6-12-2016

Valentin

rAITE

Especificación de requerimientos basado en el estándar IEEE 830

RAITE



1. **Introducción**
   1. **Propósito**
   2. **Ámbito del sistema**
   3. **Personal involucrado**
   4. **Definiciones, acrónimos y abreviaturas**
   5. **Visión general de documento**
2. **Descripción general**
   1. **Perspectiva del producto**
   2. **Interfaces de pasajero**
   3. **Interfaces del conductor**
   4. **Interfaces de comunicación**
   5. **Funciones del producto**
      1. **Registro de usuario-conductor**
      2. **Interfaz de usuario-conductor**
      3. **Registro de usuario-pasajero**
      4. **Interfaz de usuario-conductor**
   6. **Restricciones**
   7. **Suposiciones y dependencias**
   8. **Requisitos futuros**
3. **Especificación de requerimientos**

**Introducción**

El presente documento es una especificación de requisitos de software para el aprovechamiento del coche y el ahorro de combustible en el trayecto rumbo a la escuela. La estructura de este documento está basada en estándar IEEE 830-1998 para la especificación de requerimientos de software.

**Propósito**

El propósito principal del actual documento es definir las especificaciones funcionales y no funcionales del sistema y su aplicación para el desarrollo del mismo. El documento va dirigido a los **usuarios** del **sistema**.

**Ámbito del sistema**

El nombre del **sistema** será “**Raite**”. El **sistema** tendrá una aplicación móvil con el mismo nombre donde permitirá a los usuarios contactar con otros usuarios que puedan ofrecer el servicio o que quieran el servicio.

La aplicación tendrá como objetivo aprovechar el consumo de combustible en un coche y ayudar con el gasto del mismo al **usuario-conductor**. El **usuario-pasajero** tendrá la oportunidad de obtener un viaje hacia la escuela de manera segura y de confianza por parte de un compañero de escuela.

**Personal involucrado**

|  |  |
| --- | --- |
| Nombre | Jesús Valentín Martínez López |
| Rol | Analista, diseñador y desarrollador |
| Categoría profesional | Ingeniero en sistemas computacionales |
| Responsabilidades | Análisis diseño y codificación del sistema |
| Correo de contacto | jesus.martinez.sc@itszapopan.edu.mx |
| Teléfono de contacto | (044) 333 118 5885 |

**Definiciones, acrónimos y abreviaturas**

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| **ERS** | Especificación de requerimientos de software |
| **GPS** | Sistema de posicionamiento global |
| **Usuario-Conductor** | Persona quien presta el servicio dueño del coche |
| **Usuario-Pasajero** | Persona quien solicita el servicio |
| **Usuarios** | Persona registrada en el sistema que pueda hacer uso de la app |
| **Pasajeros** | Personas que se encuentran dentro del vehículo sin incluir conductor |
| **Raite** | Nombre del sistema |
| **Smartphone** | Teléfono inteligente que permite la instalación de varias aplicaciones |
| **Ruta** | Camino marcado en un mapa siguiendo ciertos puntos en el |
| **RF** | Requerimiento funcional |
| **RNF** | Requerimiento no funcional |
| **Sistema** | Conjunto que lo conforma el servicio web en línea y la app para el dispositivo móvil |
| **Servicio Web** | Protocolos y estándares para intercambiar datos entre el servidor y la aplicación móvil |
| **Aplicación** | Programa informático para una tarea específica, diseñado para el beneficio del usuario. |
|  |  |

**Referencias**

|  |  |  |
| --- | --- | --- |
| **No.** | **Título** | **Publicación** |
| **1** | **IEEE 830** | **1998** |
| **2** | **IEEE 610** | **1990** |

**Visión general de documento**

La ERS se encuentra descrita en el capítulo de introducción y está dividido en tres secciones:  
 1.- Introducción

2.- Descripción general del sistema

3.- Requerimientos funcionales y no funcionales de

**2. Descripción general**

**Perspectiva del producto**

El **sistema** “**Raite**” será diseñado a partir de un **servicio** **web** que proveerá la información esta será enviada a los dispositivos para ser ocupada por el usuario. Esta aplicación utilizará APIs, servicios de Google y la ubicación del dispositivo. La aplicación podrá ser instalado en un sistema operativo Android Lollipop 5.0 o superior. El sistema raite operará en un entorno de equipos cliente y servidor.

**Interfaces de pasajero**

* **Inicio de sesión**: Se pedirá el correo y contraseña del usuario-pasajero para ingresar a la aplicación y cargar las características de su perfil
* **Perfil**: En esta pantalla se mostrará el estatus en el que se encuentra en ese momento y también se mostraran los datos del usuario-conductor que prestará el próximo servicio

**Interfaces del conductor**

* **Inicio de sesión**: Se pedirá el correo y contraseña del usuario-conductor para ingresar a la aplicación y cargar las características de su perfil.
* **Búsqueda**: En esta pantalla el conductor podrá ver las ubicaciones de los **pasajeros** que buscan viaje en un mapa.

**Interfaces de comunicación**

* La aplicación Raite obtendrá la ubicación mediante el servicio de geolocalización.
* La aplicación se conectará al servidor que opera en XML para la transferencia de datos entre el dispositivo móvil y el servidor

**Funciones del producto**

Las funciones de la aplicación de Raite en una perspectiva general constaran de las siguientes secciones:

**Registro de usuario-conductor**

* Conductor
* Vehículo

**Interfaz de usuario-conductor**

* Visualizar el mapa con los pasajeros disponibles

**Registro de usuario-pasajero**

En este registro entra.

* Pasajero
* Ubicación

**Interfaz de usuario-conductor**

* El usuario podrá

**Restricciones**

La geolocalización que utilizara la aplicación, es asistida del propio sistema GPS del Smartphone.

La interfaz de programación de la aplicación (API) está implementado en un nivel 21, calificado para el sistema operativo Android Lollipop 5.0

Los lenguajes de programación y tecnologías aplicadas serán: HTML5, APIS Google, HTML

**Suposiciones y dependencias**

Los requerimientos aquí planteados pretenden satisfacer las aspiraciones del cliente y usuario final. Los dispositivos en los que se va a ejecutar la aplicación deben cumplir los requisitos aquí mencionadas para garantizar su buen funcionamiento.

**Requisitos futuros**

Las siguientes versiones del producto tendrá mejoras de las cuales pueden destacarse la comunicación establecida entre los usuarios vía telefónica o chat. También de manera confiable podrá hacerse la cooperación económica de apoyo al **usuario-conductor** para los gastos de combustible.

Se integrará un chat interno en el cual pueda mantener una comunicación breve entre el conductor y el peatón para fijar un punto de encuentro.

**3. Especificación de requerimientos**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Nombre** | **Como un/una** | **Quiero** | **De modo que** | **Notas** |
| RNFBD01 | Diseño de BD | Creación de una base de datos | Crear una base de datos | Se cumplan los requerimientos del sistema | Diseño de la base de datos |
| RNFBD02 | Mapeo de BD | Diagrama E-R | Estructura de tablas Orientadas a objetos | Se puedan implementar en la APP | Organización orientada a objetos |
| RNFBD03 | BD Normalización | 3 Normalización | BD para almacenar datos sin errores | Se visualice correctamente en al APP |  |
| RNFWS01 | Comunicación DB-WS | WebService | Establecer comunicación entre la app y el servicio web con la BD | La App pueda intercambiar datos con el servidor |  |
| RNFWS02 | Diseño de arquitectura MVC | WebService | Diseñar la arquitectura del MVC | El sistema tenga una estructura correcta |  |
| RNFWS03 | Codificación del WebService | WebService | Desarrollar el modelo MVC | Funcione correctamente para posteriores actualizaciones |  |
| RFPIS | Pantalla principal inicio de sesion | APP | Logo de app. caja de texto correo. caja de texto pass. boton de inicio de sesion. boton de registro | El usuario ingrese sus datos y pueda cargar su perfil. El usuario pueda abrir la pantalla de nuevo registro | Esta info sera enviada al sistema y evaluada para poder iniciar sesion y cargar su perfil. La app pueda lanzar la pantalla de nuevo registro de usuario |
| RFPRU | Registro de usuario | APP | Logo de app, cajas de texto para los datos correspondientes (nombre, apellido, correo, telefono), combo para el semestre, boton radio para conductor o pasajero, caja de texto para modelo y color del vehiculo, combo para el numero de asientos. boton para continuar registro y boton para cancelar registro | El usuario pueda ver el logo de la app, pueda ingresar sus datos para completar un registro decidiendo el tipo de usuario siendo conductor o pasajero. el usuario puede terminar el registro o cancelarlo en el momento deseado. El registro sera correcto si y solo si los datos estan completos y sean validos | El usuario podra decidir el tipo de perfil que desea. La app evaluara el tipo de usuaio y pedira los datos obligatorios segun el tipo de usuario. |
| RFLPC | Pantalla de conductor | APP | Logo de la app, boton para actualizar los puntos ativos de los pasajeros que necesitan viaje, appi google maps para mostrar los puntos de los pasajeros activos. | El usuario conductor pueda ver el logo de la app, pueda actualizar los puntos activos a traves del boton para actualizar y que pueda visualizar los puntos en el mapa. | El usuario podra actualizar las ubicaciones de los pasajeros. la app hara una peticion al servidor y el servicio web mandara respuesta con los datos de las ubicaciones de los pasajeros |
| RFLPP | Pantalla de pasajero | APP | Logo de la app, boton switch que controlara el estatus del pasajero, layout donde se mostraran los datos del conductor y datos del vehículo que prestaran el servicio, estos datos seran visibles si y solo si el pasajero es aceptado por un conductor, estos datos estaran visibles un limite de 16 horas a partir de que fue aceptdo el viaje | El usuario pueda activarse para solicitar un viaje y ver los datos del conductor y vehiculo cuando sea aceptado su solicitud, estos datos seran visibles cuando sea aceptado y tendran una duracion de 16 horas a partir de que fue aceptado | La app mostrara los datos a partir de que el servicio web le indique que el viaje fue aceptado |