|  |
| --- |
|  |

**Especificación de requisitos de software**

**Proyecto: Restful API Generator (RESTGEN)**

|  |  |  |
| --- | --- | --- |
|  |  | Febrero de 2018 |

**Ficha del documento**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Revisión** | **Autor** | **Verificado** |
|  | 1 |  | Nelson enrique Vera Parra |

Documento validado por las partes en fecha:

|  |
| --- |
| Por la universidad |
| Universidad Distrital Francisco José de Caldas |

**Contenido**

[**FICHA DEL DOCUMENTO 2**](#_gjdgxs)

[**CONTENIDO 3**](#_30j0zll)

**1** **INTRODUCCIÓN 4**

[**1.1**](#_2et92p0) **Propósito 4**

[**1.2**](#_tyjcwt) **Alcance 4**

**1.3** **Personal involucrado 4**

**1.4** **Definiciones, acrónimos y abreviaturas 5**

**1.5** **Referencias 5**

[**1.6**](#_17dp8vu) **Resumen 5**

[**2**](#_26in1rg) **DESCRIPCIÓN GENERAL 5**

**2.1** **Perspectiva del producto 6**

**2.2** **Funcionalidad del producto 7**

**2.3** **Características de los usuarios 8**

**2.4** **Restricciones 8**

**2.5** **Suposiciones y dependencias 8**

**3** **REQUISITOS ESPECÍFICOS 8**

[**3.1**](#_4i7ojhp) **Requisitos comunes de las interfaces 9**

3.1.1 Interfaces de usuario 9

[3.1.2](#_1ci93xb) Interfaces de hardware 10

[3.1.3](#_3whwml4) Interfaces de software 10

[3.1.4](#_2bn6wsx) Interfaces de comunicación 10

**3.2** **Requisitos funcionales 10**

3.2.1 Requisito funcional 1 10

[3.2.2](#_1pxezwc) Requisito funcional 2 10

[3.2.3](#_49x2ik5) Requisito funcional 3 10

[3.2.4](#_2p2csry) Requisito funcional 4 10

[3.2.5](#_147n2zr) Requisito funcional 5 10

**3.3** **Requisitos no funcionales 12**

3.3.1 Requisitos de rendimiento 12

3.3.2 Seguridad 12

3.3.3 Fiabilidad 12

3.3.4 Mantenibilidad 12

3.3.5 Portabilidad 12

1. **Introducción**

Este documento es una Especificación de Requisitos Software (ERS) para la herramienta de software llamada “Restful API Generator”. Esta especificación se ha estructurado basándose en las directrices dadas por el estándar IEEE Práctica Recomendada para Especificaciones de Requisitos Software ANSI/IEEE 830, 1998.

* 1. **Propósito**

El presente documento tiene como propósito definir las especificaciones funcionales y no funcionales para el desarrollo de una herramienta de software para la generación de APIs ( *Application Programming Interface)* RESTful la cual permite la creación de aplicaciones SPA (*Single Page Applications o Aplicaciones de una sola página)* mediante la creación de la especificación de un modelo de entidad-relación en formato YML.

* 1. **Alcance**

La presente especificación tiene aplicación en las áreas de desarrollo de software, tanto académicas como comerciales, esta herramienta permite la creación de servicios básicos de consulta, escritura y actualización de datos soportados en un modelo entidad relación. La herramienta es un punto de partida para el modelamiento de flujos de trabajo aportando como base la posiblidad de crear, actualizar, consultar y borrar información soportada en el modelo de datos definido.

Esta herramienta se limita a la creación de servicios de backend (Restful) y frontend (Componente Gráfico), la lógica de negocio y flujos de trabajo están fuera del alcance de la herramienta.

* 1. **Personal involucrado**

|  |  |
| --- | --- |
| **Nombre** | Nelson Enrique Vera Parra |
| **Rol** | Analista y diseñador |
| **Categoría Profesional** | Ingeniería |
| **Responsabilidad** | Análisis de información y diseño. |
| **Información de contacto** | neverap@udistrital.edu.co |

|  |  |
| --- | --- |
| **Nombre** | XXXXXXXXXXXXXX |
| **Rol** | Analista, diseñador y programador |
| **Categoría Profesional** | Ingeniería |
| **Responsabilidad** | Análisis de información, diseño y programación del software |
| **Información de contacto** | XXXXXXXXXXXX@correo.udistrital.edu.co |

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

|  |  |
| --- | --- |
| ***Nombre*** | ***Descripción*** |
| **Usuario** | Persona que usará el software para visualizar anotaciones de proteínas. |
| **RESTGEN** | Sistema de Información móvil para la consulta y visualización de anotaciones de proteínas. |
| **ERS** | Especificación de Requisitos Software |
| **RF** | Requerimiento Funcional |
| **API** | Requerimiento No Funcional |
| **BACKEND** | Protocolo de Transferencia de Archivos |
| **FRONTEND** | Protocolo de acceso a servidores |
| **RESTFUL** | Protocolo de comunicación web |

* 1. **Referencias**

|  |  |
| --- | --- |
| **Título del Documento** | **Referencia** |
| Standard IEEE 830 - 1998 | IEEE |

* 1. **Resumen**

Este documento consta de tres secciones. En la primera sección se realiza una introducción al mismo y se proporciona una visión general de la especificación de recursos del sistema.

En la segunda sección del documento se realiza una descripción general del sistema, con el fin de conocer las principales funciones que éste debe realizar, los datos asociados y los factores, restricciones, supuestos y dependencias que afectan al desarrollo, sin entrar en excesivos detalles.

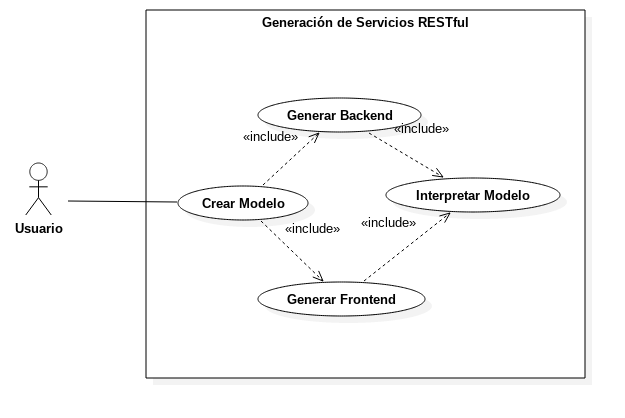
Por último, la tercera sección del documento es aquella en la que se definen detalladamente los requisitos que debe satisfacer el sistema.

1. **Descripción general**

* 1. **Perspectiva del producto**

El desarrollo de aplicaciones web en la actualidad está migrando a la arquitectura por capas de servicio, uno de los problemas iniciales al iniciar un proyecto de desarrollo web es precisamente la creación de dichos componentes. Una vez se ha establecido el modelo de datos, las clases, las entidades involucradas y sus atributos, el equipo de desarrollo tiene que optar por utilizar herramientas que ahorren tiempo y esfuerzo para responder con los tiempos, costos y cronogramas establecidos. RESTGEN es una herramienta de software que responde a la necesidad de tener un punto de partida para el desarrollo del prototipo de software sin la necesidad de escribir una sola línea de código. El equipo desarrollador tiene que enfocar su esfuerzo en la creación de un modelo de datos, que contenga las entidades, sus propiedades y restricciones, de tal manera que la herramienta interprete dicho modelo de datos y genere automáticamente los servicios de *backend* y *frontend* respectivamente.

* 1. **Funcionalidad del producto**

**

* 1. **Características de los usuarios**

|  |  |
| --- | --- |
| **Tipo de usuario** | Usuario |
| **Formación** | Formación en Ingeniería de Software, Ciencias de la computación. |
| **Actividades** | Crear el modelo en formato YML.  Generar servicios de frontend y backend. |

* 1. **Restricciones**
* La herramienta solo está soportada en sistemas operativos basados en UNIX ( GNU Linux, como Ubuntu, CentOS, Linux Mint, Debian y derivados).
* Lenguajes y tecnologías en uso: Angular 4, Python 2.7 ( Librerías Flask, flask-restless), SQLite.
* El sistema se diseñará en dos capas, Frontend ( Componente gráfico para acceder a la aplicación mediante la web) y Backend ( Compontente de servicios RESTful).

* 1. **Suposiciones y dependencias**
* Se asume que los requisitos aquí descritos son estables.
* Los equipos en los que se vaya a ejecutar el sistema deben cumplir los requisitos antes indicados para garantizar una ejecución correcta de la misma

1. **Requisitos específicos**

**Requerimientos Funcionales**

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF01 |
| **Nombre del Requerimiento:** | Interpretación del modelo |
| **Características:** | El usuario podrá generar una aplicación RESTful utilizando una descripción del modelo en formato YML. |
| **Descripción del requerimiento:** | La herramienta de software interpreta el formato YML y verifica que el formato cumpla con las reglas de sintaxis para la especificación del modelo de datos. |
| **Requerimiento NO funcional:** |  |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF02 |
| **Nombre del Requerimiento:** | Generar servicio de Frontend |
| **Características:** | Interfaz gráfica sencilla que representa el modelo descrito en el archivo con formato YML. |
| **Descripción del requerimiento:** | La herramienta generará todos los componentes gráficos para gestionar el modelo de datos representando en el archivo YML. La interfaz gráfica debe permitir: crear, actualizar, buscar y borrar información de la entidades que componen el modelo de datos. |
| **Requerimiento NO funcional:** |  |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF03 |
| **Nombre del Requerimiento:** | Generación servicio de Backend |
| **Características:** | Servicio RESTful accesible y disponible para cualquier cliente REST. |
| **Descripción del requerimiento:** | La herramienta deberá generar a partir del modelo de datos descrito en el archivo YML, todos los servicios RESTful haciendolos accesibles para cualquier cliente. |
| **Requerimiento NO funcional:** |  |
| **Prioridad del requerimiento:**  Alta | |

**Requerimientos No Funcionales.**

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RNF01 |
| **Nombre del Requerimiento:** | Interfaz de la herramienta |
| **Características:** | Accesible mediante linea de comandos. |
| **Descripción del requerimiento:** | La herramienta deberá ser accesible mediante la línea de comandos, y debe estar disponible globalmente en el sistema, permitiendo su llamado en cualquier lugar del sistema de archivos. No debe ser necesario que el usuario tenga permisos de administración sobre el equipo de desarrollo para ejecutar la herramienta. |
| **Prioridad del requerimiento:** Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RNF02 |
| **Nombre del Requerimiento:** | Disposición de los menús de la aplicación. |
| **Características:** | Entidades acccesibles a través de links en el menú principal de la aplicación. |
| **Descripción del requerimiento:** | La herramienta debe generar entradas en el menú principal de la aplicación que correspondan a las entidades que componen el modelo de datos. |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RNF03 |
| **Nombre del Requerimiento:** | Actualización en tiempo real. |
| **Características:** | Aplicación responsiva a los cambios. |
| **Descripción del requerimiento:** | La herramienta debe permitir que se actualice la información de la interfaz gráfica sin necesidad de refrescar manualmente el navegador. |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RNF04 |
| **Nombre del Requerimiento:** | Desempeño. |
| **Características:** | El sistema garantizara a los usuarios tener los recursos computacionales suficientes para que sus procesos puedan ser ejecutados. |
| **Descripción del requerimiento:** | Garantizar el desempeño del sistema informático a los diferentes usuarios. En este sentido el teléfono inteligente debe tener unos requerimientos mínimos:   * RAM disponible: 512MB o más. * Cantidad de núcleos: 1 o más. |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RNF05 |
| **Nombre del Requerimiento:** | Acceso libre. |
| **Características:** | Garantizara la posibilidad de acceder a la aplicación de manera libre. |
| **Descripción del requerimiento:** | Permitir el acceso a la plataforma sin una cuenta de usuario. La aplicación debe estar disponible en un servidor con acceso público, para la descarga. |
| **Prioridad del requerimiento:**  Alta | |

* 1. **Requisitos comunes de las interfaces**
     1. **Interfaces de usuario**

La herramienta será accesible mediante línea de comando ( consola ) y deberá estar disponible cuando ingrese el usuario al sistema en donde se encuentra instalada.

* + 1. **Interfaces de hardware**

Será necesario disponer de equipos de cómputo en perfecto estado y con el software actualizado:

* Adaptador de red.
* Procesador de mínimo: 1.0 GHz o superior con 1 o más núcleos.
* Memoria RAM mínima de 512 MB.
* Espacio mínimo de 10 MB para instalación.

* + 1. **Interfaces de software**
* Sistema Operativo: GNU Linux, o derivados de familia Unix ( Ubuntu, CentOS, Linux Mint, Debian, OpenSuse).

* + 1. **Interfaces de comunicación**

La aplicación generará los métodos de comunicación desde el frontend hacia el backend por medio de llamados soportados en el protocolo HTTP definidos en el estandar RESTful.

* 1. **Requisitos funcionales**
     1. **Requisito funcional 1**
* **Acceso de usuarios:** Los usuarios podrán acceder a cualquier funcionalidad del sistema.
* Todo usuario con acceso a la aplicación podrá utilizar sus funcionalidades.
* No se necesitan registros ni credenciales para utilizar las funcionalidades.

* + 1. **Requisito funcional 2**
* **Ejecutar consultas sobre ejemplos:** El sistema ofrecerá al usuario una lista de ejemplos, los cuales podrá consultar para probar las funcionalidades del sistema.
* El sistema presentará una lista de ejemplos para seleccionar.
* El usuario podrá seleccionar una consulta, de una lista de ejemplos para ver los resultados.

* + 1. **Requisito funcional 3**
* **Generar Menú:** El sistema creará automáticamente los menús necesarios para acceder a las entidades del modelo definidor en el archivo YML.
* El usuario podrá visualizar en el menú del encabezado el nombre de la entidad con la cual va a trabajar.

* + 1. **Requisito funcional 4**
* **Generar interfaz gráfica y servicios:** El sistema creará automáticamente las interfaces de usuario conforme al modelo provisto así como los servicios API que soportan las interacciones que se hacen en dicha interfaz.
* El usuario podrá acceder al API de la aplicación sin la necesidad de ingresar por la interfaz gráfica.
* La comunicación entre backend y frontend debe ser configurada y provista automáticamente por el sistema.

* 1. **Requisitos no funcionales**
     1. **Requisitos de rendimiento**
* Garantizar que todas las consultas que se generan en la base de datos sean eficientes y no afecten el desempeño de la base de datos y el tráfico de red.
  + 1. **Fiabilidad**
* El sistema debe tener una interfaz de uso intuitiva y sencilla
  + 1. **Mantenibilidad**
* El sistema debe disponer de una documentación fácilmente actualizable que permita realizar operaciones de mantenimiento con el menor esfuerzo posible
  + 1. **Portabilidad**
* El sistema podrá ser utilizado en cualquier sistema operativo que soporte docker.