destino de salida.*

*d)Rango valido, exactos y/o tolerables.*

*e)Unidades de medición.*

*f)Tiempo.*

*g)Relaciones de entradas/salidas.*

*h)Pantallas formatos/organización.*

*i)Ventanas formatos/organización.*

*j)Formatos de datos.*

*k)Formatos de comandos.*

*l)Mensajes finales.>*

* 1. **Funciones (requerimientos funcionales).**

*<Los requerimientos funcionales deben definir acciones fundamentales que tomarán lugar en el software, en la aceptación y procesamientos de entradas y en el procesamiento y generación de salidas. Se incluyen las siguientes:*

*a)Validación de entradas.*

*b)Secuencias de operación exactas.*

*c)Repuestas a situaciones anormales, incluyendo:*

*1.Sobreflujo.*

*2.Facilidades de comunicación.*

*3.Manipulación y recuperación de errores.*

*d)Efectos de parámetros.*

*e)Relación de salidas con las entradas, incluyendo:*

*1.Entrada/salidas de secuencias.*

*2.Fórmulas de conversión de entrada/salida.*

*Puede ser apropiado dividir los requerimientos funcionales en subfunciones o subprocesos. Esto no implica que el proceso de diseño de software deba ser dividido también.>*

* 1. **Desempeño de requerimientos (requerimientos no funcionales).**

*<Especifica los requerimientos numéricos, estáticos y dinámicos incluidos en el software o en una interacción humana con el software completo. Los requerimientos numéricos estáticos incluyen:*

*a)Número de terminales que deben ser soportadas.*

*b)Número de usuarios simultáneos que deben ser soportados.*

*c)Cantidad y tipos de información que debe ser manipulada.*

*Los requerimientos numéricos estáticos son identificados bajo una sección separada titulada "capacidad".*

*Los requerimientos numéricos y dinámicos incluyen por ejemplo, el número de transacciones y tareas y la cantidad de datos a ser procesados con o en un cierto período de tiempo para condiciones de trabajo normales y críticas. Todos estos requerimientos deben ser enunciados en términos cuantificables.*

*Nota: Las limitaciones numéricas aplicadas a funciones específicas son normalmente especificadas como parte de la descripción del apartado del procesamiento en esa función.>*

* 1. **Requerimientos de bases de datos lógicas.**

*<Se deben especificar los requerimientos lógicos para cualquier información incluida en la base de datos. Esta información debe contener lo siguiente:*

*a)Tipo de información usada por varias funciones.*

*b)Frecuencia de uso.*

*c)Capacidades de acceso.*

*d)Entidades de datos y sus relaciones.*

*e)Restricciones de integridad.*

*f)Requerimientos de retención de datos.>*

* 1. **Restricciones de diseño.**

*<Se especifican las restricciones de diseño que son impuestas por otros estándares, limitaciones de hardware, etc.>*

* + 1. **Estándares de conformidad.**

*<Se especifican los requerimientos derivados de reglas y estándares existentes. Esto debe incluir lo siguiente:*

*a)Formatos de reportes.*

*b)Nombramientos de datos.*

*c)Procedimientos contables.*

*d)Rastreo de auditoría.>*

* 1. **Documentación de usuario**

*<Listar los componentes de documentación de usuario como manuales de usuario, ayuda en línea y tutoriales que deben ser entregados con el software. Identificar los formatos y estándares de documentación.>*

* 1. **Atributos del sistema de software.**

*<Hay un número de atributos de software que pueden servir como requerimientos. Es importante que los atributos requeridos sean especificados de manera que sus logros sean objetivamente verificados.>*

* + 1. **Confiabilidad.**

*<Se especifican los factores requeridos para establecer la confiabilidad deseada del sistema de software en tiempo de entrega.>*

* + 1. **Disponibilidad.**

*<Se especifican los factores requeridos que garanticen el nivel de disponibilidad requerida por todo el sistema tales como puntos de control, recuperación y reestablecimiento.>*

* + 1. **Seguridad.**

*<Se especifican los factores que protegen al software de accesos accidentales o maliciosos, uso, modificación, destrucciones o divulgación de información no autorizada.*

*Los requerimientos específicos en esta área podrán incluir la necesidad de:*

*a)Utilizar ciertas técnicas de criptografía.*

*b)Llevar una bitácora o historia de los datos.*

*c)Asignar ciertas funciones a diferentes módulos.*

*d)Restringir la comunicación entre algunas áreas del programa.*

*e)Verificar la integridad de datos para variables críticas.*

*f) Identificación de usuarios.>*

* + 1. **Mantenimiento.**

*<Se especifican los atributos de software que asocien el fácil mantenimiento del software. Puede ser que existan ciertos requerimientos para cierto tipo de modularidad, interfases, complejidad, etc. Estos requerimientos no deben ser definidos en esta sección.>*

* + 1. **Portabilidad.**

*<Se especifican atributos de software que especifican la fácil portabilidad del software a otros servidores y/o sistemas operativos. Esta especificación debe incluir:*

*a)Porcentaje de componentes con código dependiente del host.*

*b)Porcentaje de código que es dependiente del host.*

*c)El uso de un lenguaje portable probado.*

*d)El uso de un compilador particular o conjunto de lenguajes.*

*e)Uso de un sistema operativo particular.>*

* 1. **Organización de requerimientos específicos.**

*<Para cualquier sistema el detalle de requerimientos tiende a ser extensivo. Por esta razón se recomienda que sean dadas consideraciones cuidadosas para organizarlas de manera óptima y entendible. Una vez descritos, crear una estructura usando el id de los requerimientos que permita estructurarlos –clasificarlos, jerarquizarlos-.>*

1. **Dependencias y Conflictos entre requerimientos**

*<En esta sección se presenta la matriz de conflictos y dependencias entre requerimientos, es recomendable que se de una descripción del por qué del conflicto o dependencia. Esta matriz puede esta referenciada a un archivo en Excel por ejemplo.>*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Req. 1 | Req. 2 | **...** | Req. n |
| ... |  |  |  |  |
| Req. 21 |  | 0 |  | 1 |
| Req. 42 | -1 | 1 |  | 0 |
| Req. 103 | 0 | 0 |  | 0 |
| ... |  |  |  |  |
| Req. n | 1 | 0 |  | 0 |

***<***

***0 Independiente***

***1 Dependiente***

***-1 En conflicto***

***>***

1. **Grado de Satisfacción e Instalación de los Usuarios Respecto a los Requerimientos**

*<En esta sección se presenta la matriz de satisfacción y de insatisfacción. El grado de satisfacción en caso de que el requerimiento sea implementado (1 extremadamente desinteresado, 2 no tiene interés, 3 Podría interesarle, 4 Complacido, 5 extremadamente complacido). El grado de insatisfacción si el requerimiento no es implementado (1 extremadamente sin cuidado, 2 sin cuidado, 3 neutral, 4 insatisfecho, 5 extremadamente insatisfecho).*

*Esto es el consenso –promedio o algún otro mecanismo- de las opiniones de los diversos usuarios>*

|  |  |  |
| --- | --- | --- |
|  | Satisfacción | Insatisfacción |
| Req 1 | 2 | 3 |
| Req 2 | 5 | 1 |
| ... |  |  |
| Req n | 4 | 4 |

1. **Temas o Detalles por Resolver**

*<Aquí se listan todos aquellos temas o detalles pendientes, que no han sido solucionados o no han quedado claros.>*

1. **Glosario**
2. **Apéndices**

*<Deben incluir lo siguiente:*

*a)Ejemplos de formatos de entrada salidas, descripción de estudios de análisis de costos o resultados de encuestas a usuarios.*

*b)Soporte o referencias de información que puedan ayudar a los lectores de la ERS*

*c)Descripción de problemas a ser resueltos por el software.*

*d)Empaquetamientos especiales del código y los medios para encontrar la seguridad, exportación, carga inicial u otros requerimientos.>*