



ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE

Proyecto: iKeeper.
Revisión [].



Ficha de documento:

Fecha	Revisión	Autor	Verificado dep. calidad
13/09/2017	[Rev]	[Descripcion]	

Documento validado por las partes en fecha: 13/09/2017

Por el cliente	Por la empresa suministradora
Fdo. D./ Dña [Nombre]	Fdo. D./Dña [Nombre]



Índice

1	INTRODUCCIÓN	3
1.1	Propósito	3
1.2	Alcance	3
1.3	Personal involucrado	3
1.4	Definiciones, acrónimos y abreviaturas.....	4
1.5	Referencias.....	5
1.6	Resumen.....	5
2	DESCRIPCIÓN GENERAL	5
2.1	Perspectiva del producto.....	5
2.2	Funcionalidad del producto	5
2.3	Características de los usuarios	6
2.4	Restricciones	6
2.5	Suposiciones y dependencias.....	6
2.6	Evolución previsible del sistema	6
3	REQUISITOS ESPECÍFICOS	7
3.1	Requisitos comunes de los interfaces.....	8
3.2	Interfaces de usuario	8
3.3	Interfaces de hardware	8
3.4	Interfaces de software.....	9
3.5	Interfaces de comunicación	9
3.6	Requisitos funcionales.....	9
3.7	Requisito funcional 1	10
3.8	Requisito funcional 2	10
3.9	Requisito funcional 3	10
3.10	Requisito funcional 4	10
3.11	Requisito funcional 5	11
3.12	Requisito funcional 6	11
3.13	Requisitos no funcionales.....	11
3.14	Requisitos de rendimiento	11
3.15	Seguridad.....	11
3.16	Fiabilidad	11
3.17	Disponibilidad.....	11
3.18	Mantenibilidad	12
3.19	Portabilidad	12
3.20	Otros requisitos.....	12
4	APÉNDICES	12



1 Introducción

La Especificación de requerimiento de software en base a un proyecto, permite que proporcionar información útil para que el cliente del producto tenga una visión detallada de este, así también permitir a los desarrolladores, analistas y al resto de colaboradores contar con información necesaria del sistema para entender el sistema y servir para futuras versiones.

1.1 Propósito

- **Propósito del documento:** Detallar una serie de parámetros en base a un sistema, creando un software practico y eficiente para el usuario final o al mismo equipo de trabajo.
- **Audiencia a la que va dirigida:** A usuarios finales, ayudando a entender detalladamente todos los parámetros que integra un sistema.

1.2 Alcance

- **Identificación de producto a desarrollar:** “iKeeper”.
- “iKeeper” será un sistema qué contará con una aplicación ya sea móvil o PC qué permitirá al usuario ingresar a una cuenta personal y consultar información de coordenadas respecto a ciertas horas del día.
- **Objetivo general:** Ser la mejor marca de localizadores satelitales del mercado en México.
- **Objetivos específicos:** Brindar un excelente soporte y servicio al cliente y así nos prefieran.
 - Estar siempre al margen de las nuevas tecnologías y mantener nuestro producto actualizado para competir con el resto de fabricantes.
 - Crear un ambiente de fácil acceso a usuarios de todo tipo, creando interfaces comprensibles y de fácil acceso y manipulación.

1.3 Personal involucrado

Nombre	Daniela Beatriz Martínez Montes
Rol	Programadora y diseñadora de bases de datos
Categoría profesional	Programadora
Responsabilidades	Programar el sistema y diseño de base de datos
Información de contacto	dabe@hotmail.es
Aprobación	



Nombre	Alfredo Villegas Nava
Rol	Analista y diseñador del sistema
Categoría profesional	Programador
Responsabilidades	Diseño de la arquitectura del sistema
Información de contacto	Alfredo_v@hotmail.com
Aprobación	

1.4 Definiciones, acrónimos y abreviaturas

DEFINICIONES:

- **Sistema Operativo:** Software básico que controla una computadora. Cuenta con tres funciones: coordinar y manipular el software de la computadora (teclado, memoria, mouse, impresora...), organizar archivos de los dispositivos de almacenamiento (discos duros, discos magnéticos, entre otros).
- **Servidor:** Computadora conectada a una red que pone sus recursos a disposición del resto de los integrantes de una red.
- **Administrar:** Acción de agregar, modificar, eliminar y consultar la información de un objeto o persona.
- **Usuario:** Persona que puede ingresar a una zona privada, a través de un proceso de autenticación utilizando nombre de usuario y contraseña.
- **Administrador:** Persona encargada de ofrecer el soporte técnico y operativo al sistema.
- **Rendimiento:** resultado deseado efectivamente obtenido por un equipo o individualmente.
- **Protocolo:** Señal mediante la cual se reconoce que puede tener lugar la comunicación o la transferencia de información.
- **Login:** Nombre que se le da a una persona para permitirle el acceso a un sistema, siempre y cuando estén registrados.
- **Password:** clave o contraseña para autenticar el ingreso a un sitio.
- **Interfaz:** Medio que permite la comunicación entre usuario y el sistema.
- **Bases de datos:** Cualquier conjunto de datos organizados para su almacenamiento en la memoria de una computadora, la información se organiza en campos y registros. Campo es atributo de información. Registro es toda la información sobre un individuo.
- **Actualización:** insertar, eliminar, modificar los registros de un usuario.
- **Almacenamiento:** Cualquier dispositivo capaz de almacenar información de un sistema informático.
- **Aplicación:** Es un programa informático diseñado para facilitar al usuario la realización de un determinado tipo de trabajo.

ACRÓNIMO:

- **MySQL:** Sistema de gestión de base de datos relacional.



- **SRS:** Software Requirements Specifications (Especificación de Requerimientos de Software)

ABREVIATURAS:

1.5 Referencias

Referencia	Titulo	Ruta	Fecha	Autor
[Ref.]	[Título]	[Ruta]	[Fecha]	[Autor]

1.6 Resumen

La especificación de requisitos de Software (SRS) se clasifica de lo siguiente:

- **Descripción General:** Describe de una manera no tan a detalle de las funcionalidades del producto, sus características, versiones futuras o actualizaciones e incluso sus restricciones.
- **Requerimientos específicos:** Mostrar más a detalle todos los requerimientos del producto final, y el usuario tenga una mejor visión del producto final.

2 Descripción general

2.1 Perspectiva del producto

iKeeper trabajara en un entorno móvil para que sea más accesible para el usuario, creando un diseño practico, visual y funcional. El sistema está desarrollado de forma independiente ya que no está relacionado con otros sistemas.

2.2 Funcionalidad del producto

Resumen de funcionalidades principales de iKeeper:

- **Administración de usuarios:** El administrador podrá tener un control hacia los usuarios ya sea, eliminar, agregar, modificar y buscar.
- **Tecnología:** utiliza comunicación por satélites para obtener ubicaciones
- **OS:** utiliza exclusivamente sistema operativo Windows
- **Muestreo de locación:** El usuario tendrá la opción de visualizar en un mapa la ubicación de la hora que seleccione.
- **Eliminar campos:** El usuario tendrá la opción de eliminar los campos de la tabla que muestra el historial de coordenadas.
- **Crear cuenta:** Proceso para darse de alta en el sistema a nuevos usuarios.



- **Mapa:** Mostrar la ubicación en tiempo real del lugar en el que se encuentre el dispositivo.

2.3 Características de los usuarios

Tipo de usuario	Usuario común
Formación	Ninguna en específica
Habilidades	Manejo de la tecnología
Actividades	Entendimiento del funcionamiento de google maps

2.4 Restricciones

El sistema se desarrollará en licencias abiertas, como MySQL (Sistema de gestión de base de datos) y Visual Studio (C#), por lo tanto, no se pagará por su uso, es por eso que se hará mediante las políticas que estos sistemas establezcan.

2.5 Suposiciones y dependencias

A continuación se enlistan las suposiciones que pueden llegar a afectar:

- Los requerimientos del sistema no cambiarán durante el desarrollo de la aplicación.
- Antes de la instalación de la aplicación se debe incluir el .NET framework 4.5 o 4.6 en caso que no lo tenga instalado.
- Contar con OS Windows

2.6 Evolución previsible del sistema

Añadir más funciones, tales como un sensor de proximidad, una cámara escondida, los cuales serán activados en caso de robo



3 Requisitos específicos

- R1: Permitir tener un control de los usuarios, darlos de alta (user-password).
R2: Realizar consultas de fechas y mostrar sus respectivos datos (latitud, longitud, fecha y hora).
R3: Mostrar un mapa en el que se implemente la localización en tiempo real.
R4: conexión a red y bluetooth.
R5: base de datos.
R6: recibimiento de coordenadas.

Número de requisito	R1		
Nombre de requisito	Alta de usuarios del sistema		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Fuente del requisito			
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Número de requisito	R2		
Nombre de requisito	Realizar consultas de datos		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Fuente del requisito	R4, R5		
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Número de requisito	R3		
Nombre de requisito	Visualización de mapa		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Fuente del requisito	R4		
Prioridad del requisito	<input type="checkbox"/> Alta/Esencial	<input checked="" type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Número de requisito	R4		
Nombre de requisito	Conexión Red y Bluetooth		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Fuente del requisito			
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Número de requisito	R5		
Nombre de requisito	Base de datos		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Fuente del requisito			
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional



	Alta/Esencial	Media/Deseado	Opcional
Número de requisito	R6		
Nombre de requisito	Recibir Coordenadas		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Fuente del requisito			
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

3.1 Requisitos comunes de los interfaces

Entrada: ingreso de coordenadas vía satelital al sistema.

Salida: Mostrar los datos en pantalla, adjunto con un mapa de visualización de coordenadas.

Entrada: Registro o alta de usuarios al sistema.

Salida: guardar información en base de datos para próximas consultas.

3.2 Interfaces de usuario

El usuario debe controlar una interfaz sencilla y fácil de manipular. Dicha manipulación el usuario la realizará por medio del teclado y el Mouse (ratón).

Las interfaces de usuario en base a formularios, ayudan a cada uno de los procesos que ejecute el sistema. Por lo que dicha interfaz incluirá:

- Botones.
- Mensajes informativos.
- Mensajes de error.
- Formularios para el registro de usuarios no registrados, ingresando información personal.
- Formulario para ingresar al sistema por medio de un nombre de usuario y contraseña.
- Tabla para el muestreo de datos.
- Mapa interactivo.
- Otros.

3.3 Interfaces de hardware

La pantalla del monitor.- el software deberá mostrar información al usuario a través de la pantalla del monitor.

Ratón.- el software debe interactuar con el movimiento del ratón y los botones del ratón. El ratón se activa las zonas de entrada de datos, botones de comando y seleccione las opciones de los menús o tablas.



Teclado.- el software deberá interactuar con las pulsaciones del teclado.

Arduino.- Arduino es una plataforma para operar microcontroladores, que en este caso es indispensable su utilización para poder operar nuestros módulos.

Módulo GPS.- Este módulo será capaz de recibir coordenadas por medio de satélites, y por consiguiente enviar a nuestro sistema datos para su visualización.

Módulo Bluetooth.- el software recibirá datos del módulo bluetooth los cuales les llegan por el módulo GPS. Sirve como canal entre arduino y nuestro sistema.

3.4 Interfaces de software

Arduino

Arduino es una plataforma de prototipos electrónica de código abierto (open-source) basada en hardware y software flexibles y fáciles de usar. Está pensado para artistas, diseñadores, o para cualquiera interesado en crear objetos o entornos interactivos.

Arduino puede sentir el entorno mediante la recepción de entradas desde una variedad de sensores y puede afectar a su alrededor mediante el control de luces, motores y otros artefactos.

3.5 Interfaces de comunicación

La interfaz de comunicación entre arduino y el sistema fue a través de puertos COM, recibiendo los parámetros por medio de un módulo Bluetooth.

La interfaz entre la base de datos y el sistema, se realizó por medio de la instalación de los paquetes MySQL en nuestra plataforma, y así mismo establecer la conexión exitosa.

El sistema cuenta con conexión a internet.

El módulo GPS establezca una conexión con satélites.

3.6 Requisitos funcionales

- Visualización en GoogleMaps.
- Base de datos.
- Conexión red y bluetooth.
- Contar con una cuenta de acceso al sistema.
- Alta de usuarios.
- Consulta de datos.



3.7 Requisito funcional 1

Alta de usuarios del sistema

El sistema debe permitir el ingreso de nombre, apellido, correo electrónico, sexo, nombre de usuario y contraseña. Al detectar que el nombre de usuario está repetido, enviará un mensaje de error.

En caso de no llenar todos los campos, el sistema se encargará de recordarnos que debemos hacerlo.

Al contar con una cuenta de acceso, el sistema debe permitir el ingreso del nombre y password del usuario para realizar las diferentes funciones que tendrá cada uno. Sus estradas son nombre de usuario y contraseña. Y el sistema verificara que todos los datos ingresados sean correctos.

En caso de no llenar un campo, enviar un mensaje de error, de lo contrario, se ingresará al sistema.

3.8 Requisito funcional 2

Realizar consultas de datos

Se requiere que el sistema almacene datos recibidos, y que sea posible mandar llamar estos datos, para su visualización por medio de una tabla.

La consulta de estos datos contemplaran: fecha, hora, longitud y latitud.

En caso de no obtener estos datos, enviar un mensaje de alerta donde especifique el problema de la conexión.

3.9 Requisito funcional 3

Visualización del mapa

Se requiere que el sistema implemente la visualización del mapa por medio de un DataGrid. En caso de que el mapa no obtenga los datos necesarios, se enviará un mensaje de error al usuario.

3.10 Requisito funcional 4

Conexión Red y Bluetooth

Se requiere que el sistema reciba información del prototipo por vía Bluetooth, y que se muestren los datos en la interfaz. En caso contrario, recibir un aviso de: “no existe conexión con los puertos”.



3.11 Requisito funcional 5

Base de datos

Se requiere que el sistema almacene información en una base de datos, guardando dicha información para futuras consultas sin perder la información.

Siempre antes de iniciar sesión como usuario, verificar que ya exista una conexión con la base de datos, de lo contrario, enviar mensaje de error.

3.12 Requisito funcional 6

Recibimiento de coordenadas

El sistema debe obtener los datos satelitales en un patrón correcto, con una longitud no menor a 9 dígitos. De no ser así obtendremos valores nulos.

Los valores de entradas son 2, Longitud y latitud. Y las salidas serán estos mismos.

3.13 Requisitos no funcionales

- Diseño del sistema.
- Interfaz gráfica (color).

3.14 Requisitos de rendimiento

N/A

3.15 Seguridad

Se emplean técnicas criptográficas para proteger las cuentas de los usuarios, es decir su información. Las bases de datos solo podrán hacer uso de sentencias el usuario root, por lo tanto no todos los usuarios tendrán una manipulación de los datos.

Contar con excepciones en caso de que el usuario teclee valores inválidos.

3.16 Fiabilidad

Uno de los factores que dará iKeeper es brindar confianza al cliente, para lo cual el sistema está controlando todo tipo de excepciones y está apto a responder todo tipo de incidente.

3.17 Disponibilidad

El sistema ha sido desarrollado tomando en cuenta las necesidades, requerimientos de seguridad, misión, objetivos etc. Por lo que se encuentra disponible el 100% del tiempo del día tomando en cuenta que el día tiene 24 horas.



3.18 Mantenibilidad

El sistema va a realizar una actualización de las referencias o librerías de extensiones como el mapa de google.

Cada seis meses se va a realizar un mantenimiento preventivo, encargado de hacerlo están los desarrolladores. Una de las ventajas de nuestro sistema es que no se cobraran cuotas por mantenimiento.

3.19 Portabilidad

Una de las ventajas es los lenguajes implementados son de Software libre, lo cual no se cobra por su uso.

- 99.9% del código arduino es portable.
- Solo podrá ser utilizado en sistema operativo Windows.

3.20 Otros requisitos

N/A

4 Apéndices

[Inserte aquí el texto]

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