**SISTEMA DE GESTIÓN DE ESTACIONAMIENTO (SIGE)**

**Versión 1.0**

**Fecha: [08/01/2025]**

**CONTROL DE VERSIONES**

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Fecha** | **Responsable** | **Descripción** |
| 1.0 | 08/01/2025 | Jhon Erick Jara Cardenas | Versión Final del documento |

**TABLA DE CONTENIDOS**

1. VISIÓN DEL PROYECTO
2. ALCANCE Y OBJETIVOS
3. ARQUITECTURA DEL SISTEMA
4. HISTORIAS DE USUARIO
5. ANEXOS
6. **VISIÓN DEL PROYECTO**
   1. **DESCRIPCIÓN GENERAL**

El Sistema de Gestión de Estacionamientos (SIGE) es una solución integral diseñada para automatizar y optimizar los procesos relacionados con la gestión de cocheras o garajes. Este sistema facilita la administración de vehículos, clientes, tarifas, y generación de boletos, mejorando la eficiencia operativa y la experiencia del usuario.

* 1. **OBJETIVOS PRINCIPALES**
* Automatizar los procesos de registro de entrada y salida de vehículos.
* Facilitar la gestión de tarifas y generación de boletos.
* Mejorar el control y seguimiento de clientes y vehículos.
* Optimizar la administración de recursos del estacionamiento.
* Proporcionar análisis y reportes en tiempo real sobre la operación del estacionamiento
  1. **STAKEHOLDERS**
* Clientes del estacionamiento.
* Personal administrativo del garaje.
* Propietarios/gestores del estacionamiento

1. **ALCANCE Y OBJETIVOS**
   1. **MÓDULOS DEL SISTEMA**
      1. **Gestión de Vehículos**

* Registro y actualización de datos de vehículos.
* Control de entrada y salida.
* Cálculo automático de tarifas según tiempo de permanencia.
* Generación de boletos electrónicos o físicos.

**2.1.2. Gestión de Clientes**

* Registro y mantenimiento de información de clientes frecuentes.
* Asignación de vehículos a clientes.
* Historial de visitas y pagos.
* Portal del cliente para consultas y pagos en línea.

**2.1.3. Gestión de Estacionamientos**

* Configuración de áreas de estacionamiento.
* Monitoreo en tiempo real de espacios disponibles.
* Asignación y control de reservas de espacios.
* Estadísticas sobre ocupación y uso del estacionamiento.

**2.1.4. Gestión Administrativa**

* Configuración de tarifas y promociones.
* Control de pagos y facturación.
* Generación de reportes financieros y operativos.
* Configuración del sistema y roles de usuario.

1. **ARQUITECTURA DEL SISTEMA**
   1. **STACK TECNOLÓGICO**

**Frontend:**

* HTML y CSS para la estructura y diseño.
* Bootstrap para un diseño responsivo y componentes reutilizables.
* JavaScript para interactividad en el cliente.

**Backend:**

* Java utilizando el modelo MVC (Modelo-Vista-Controlador).
* Eclipse IDE como entorno de desarrollo.
* JSP/Servlets para la capa de presentación y lógica del negocio.
* Scripts de validación (JavaScript o frameworks como JQuery) para validaciones en la interfaz de usuario.

**Base de Datos**:

* MySQL Workbench 8.0 como gestor de Base de Datos.

**Infraestructura:**

* Apache Tomcat como servidor de aplicaciones para desplegar el sistema.
* GitHub para control de versiones.
* Pruebas manuales y validaciones a través de Eclipse.
  1. **REQUISITOS NO FUNCIONALES**

**Disponibilidad:**

* El sistema debe garantizar una disponibilidad del 99.9%, minimizando interrupciones en el servicio.
* Tiempo máximo de inactividad permitido: 8.76 horas al año.
* Implementar un servidor de respaldo en caso de fallos del servidor principal.

**Rendimiento:**

* Tiempo de respuesta: Las páginas deben cargarse en menos de 2 segundos bajo condiciones normales de uso.
* Capacidad del sistema: Soportar hasta 500 solicitudes concurrentes, ajustable según demanda.
* Latencia máxima permitida: 200 ms en operaciones críticas (registro de entrada/salida de vehículos).

**Seguridad:**

* Autenticación: Implementar control de acceso basado en roles y autenticación mediante tokens de sesión.
* Encriptación: Proteger datos sensibles como contraseñas y datos personales utilizando algoritmos de cifrado como AES-256 para datos almacenados.
* Auditoría: Registrar todas las acciones críticas (generación de boletos, cambios en tarifas) para auditorías futuras.

**Escalabilidad:**

* Escalabilidad horizontal: Permitir la expansión mediante el despliegue de múltiples instancias del sistema en servidores adicionales.
* Escalabilidad vertical: Incrementar los recursos del servidor principal (CPU, RAM) para la base de datos y componentes dependientes.

**Mantenibilidad:**

* Código estructurado en base al patrón Modelo-Vista-Controlador (MVC) para facilitar actualizaciones y mantenimiento.
* Utilizar herramientas de control de versiones como GitHub para gestionar cambios en el código.

**Compatibilidad:**

* Compatible con los navegadores más populares (Chrome, Firefox, Edge) y dispositivos móviles con diseño responsivo (gracias a Bootstrap).
* Soporte para múltiples bases de datos en caso de migraciones futuras.

1. **HISTORIAS DE USUARIO**

**EPIC 1: Gestión de Clientes y Vehículos**

**4.1.1. HU-001: Registro de Clientes y Vehículos (Nivel Básico)**  
**Como** empleado del estacionamiento,  
**Quiero** registrar los datos de identificación del cliente y su vehículo,  
**Para** gestionar su ingreso al estacionamiento.

**Criterios de Aceptación:**

1. Permitir ingresar:
   * Nombre completo y número de identificación del cliente.
   * Placa del vehículo y descripción opcional (marca, modelo, color).
2. Validar que la placa del vehículo no esté ya registrada como activa en el sistema.
3. Generar un comprobante digital o impreso con los datos ingresados.
4. Registrar automáticamente la fecha y hora de entrada del vehículo.

**Estimación:** 3 puntos  
**Prioridad:** Alta

**4.2 EPIC 2: Gestión de Pagos**

**4.2.1. HU-002: Calculo y Cobro de Tarifas (Nivel Medio)**  
**Como** empleado del estacionamiento,  
**Quiero** calcular el monto total a cobrar al cliente según el tiempo de estadía,  
**Para** emitir un recibo y completar la transacción.

**Criterios de Aceptación:**

1. Permitir seleccionar:
   * Período de estadía (diario, nocturno, mensual).
   * Descuentos aplicables (si los hubiera).
2. Mostrar automáticamente el monto total calculado según las tarifas configuradas.
3. Registrar el pago en el sistema, indicando:
   * Método de pago (efectivo, tarjeta, transferencia).
   * Fecha y hora de la transacción.
4. Emitir un recibo digital o impreso con el desglose del cobro.

**Estimación:** 5 puntos  
**Prioridad:** Alta

**4.3 EPIC 3: Gestión de Espacios de Estacionamiento**

**4.3.1. HU-003: Asignación de Espacios (Nivel Básico)**  
**Como** empleado del estacionamiento,  
**Quiero** asignar un espacio de estacionamiento al vehículo registrado,  
**Para** mantener un control de los espacios disponibles y ocupados.

**Criterios de Aceptación:**

1. Mostrar en tiempo real el estado de los espacios (disponible/ocupado/reservado).
2. Permitir asignar manualmente un espacio según disponibilidad.
3. Registrar automáticamente el estado del espacio como "ocupado" al completar la asignación.
4. Mostrar alertas si no hay espacios disponibles.

**Estimación:** 3 puntos  
**Prioridad:** Alta

**4.4 EPIC 4: Reportes y Análisis**

**4.4.1. HU-004: Reporte de Ingresos (Nivel Complejo)**  
**Como** administrador del sistema,  
**Quiero** generar reportes de ingresos por períodos específicos,  
**Para** analizar el rendimiento del estacionamiento.

**Criterios de Aceptación:**

1. Mostrar detalles de ingresos por:
   * Día, noche, y periodos mensuales.
   * Métodos de pago.
2. Permitir exportar los reportes en formato PDF y Excel.
3. Generar gráficos para visualizar tendencias de ingresos.
4. Configurar envíos automáticos de reportes al correo del gerente/superadmin.

**Estimación:** 8 puntos  
**Prioridad:** Media

**4.5 EPIC 5: Gestión Administrativa (Solo Superadmin)**

**4.5.1. HU-005: Configuración de Tarifas (Nivel Medio)**  
**Como** superadministrador del sistema,  
**Quiero** configurar las tarifas para cada tipo de servicio (día, noche, mensual),  
**Para** mantener actualizados los montos a cobrar según políticas del negocio.

**Criterios de Aceptación:**

1. Permitir definir:
   * Tarifas base por tipo de servicio.
   * Tarifas especiales (descuentos o promociones).
2. Registrar automáticamente los cambios realizados, indicando fecha y usuario.
3. Mostrar alertas si las tarifas configuradas son inconsistentes (ej. valores negativos).

**Estimación:** 5 puntos  
**Prioridad:** Alta

**4.5.2. HU-006: Gestión de Usuarios del Sistema (Nivel Complejo)**  
**Como** superadministrador,  
**Quiero** gestionar los roles y permisos de los empleados que usan el sistema,  
**Para** garantizar un acceso controlado y seguro a las funciones del sistema.

**Criterios de Aceptación:**

1. Permitir crear, editar y desactivar usuarios del sistema.
2. Asignar roles específicos:
   * Empleado: Registro de clientes y cobros.
   * Administrador: Generación de reportes y auditorías.
   * Superadmin: Configuración avanzada y gestión total.
3. Registrar un historial de las actividades realizadas por cada usuario.

**Estimación:** 8 puntos  
**Prioridad:** Alta

**Priorización:**

* Alta: Crítico para el MVP
* Media: Importante pero no bloqueante
* Baja: Deseable, pero puede esperar

**Estimación:**

* 1-2 puntos: Tarea simple
* 3-5 puntos: Complejidad media
* 8 puntos: Tarea compleja
* 13+ puntos: Requiere división

1. **ANEXOS**
   1. **DIAGRAMAS**

* Arquitectura del sistema
* Modelo de datos
  1. **MOCKUPS**
* Interfaces principales

**GITHUB**

* CÓDIGO FUENTE
* DOCUMENTACION
* INFORME
* BD (SCRIPT)
* MANUAL DE INSTALACIÓN Y CONFIGURACIÓN