



UNIVERSIDAD NACIONAL DE ASUNCION  
FACULTAD POLITÉCNICA



# **ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE**

**Según estándar IEEE 830**

**Sistema de Control de Acceso Para Administración  
de una Tienda Musical**

**Paradigmas de la Programación**

**Encargado de cátedra:** Lic. Gustavo Daniel Sosa  
Cabrera

**Autor:** Univ. Fernando Nicolás Formigli Báez

**San Lorenzo – Paraguay**

**Año: 2018**

# Índice

1. Introducción
  - 1.1. Propósito
  - 1.2. Ámbito del Sistema
  - 1.3. Definiciones, Acrónimos y Abreviaturas
  - 1.4. Referencias
  - 1.5. Visión General del documento
2. Descripción General
  - 2.1. Perspectiva del Producto
  - 2.2. Funciones del Producto
  - 2.3. Características de los Usuarios
  - 2.4. Restricciones
  - 2.5. Suposiciones y Dependencias
  - 2.6. Requisitos Futuros
3. Requisitos Específicos
  - 3.1. Interfaces Externas
  - 3.2. Funciones
  - 3.3. Requisitos de Rendimiento
  - 3.4. Restricciones de Diseño
  - 3.5. Atributos del Sistema
4. Apéndices
  - 4.1. Diagrama de clases

# **1. INTRODUCCIÓN**

El presente documento corresponde a una Especificación de Requisitos de Software (ERS) del Sistema de Administración de la tienda de música Bullet Song en Asunción.

Este documento fue realizado en base al estándar IEEE 830 “Recommended Practice for Software Requirements Specification ANSI/IEEE 830-1998”, cumpliendo con las directrices dictadas en la misma.

## **1.1. Propósito**

La presente ERS pretende establecer los parámetros y detalles del Sistema de Administración de una tienda de música para su posterior desarrollo, lo que posibilitará al equipo de desarrollo iniciar las actividades para la creación del sistema, además de servir como referencia ante eventuales cuestiones o dudas.

## **1.2. Ámbito del Sistema**

La finalidad de BulletSongApp es la de automatizar los procesos de registro e informes de ventas, control de existencia de productos y cálculo de salarios, correspondientes a la tienda de música Bullet Song en Asunción.

BulletSongApp se encargará de registrar operaciones efectuadas sobre los productos, ya sea venta o alquiler, pudiendo verificarse la existencia de los mismos previamente. También, BulletSongApp realizará el cálculo de salarios de los empleados de acuerdo a su función (empleado, vendedor y bonificado).

El objetivo principal del sistema es el de agilizar y realizar los procesos antes mencionados, necesarios para la administración factible de la empresa.

## **1.3. Definiciones, Acrónimos y Abreviaturas**

- Protocolo: Procedimiento el cual es realizado dependiendo de la función a llevarse a cabo.
- BulletSongApp: Nombre del software, hace referencia al nombre de la tienda.
- ERS: Especificación de Requisitos de Software
- POO: Programación Orientada a Objetos.

## **1.4. Referencias Título del documento:**

Standard IEEE 830 – 1993 - Recommended Practice for Software Requirements Specification ANSI/IEEE 830-1998. Referencia: IEEE.

## **1.5. Visión General del Documento**

Esta Especificación de Requisitos de Software se constituye de tres secciones principales. La primera de ellas se centra en la introducción a toda la ERS con una visión general del sistema. Posteriormente, en la segunda parte se trata al software en sí, con una descripción general, describiendo quiénes actúan en todo el sistema, con sus respectivos datos, y qué funciones se harán sobre ellos, además de otros datos y condiciones que podrían afectar el desarrollo del mismo. La tercera y última parte se centra en detallar y definir los requisitos que deberán de ser satisfechos por el software a desarrollar.

## **2. DESCRIPCION GENERAL BulletSongApp**

Nace a partir de la necesidad de agilizar y realizar de manera más sencilla los procesos de registro e informe operaciones (venta y alquiler), además del control de existencia de productos y el cálculo de salarios de empleados de la tienda de música Bullet Song de Asunción. Los datos que se pretenden almacenar son de interés directo para la Administración de la empresa; estos pueden ser datos de clientes, funcionarios y productos.

### **2.1. Perspectiva del Producto**

El sistema BulletSongApp es totalmente independiente a otros dentro de la empresa, debido a que se encargará netamente de la automatización de los registros de ventas o alquileres de productos, el control de existencia de productos, y el cálculo de salario de empleados, correspondientes a la administración, sin afectar las funciones de los demás departamentos.

### **2.2. Funciones del Producto**

El software BulletSongApp cumplirá con las siguientes funciones:

- Registrar a datos de clientes, funcionarios y productos; almacenando información de los mismos.

- Registrar ventas o alquiler de productos, se guardarán los detalles.
- Realizar el control de existencia de productos previo a la operación (venta o alquiler) a llevarse a cabo.
- Realizar el cálculo de salarios correspondientes a funcionarios (empleado, vendedor o bonificado).

### **2.3. Características de los Usuarios BulletSongApp**

Los usuarios del Software BulletSongApp serán los funcionarios de la tienda de música Bullet Song de Asunción, ya que el sistema se centra en procesos relacionados a la administración de la misma, estos tendrán una capacitación de 2 horas para la correcta utilización del Sistema.

### **2.3. Restricciones**

El software deberá ser desarrollado en base a netamente las necesidades administrativas de la empresa, ya que no se pretende el control de acceso de los funcionarios debido a que no se controlan los horarios de llegadas ni influye en las labores de los departamentos, ya que cada uno de los sectores de la Embajada del Brasil maneja su horario de acuerdo a las labores que deba de realizar. En lo que respecta al hardware, se presenta una gran limitante debido a que los futuros usuarios disponen tan solo de un computador portátil de gama baja, esto significa que el software a desarrollar debe de ser eficaz y más que nada, eficiente, además este computador deberá de tener instalado el Python3.

### **2.4. Suposiciones y Dependencias BulleSongApp**

Para el alquiler o ventas, el sistema dependerá del previo registro del producto-empleado-cliente.

Para el cálculo de salarios, el sistema dependerá de que el empleado tenga registrado su salario, comisión o bonificación.

### **2.5. Requisitos Futuros**

A posteriori, se pretende que también se pueda efectuar el alquiler de accesorios.

### **3. REQUISITOS ESPECÍFICOS**

#### **3.1 Interfaces Externas BulletSongApp**

Constará de una interfaz de usuario, sencilla y amena, con operaciones simples que faciliten la carga de los datos para cada registro, mediante campos y botones correspondientes. Dicha interfaz será totalmente independiente a las partes del programa que involucren los procedimientos críticos de verificación y carga de datos.

#### **3.2 Funciones**

El software deberá ser capaz de registrar todas las visitas de personas ajenas a la Embajada del Brasil en Asunción, excluyendo de esta manera a funcionarios internos (contratados locales y extranjeros en misión) y diplomáticos. Por cada usuario del software, es decir, cada guardia de seguridad, tendrá un nickname y contraseña correspondiente, que se utilizará para iniciar y acceder, y posteriormente realizar las funciones. En el caso de digitar mal cualquiera de los datos, se notificará del mismo al usuario. De manera predeterminada existirá un usuario administrador ante cualquier eventualidad. Por cada visitante que llegue a la Embajada del Brasil, se verificará si ya fue registrado con anterioridad, de no ser así, se registrará como nuevo visitante, obteniéndose de él los siguientes datos: número de cédula de identidad civil, o su equivalente para otros países, nombre(s), apellido(s), sexo y contacto. Cabe resaltar que ninguno de estos campos puede quedar vacíos, ya que son datos claves del visitante; en el caso que no sean cargados, se notificará al usuario del software que verifique todos los campos y que carguen los que aún estén vacíos. Por cada visita realizada, para la entrada, previamente verificada la existencia de tal visitante, se procederá a obtener los datos acerca de qué empresa o ente proviene, o si es en carácter de particular, el departamento al cual pretende visitar, a qué funcionario del mismo, la fecha y hora de la visita, y cuál de los usuarios realizó el registro de la visita. De todos los datos obtenidos para el registro de la visita, en el caso que se deje vacío el campo correspondiente a la empresa de la cual proviene, se estará haciendo referencia a que viene como particular; por otra parte, será posible dejar vacío el campo que corresponde al funcionario a visitar, pues es muy probable que el visitante no conozca a los funcionarios, pero sí desee realizar una visita; sin embargo no es posible dejar en blanco los demás

campos y se debe de notificar al usuario que verifique todos los campos y que carguen los que aún estén vacíos. Una vez, registrada la visita con los respectivos datos, dependiendo del departamento al cual se está visitando, se emitirá un mensaje al usuario encargado, que en este caso es el guardia de seguridad, notificando acerca del protocolo correspondiente de modo a proceder; dichos protocolos figuran de la siguiente manera: - En el caso de visita al Embajador o al Ministro, se notificará que se debe de avisar a la correspondiente secretaria y que la misma verifique si tiene una visita agendada y aprobada. - En el caso de que la visita sea a algún miembro del cuerpo diplomático, se notificará que se debe de contactar directamente, vía telefónica, al diplomático en cuestión para que se acerque hasta la portería a recibir dicha visita.

- En el caso de que la visita sea a algún funcionario de algún departamento del ente, excluyendo a los del cuerpo diplomático, pero incluyendo también a las agregadurías, se notificará que se debe de contactar con algún miembro cualquiera de dicho departamento o agregaduría, para que se acerque hasta la portería a recibir dicha visita. Una vez culminada la visita, se debe almacenar la hora de salida del determinado visitante, para lo cual deberá de existir una función para marcar la salida de una correspondiente visita, mediante un listado enumerado de las visitas actuales. En el caso de digitar un valor no existente o incorrecto, se notificará al usuario que el valor de opción es incorrecto, y que vuelva a digitarlo para finiquitar una visita. Respecto a los informes, se deberán de poder emitir informes sobre las visitas al ente de dos maneras; la primera, digitando un determinado mes de un determinado año, validando solo si el mes está dentro del rango, se proveerá de todas las visitas correspondientes a ese período de tiempo sin distinción alguna; la segunda, digitando el número de cédula de identidad (o equivalente para otros países) del visitante, obteniéndose todas las visitas realizadas por la determinada persona desde que se tienen registros. Cabe resaltar, que los datos de los guardias, las visitas y los visitantes, serán persistidos mediante Pickle, incluido en Python.

### **3.3 Requisitos de Rendimiento**

El software BulletSongApp registrará mensualmente entre 1000 y 1200 productos y se realizarán 2000 transacciones entre ventas y alquiler.

Para poder vender un producto se considera el precio de venta por la cantidad. En el caso del alquiler, tiene un costo base por un determinado periodo de tiempo para la devolución; pasado este periodo se aplica un recargo en concepto de multa.

Para el cálculo de salarios de empleados se considera siempre el salario base, y, en caso de que sea vendedor se le adiciona un porcentaje del salario base como comisión. En el caso del bonificado se le adiciona su bonificación.

### **3.4 Restricciones de Diseño**

Las limitaciones encontradas, al momento de diseñar el software en sí, se centran básicamente en los recursos computacionales existentes en la empresa (equipos de baja gama).

### **3.5 Atributos del Sistema**

La calidad del este software estará marcada por el cumplimiento de los principales factores de calidad en lo que respecta al desarrollo, los cuales son: correctitud, robustez, reutilización, extensibilidad, eficiencia y facilidad de uso para el usuario. Por otra parte, facilidad en la legibilidad del código para el encargado del mantenimiento del software, debido al diseño en base al paradigma de la programación orientada a objetos (POO).

## **4. APÉNDICE**

### **4.1. Diagrama de Clases**