Requerimientos de software

Servicio social

Universidad Veracruzana

Miguel Angel Acosta, Omar Tehuactle

Tabla de contenido

[1 Introducción 2](#_Toc435080194)

[1.1 Propósito 2](#_Toc435080195)

[1.2 Alcance 2](#_Toc435080196)

[1.3 Apreciación global 3](#_Toc435080197)

[2 Descripción Global 3](#_Toc435080198)

[2.1 Perspectivas del producto 3](#_Toc435080199)

[2.2 Funciones del producto 3](#_Toc435080200)

[2.2.1 Registro de proyectos 3](#_Toc435080201)

[2.2.2 Registro de asistencias y actividades 4](#_Toc435080202)

[2.2.3 Registro de estudiantes 5](#_Toc435080203)

[2.2.4 Registro de horario 6](#_Toc435080204)

[2.2.5 Registro del Responsable 6](#_Toc435080205)

[2.2.6 Crear reporte 6](#_Toc435080206)

[2.3 Características del usuario 6](#_Toc435080207)

[2.4 Restricciones 7](#_Toc435080208)

[2.5 Atenciones y dependencias 7](#_Toc435080209)

[3 Requisitos específicos 7](#_Toc435080210)

[3.1 Modo 1 7](#_Toc435080211)

[3.1.1 Registro de proyectos. 7](#_Toc435080212)

[3.1.2 Registro de asistencias y actividades 8](#_Toc435080213)

[3.1.3 Registro de estudiantes del servicio social 10](#_Toc435080214)

[3.1.4 Registro de horario 10](#_Toc435080215)

[3.1.5 Registro del Responsable 11](#_Toc435080216)

[3.1.6 Generar reporte 12](#_Toc435080217)

[3.2 Requisitos de rendimiento 12](#_Toc435080218)

[3.3 Restricciones de diseño 13](#_Toc435080219)

[3.4 Atributos del sistema 13](#_Toc435080220)

[3.4.1 Seguridad 13](#_Toc435080221)

[3.4.2 Mantenibilidad 13](#_Toc435080222)

[3.4.3 Portabilidad 14](#_Toc435080223)

[3.4.4 Fiabilidad 14](#_Toc435080224)

[Ilustración 3‑1 5](#_Toc435079082)

[Ilustración 3‑2 6](#_Toc435079083)

[Ilustración 3‑3 6](#_Toc435079084)

[Ilustración 3‑4 7](#_Toc435079085)

# Introducción

En este documento se describirán los requerimientos de software que se levantaron para el sistema de Servicio social de la Universidad veracruzana.

En estos requerimientos se muestran los diferentes casos de uso y distintos modelados que se utilizaron para el levantamiento de los requerimientos de software. Cabe mencionar que en este documento no se adjuntan en su totalidad los requerimientos que se pretenden en el sistema.

## Propósito

El sistema de control del servicio social permitirá a los estudiantes que cursan el servicio social llevar a cabo los trámites que son necesarios realizar durante el transcurso de éste de manera electrónica, logrando con esto una mayor eficacia y rapidez.

## Alcance

Este sistema está en su mayoría orientado hacia los alumnos que se encuentran cursando el servicio social, aunque también se tomaron otros actores que tendrían interacción con el sistema, como son los Responsables de revisar los reportes mensuales que son entregados por los estudiantes que cursan el servicio social.

Dara a los Responsables la oportunidad de registrarse de manera electrónica, dándole al sistema de control de servicio social información para la asignación de un estudiante que iniciará el proceso del servicio social.

## Apreciación global

En sistema se podrán registrar de manera más eficaz las actividades que el estudiante lleva a cabo, agregando una descripción de la actividad que realizó.

El SRS está organizado siguiendo los lineamientos de la plantilla del IEEE830, Plantilla organizada por el modo: Versión 2.

# Descripción Global

## Perspectivas del producto

El producto está diseñado de manera independiente y es autónomo, aunque requiere de la base de datos de los estudiantes de la universidad veracruzana para evitar un proceso de registro en donde sea necesario recaudar toda la información del estudiante.

## Funciones del producto

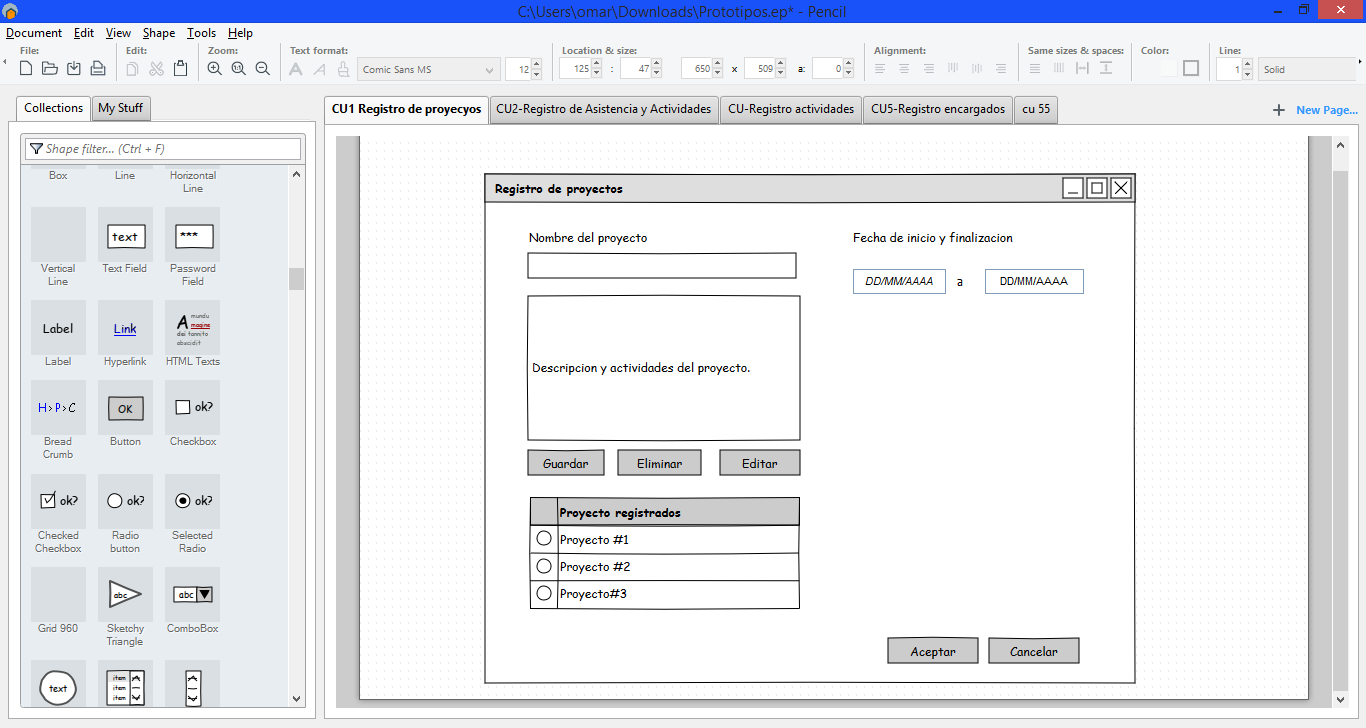
En el sistema se dan funcionalidades que ayudan al estudiante a llevar a cabo los trámites necesarios durante el proceso del servicio social. En la ilustración 3-1 se muestra el diagrama de casos de uso donde se muestra las funcionalidades del sistema. Las funciones se describen mejor a continuación.



Ilustración 1

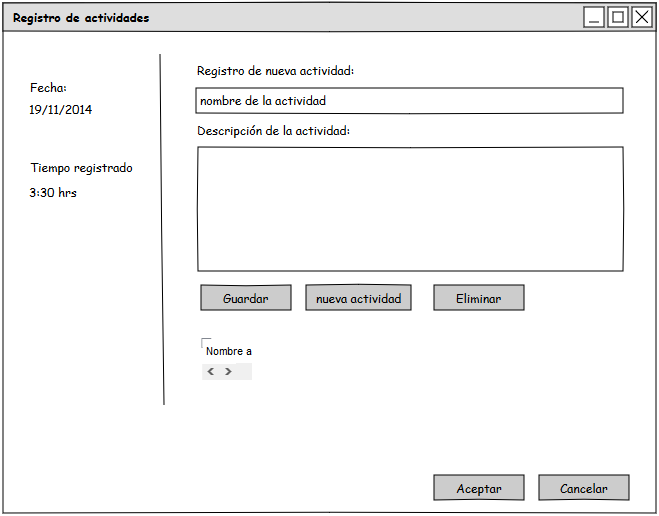
### Registro de proyectos

En este requisito funcional el alumno tendrá una interfaz (ilustración 2) donde podrá registrar los proyectos en los que se desarrollará durante su proceso de servicio social.

Ilustración 2

### Registro de asistencias y actividades

En esta funcionalidad el usuario alumno ingresara sus actividades y las horas de ingreso y de salida. En la ilustración 3-2 se muestra la interfaz donde el alumno registrará tanto su hora de llegada como su hora de salida y en la ilustración 3-3 se muestra la del registro de las actividades realizadas.

Ilustración 3Ilustración 4

### Registro de estudiantes

Esta funcionabilidad está diseñada para los estudiantes que piensan hacer el registro para cursar el servicio social.

### Registro de horario

Permite al alumno verificar y si es necesario modificar el horario generado por el sistema.

### Registro del Responsable

En esta funcionabilidad, el coordinador del servicio social podrá registrar tanto los datos del Responsable como los alumnos que están bajo su supervisión. En la imagen 3-4 se muestra una imagen de la interfaz donde el Responsable registrará sus datos.

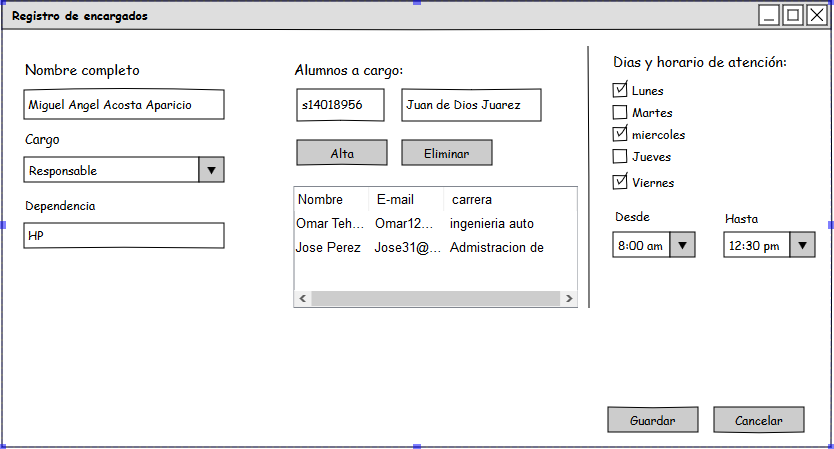


Ilustración 5

### Crear reporte

Esta función toma del sistema las actividades registradas por el usuario y las implementa en un reporte mensualmente.

## Características del usuario

El sistema no requiere de una experiencia elevada en sistemas computacionales, hace uso de interfaces básicas y fáciles de completar. No requiere un nivel educativo específico, aunque va dirigido directamente a los estudiantes universitarios de la Universidad Veracruzana que están cursando o van a iniciar el servicio social y los encargados de la verificación de sus actividades.

## Restricciones

El sistema cuenta con restricciones que limitan el sistema en apartados como la asignación de alumnos a las diferentes dependencias, ya que estos procesos son llevados a cabo por el coordinador y el comité Responsable del servicio social.

## Atenciones y dependencias

El sistema está diseñado para funcionar en el sistema operativo Windows 7, si el sistema operativo no está disponible las especificaciones de requisitos de software cambiaran.

# Requisitos específicos

## Modo 1

### Registro de proyectos.

Req 1.-El alumno registra un proyecto

Req 2.-El alumno registra una descripción del proyecto

Req 3.-El alumno define un tiempo aproximado por proyecto

En la tabla 3.1 se muestran las características completas del requisito.

Tabla 3.1

|  |  |
| --- | --- |
| **­­ID:** | CU-1 |
| **Nombre:** | Registro de proyectos |
| **Autor(es):** | Miguel Ángel Acosta, Omar Tehuactle De La Cruz |
| **Fecha de creación:** | 08/11/2015 |
| **Fecha de actualización:** | 23/11/2015 |
| **Actor(es):** | Responsable |
| **Descripción:** | El responsable deberá registrar los proyectos que se deberán llevar a cabo durante el transcurso del servicio social. En este se describirá las actividades que se deberán llevar a cabo durante el proyecto y el tiempo estimado que se espera se lleve a cabo. |
| **Precondiciones:** | Estar dado de alta en el sistema |
| **Flujo Normal:** | 1.- El responsable pone un nombre al proyecto a registrar.  2.- El responsable escribe una descripción y las actividades del proyecto en cuestión.  3.-el responsable elige el lapso de tiempo a realizarse el proyecto.  4.-El responsable guarda el proyecto.  5.-Si el responsable desea puede eliminar o editar un proyecto ya guardado.  6.-Una vez terminado el registro de proyectos, el responsable le dará click al botón aceptar o canelar ya sea para que se almacenen los datos registrado o no se guarden nada respectivamente, para salirse de la ventana y así terminar con este paso. |
| **Flujos Alternos:** |  |
| **Excepciones:** |  |
| **Poscondiciones:** | Guardar proyectos en una base de datos. |
| **Entradas:** | 1.-Nombre al proyecto.  2.-Describir y poner las actividades del proyecto.  3.-Escribir lapso del tiempo para realizar el proyecto.  4.-Guardar proyecto.  5.-Crear otro proyecto o puede editar u eliminar proyectos ya registrados anteriormente.  6.\_Dar click al botón aceptar o cancelar. |
| **Salidas:** | Mensajes:  -Botón aceptar: Sobre si se guardaron los proyectos.  -Botón Cancelar: si deberás desea no guardar lo anteriormente hecho. |
| **Incluye:**  **(relación Include)** |  |
| **Extiende:**  **(relación Extend)** |  |
| **Prioridad:** | Alta |
| **­­ID:** | CU-1 |
| **Nombre:** | Registro de proyectos |
| **Autor(es):** | Miguel Ángel Acosta, Omar Tehuactle De La Cruz |
| **Fecha de creación:** | 08/11/2015 |
| **Fecha de actualización:** | 19/11/2015 |
| **Actor(es):** | Responsable |
| **Descripción:** | El responsable deberá registrar el proyecto que se le asignara al alumno. En este se describirá las actividades que se deberán llevar a cabo durante el proyecto y el tiempo estimado que se espera se lleve a cabo. |
| **Precondiciones:** | Estar registrado en el sistema |
| **Flujo Normal:** | 1.- El responsable asigna un proyecto.  2.- El responsable registrara las actividades que se deberán llevar a cabo en el proyecto.  3.-el responsable pondrá el tiempo en el que se deberá realizar. |
| **Flujos Alternos:** | Registrar actividades sin tener un proyecto asignado. |
| **Excepciones:** |  |
| **Pos condiciones:** | Enviar correo al alumno sobre el proyecto que se le asigno.  Guardar proyecto en una base de datos |
| **Entradas:** | 1.-asignar proyecto.  2.-Encargar actividades a llevarse a cabo durante el proyecto.  2.-Registrar lapso de tiempo en el que se deberá realizar el proyecto. |
| **Salidas:** | Mensajes sobre si se registró el proyecto correctamente.  Confirmación de envió del correo |
| **Incluye:**  **(relación Include)** | Registro de estudiantes del servicio social |
| **Extiende:**  **(relación Extend)** |  |
| **Prioridad:** | Alta |

### Registro de asistencias y actividades

Req 1.- El alumno registra la hora de entrada

Req 2.- El alumno registra si tiene alguna actividad pendiente o si comenzara una nueva.

Req 3.- El alumno registra el tiempo total utilizado en la actividad.

Req 4.- El sistema lleva un conteo del tiempo.

Req 5.- El alumno registra la salida.

Req 6.- El sistema suma el tiempo al total del estudiante.

En la tabla 4.2 se muestran los detalles del requisito.

### Registro de estudiantes del servicio social

Req 1.-Que el alumno tenga como mínimo el 75% de los créditos.

Req 2.-los datos que se deben ingresar son:

-matricula.

-Nombre completo.

-carrera.

- lugar donde estudia actualmente.

- materias que cursara durante el servicio social y en que horario.

Req 3.-El alumno escogerá tres opciones de dependencia.

### Registro de horario

Req 1.-el alumno registrara su hora de entrada.

Req 2.-el alumno registrara su hora de salida.

Tabla 4.2

|  |  |
| --- | --- |
| **­­ID:** | CU-2 |
| **Nombre:** | Registro de asistencia y actividades |
| **Autor(es):** | Miguel Ángel Acosta, Omar Tehuactle De La Cruz |
| **Fecha de creación:** | 08/11/2015 |
| **Fecha de actualización:** | 19/11/2015 |
| **Actor(es):** | Alumno |
| **Descripción:** | El alumno registra en el sistema su asistencia al servicio social. Registrará la hora de entrada, salida y las actividades que llevó a cabo durante el tiempo que permaneció en ahí. |
| **Precondiciones:** | Estar registrado en el servicio social. |
| **Flujo Normal:** | 1. El alumno registra la hora de entrada presionando el botón registrar entrada. 2. El alumno registra sus actividades al final del tiempo dando clic en registro de actividades. 3. El sistema despliega otra ventana. 4. El Alumno escribe el nombre, una descripción de la actividad y el tiempo que tardo y da clic en guardar. 5. El sistema muestra una tabla con las actividades anteriores registradas. 6. El alumno da clic en aceptar y regresa a la primera ventana. 7. El alumno registra salida dando clic en registrar salida. |
| **Flujos Alternos:** | 1. El alumno registra hora de entrada dando clic en registrar entrada 2. El alumno da clic en registrar actividades. 3. El alumno registra una nueva actividad dando clic en registro de actividades. 4. El alumno registra las horas invertidas en la actividad nueva. 5. El alumno da clic en aceptar. 6. El alumno registra la salida dando clic en registrar salida. |
| **Excepciones:** | El alumno trabajo en actividades menores a 10 minutos. |
| **Poscondiciones:** | Guardas las actividades que realizo en una base de datos. |
| **Entradas:** | 1. hora de entrada. 2. actividades. 3. hora salida. |
| **Salidas:** | Las actividades han sido registradas con éxito.  ¿Deseas eliminar esta actividad? |
| **Incluye:**  **(relación Include)** | Registro de horario |
| **Extiende:**  **(relación Extend)** |  |
| **Prioridad:** | Alta |

Req 3.-El sistema calculara y guardara el tiempo hecho en el día.

Req 4.-El sistema ira acumulando las horas hechas.

### Registro del Responsable

Req 1.-El Coordinador del servicio social ingresa los datos del responsable en el sistema.

Req 2.-El coordinador le asigna un estudiante al responsable.

Req 3.-El responsable ingresa al sistema con los datos que se generaron cuando el coordinador lo registró.

Req 4.-El responsable puede ver la información de sus alumnos asignados en el sistema.

En la tabla 4.3 se muestran los detalles del requisito.

### Generar reporte

Req 1.- el sistema generara con las actividades registradas durante un mes un reporte mensual. (Siempre y cuando sobrepase las 6 horas de servicio del mes en cuestión)

## Requisitos de rendimiento

* El sistema deberá soportar una cantidad lo suficientemente grande de usuarios para que no haya saturación y provoque falla en el sistema.
* La respuesta que dará la página con respecto  a la petición del usuario deberá ser en tiempo real.
* Cada vez que se realiza una orden al sistema correctamente, todos los datos del usuario deben ser ingresados sin ninguna complicación.
* La aplicación debe funcionar sobre un sistema operativo convencional.
* Al momento de realizar un proceso, este no debe de sobrepasar el 40% del uso de los recursos de internet.
* El 95% de las operaciones deben de realizarse en menos de 1 minuto

## Restricciones de diseño

* El sistema tendrá que tener un diseño de implementación sencilla, independientemente de la plataforma o lenguaje de programación.
* Los lenguajes en que se podrá implementar, serán HTML, Javascript y java.
* No todos los usuarios tendrán los mismos accesos a determinada información.
* El sistema deberá funcionar solo teniendo acceso a internet.
* Tendrá que adaptarse a diferentes plataformas (menos dispositivos móviles).
* Se tendrá que respetar algunos de los estándares internos que la universidad permite en su página web.
* Se utilizara hardware ajeno a la pc que utilizara el Responsable para realizar firma electrónica para los reportes (note pad).
* El sistema contara con una base de datos para el registro.

|  |  |
| --- | --- |
| **­­ID:** | CU-5 |
| **Nombre:** | Registro de responsables. |
| **Autor(es):** | Miguel Ángel Acosta, Omar Tehuactle De La Cruz |
| **Fecha de creación:** | 8/11/2015. |
| **Fecha de actualización:** | 19/11/2015. |
| **Actor(es):** | Responsable. |
| **Descripción:** | El coordinador da de alta al responsable de llevar al corriente las actividades del alumno que está cursando el servicio social. |
| **Precondiciones:** | Tener asignado alumnos del servicio social. |
| **Flujo Normal:** | 1. El coordinador ingresa los datos del responsable en el sistema rellenando los campos.. 2. El coordinador asigna alumnos al encargado. 3. El coordinador da clic en guardar. |
| **Flujos Alternos:** | 1. El coordinador ingresa los datos del responsable rellenando los campos.. 2. El coordinador deja sin asignar alumnos temporalmente. 3. El coordinador da clic en guardar. |
| **Excepciones:** | El responsable se encuentra registrado con anterioridad. |
| **Poscondiciones:** | Almacenar datos del responsable en la base de datos. |
| **Entradas:** | 1.-Juan Manuel Díaz.  2.- Compaq. |
| **Salidas:** | 1.-Los datos han sido guardados. |
| **Incluye:**  **(relación Include)** |  |
| **Extiende:**  **(relación Extend)** |  |
| **Prioridad:** | Alta |

Tabla 3.3

## Atributos del sistema

### Seguridad

Este atributo es de los más importantes cuando el usuario lo intente utilizar, deberá introducir su matrícula y contraseña por lo cual el sistema deberá de comprobar de que se trata de un usuario que anteriormente estaba registrado. De lo contrario se dará una indicación de error y por consecuente no permitirá al usuario ingresar al sistema.

El sistema tendrá distintos tipos de usuarios y a cada uno de ellos se le permitirá únicamente el acceso a aquellas funciones que le correspondan.

Para prevenir una caída del sistema o pérdidas de información, el sistema hará copias de seguridad para no perder los datos.

### Mantenibilidad

Se le deberá dar mantenimiento de forma de fácil y de esta manera pueda ser corregido en caso de ser necesario

### Portabilidad

El software final debe ser soportado en diferentes equipos de cómputo sin tener problema alguno.

### Fiabilidad

El usuario debe de tener un alto grado de confianza en que el sistema va a guardar la información de forma correcta.

A continuación se agregan gráficos utilizados para priorizar y determinar los requerimientos no funcionales.

Tabla 4

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Interesado** | **Requerimientos explicitos** | **In/out** |
| **Fiable** | Usuario |  | in |
| **Robustes** | Usuario |  | in |
| **Validez** | Usuario |  | out |
| **Integridad** | Usuario |  | out |
| **Flexibilidad** | Usuario |  | in |
| **Usabilidad** | Usuario |  | out |
| **Interoperabilidad** | Usuario |  | in |
| **Eficiencia** | Usuario |  | out |
| **Testeabilidad** | Desarollador |  | in |
| **Mantenibilidad** | Desarollador |  | in |
| **Reusabilidad** | Desarollador |  | out |
| **Portabilidad** | Desarollador |  | in |

Tabla 5

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Atributo** | **In/Out** | **Score** | **Fiable** | **Robustes** | **availability** | **Validez** | **Flexibilidad** | **Usabilidad** | **Interoperabilidad** | **Eficiencia** | **Testeabilidad** | **Mantenibilidad** | **Reusabilidad** | **Portabilidad** |
| **Fiable** | in | 4 |  | < | ^ | ^ | < | ^ | < | < | ^ | ^ | ^ | ^ |
| **Robustes** | in | 4 |  |  | ^ | ^ | < | ^ | < | ^ | < | ^ | < | ^ |
| **Validez** | out | 4 |  |  |  | ^ | ^ | ^ | < | < | ^ | ^ | ^ | ^ |
| **Integridad** | out | 5 |  |  |  |  | ^ | ^ | ^ | ^ | < | ^ | ^ | < |
| **Flexibilidad** | in | 3 |  |  |  |  |  | ^ | ^ | < | ^ | ^ | ^ | ^ |
| **Usabilidad** | out | 6 |  |  |  |  |  |  | ^ | ^ | ^ | < | ^ | ^ |
| **Interoperabilidad** | in | 4 |  |  |  |  |  |  |  | ^ | ^ | ^ | < | ^ |
| **Eficiencia** | out | 6 |  |  |  |  |  |  |  |  | ^ | < | ^ | < |
| **Testeabilidad** | in | 7 |  |  |  |  |  |  |  |  |  | < | ^ | ^ |
| **Mantenibilidad** | in | 8 |  |  |  |  |  |  |  |  |  |  | < | < |
| **Reusabilidad** | out | 7 |  |  |  |  |  |  |  |  |  |  |  | ^ |
| **Portabilidad** | in | 8 |  |  |  |  |  |  |  |  |  |  |  |  |

Tabla 6

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Attribute** | **Fiable** | **Robustes** | **Valides** | **Integridad** | **Flexibilidad** | **Usabilidad** | **Interoperabilidad** | **Eficiencia** | **Testeabilidad** | **Mantenibilidad** | **Reusabilidad** | **Portabilidad** |
| **Score** | 4 | 4 | 4 | 5 | 3 | 6 | 4 | 6 | 7 | 8 | 7 | 8 |
| *Access Audit* |  |  |  | x |  |  |  |  |  |  |  |  |
| *Control de acceso* |  |  |  | x |  |  |  |  |  |  |  |  |
| **Exactitud** | x |  |  |  |  |  |  |  |  |  |  |  |
| *Independencia de aplicaciòn* |  |  |  |  |  |  |  |  |  |  | x |  |
| *Aplocacion de fiabilidad* | x |  | x |  |  |  |  |  |  |  |  |  |
| **Auditability** |  |  |  | x |  |  |  |  | x |  |  |  |
| **Costos de aumento** |  |  |  |  | x |  |  |  |  |  |  |  |
| *Comunalidad* |  |  |  |  |  |  | x |  |  |  |  |  |
| *Eficiencia de cominicaciòn* |  |  |  |  |  |  |  | x |  |  |  |  |
| **Comunicatividad** |  |  |  |  |  | x |  |  |  |  |  |  |
| **Complejidad** | x |  |  |  | x |  |  |  | x | x |  |  |
| *Recuperar fallos computacionales* | x |  | x |  |  |  |  |  |  |  |  |  |
| **Conciseness** |  |  |  |  |  |  |  | x |  | x |  |  |
| **Consistencia** | x |  |  |  | x |  |  |  |  | x |  |  |
| **Común de datos** |  |  |  |  |  |  | x |  |  |  |  |  |
| *Accesibilidad de documentos* |  |  |  |  |  |  |  |  |  | x | x |  |
| **Alojamiento de la memoria dinamica** | x |  |  |  |  |  |  |  | x | x |  |  |
| **Manejo de errores** |  | x |  |  |  | x | x |  |  |  |  |  |
| **Tolerancia de errorers** | x |  |  |  |  |  |  |  |  |  |  |  |
| **Expadibilidad** |  |  |  |  | x |  |  |  |  |  |  |  |
| *Prueba de fallos* |  |  |  |  |  |  |  |  | x |  |  |  |
| *Generalidad* |  |  |  |  | x |  | x |  |  |  | x | x |
| **GUI Standards** |  |  |  |  |  | x |  |  |  | x |  |  |
| *Consistencia de las guias* |  |  |  |  |  |  |  |  |  | x |  |  |
| *Recuperacion de errores de hardware* | x |  | x |  |  |  |  |  |  |  |  |  |
| **Independencia de hardware** |  |  |  |  |  |  |  |  |  |  | x | x |
| **Analisis de peligro** |  | x |  |  |  |  |  |  |  |  |  |  |
| **uso de codigo en linea** |  |  |  |  |  |  |  |  | x | x |  |  |
| **Instrumentación** |  |  |  | x |  |  |  |  | x | x |  |  |
| **Estandares de interfaz** |  |  |  |  |  |  | x |  |  |  |  |  |
| *Modularidad* | x |  |  |  | x |  | x |  | x | x | x | x |
| **Tamaño del modulo** |  |  |  |  | x |  |  |  | x |  |  |  |
| **MTBF** | x |  | x |  |  |  |  |  |  |  |  |  |
| **MTTR** |  |  | x |  |  |  |  |  |  |  |  |  |
| *Operabilidad* |  |  |  |  |  | x |  | x |  |  |  |  |
| *Eficiencia de uso de energia* |  |  |  |  |  |  |  | x |  |  |  |  |
| *Eficiencia de procesamiento* |  |  |  |  |  | x |  | x |  |  |  |  |
| **Tiempo de respuesta** |  |  |  |  |  | x |  | x |  |  |  |  |
| **Seguridad** |  |  |  | x |  |  |  |  |  |  |  |  |
| *Autodescriptivo* |  |  |  |  | x |  |  |  | x | x | x | x |
| *Simplicidad* | x |  |  |  | x |  |  |  | x | x | x | x |
| *Independencia de software* |  |  |  |  |  |  | x |  |  |  | x | x |
| *Eficiencia de almacenamiento* |  |  |  |  |  |  |  | x |  |  |  |  |
| *Compatibilidad de sistemas* |  |  |  |  |  |  | x |  |  |  |  |  |
| *Fiabilidad del sistema* | x |  | x |  |  |  |  |  |  |  |  |  |
| *Trazabilidad* |  |  |  |  |  |  |  |  | x | x |  |  |
| *Entrenamiento* |  |  |  |  |  | x |  |  |  |  |  |  |
| *Testeabilidad del usuario* |  |  |  |  |  |  |  |  | x |  |  |  |
| **count** | 12 | 2 | 6 | 5 | 9 | 7 | 8 | 7 | 12 | 13 | 8 | 6 |

Tabla 7

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Score** | **reliability** | **robustness** | **availability** | **integrity** | **flexibility** | **usability** | **interoperability** | **efficiency** | **testability** | **maintainability** | **reusability** | **portability** |
| **reliability** | 4 |  | **+** | **+** |  | **+** | **+** |  | **-** | **+** | **+** |  |  |
| **robustness** | 4 | **+** |  | **+** |  |  | **+** |  | **-** |  |  |  |  |
| **availability** | 4 | **+** | **+** |  |  |  |  |  |  |  |  |  |  |
| **integrity** | 5 |  |  |  |  |  | **-** | **-** | **-** | **-** |  | **-** |  |
| **flexibility** | 3 | **+** |  |  | **-** |  |  |  | **-** | **+** | **+** |  | **+** |
| **usability** | 6 |  | **+** |  |  |  |  |  | **-** | **-** |  |  |  |
| **interoperability** | 4 |  |  |  | **-** | **+** |  |  | **-** |  |  |  | **+** |
| **efficiency** | 6 | **-** | **-** |  |  | **-** | **-** | **-** |  | **-** | **-** |  | **-** |
| **testability** | 7 | **+** |  | **+** |  | **+** | **+** |  | **-** |  | **+** |  |  |
| **maintainability** | 8 | **+** |  | **+** |  | **+** |  |  | **-** | **+** |  |  |  |
| **reusability** | 7 | **-** |  |  | **-** | **+** |  | **+** | **-** | **+** | **+** |  | **+** |
| **portability** | 8 |  |  |  |  | **+** | **-** | **+** | **-** | **+** | **-** | **+** |  |