# 📊 Sistema OEE Fábrica de Pasta Rosana C.A.

# **Documento de Control de Progreso y Objetivos**

Versión: 1.0

Fecha: 8 de Enero 2025

Estado: Desarrollo Inicial - Backend Core Implementado

# **© OBJETIVO GENERAL DEL SISTEMA**

Desarrollar un sistema web completo para monitoreo y gestión de indicadores OEE (Overall Equipment Effectiveness) que reemplace el actual sistema basado en Excel, proporcionando:

- Registro digital de datos de producción por turnos
- Cálculo automático de indicadores OEE en tiempo real
- Dashboards interactivos para supervisión
- Reportes históricos y análisis de tendencias
- Alertas automáticas para desviaciones críticas
- Gestión de usuarios con roles diferenciados

# 🏭 CONTEXTO EMPRESARIAL

**Empresa:** Fábrica de Pasta Rosana C.A.

**Departamento:** Gerencia de Operaciones

### Áreas de Producción Monitoreadas:

- 1. Empaque Cobra Línea principal de empaque
  - Capacidad: 2,500 bultos/turno (teórica), 2,300 bultos/turno (real)
  - Métricas: Bultos producidos, KG totales, formato de producto
- 2. Empaque P-10 Línea secundaria de empaque
  - Capacidad: 2,000 bultos/turno (teórica), 1,850 bultos/turno (real)
  - Métricas: Bultos producidos, KG totales, formato de producto
- 3. **Tallarín** Línea especializada de tallarín
  - Capacidad: 1,800 bultos/turno (teórica), 1,650 bultos/turno (real)
  - Métricas: Bultos producidos, KG totales, formato de producto

- 4. Prensa Cobra Prensa principal de pasta
  - Capacidad: 1,600 KG/H (teórica), 1,283 KG/H (real)
  - Métricas: Lecturas de contador, horas de operación
- 5. Prensa P-10 Prensa secundaria de pasta
  - Capacidad: 1,100 KG/H (teórica), 960 KG/H (real)
  - Métricas: Lecturas de contador, horas de operación

#### Sistema de Turnos:

• Turno A: 06:00 - 14:00 horas

• Turno B: 14:00 - 22:00 horas

Turno C: 22:00 - 06:00 horas

# TARQUITECTURA DEL SISTEMA

### **Stack Tecnológico:**

- Backend: Django 4.2+ con Django REST Framework
- Frontend: Vue.js 3 con TypeScript (cambio decidido desde React)
- Base de Datos: SQLite (desarrollo) / PostgreSQL (producción)
- Autenticación: Token-based authentication
- API: RESTful API con serialización JSON

# Estructura de Aplicaciones Django:

# **ESTADO ACTUAL - IMPLEMENTADO**

# **☑** Backend Django Core (100% Completado)

### 1. Modelo de Datos Implementado:

#### Modelo Area:

- ID único autoincremental
- Nombre descriptivo del área
- Código único (EMPAQUE\_COBRA, PRENSA\_P10, etc.)
- Tipo de área (empaque/prensa)
- Capacidad teórica (KG/H o Bultos/H)
- Capacidad real promedio
- Estado activo/inactivo
- Timestamp de creación

#### Modelo Usuario Personalizado:

- Hereda de AbstractUser de Django
- Rol del usuario (administrador/supervisor/operador/viewer)
- Área asignada (FK a Area)
- Teléfono de contacto
- Estado activo/inactivo
- Registro de último acceso

#### Modelo RegistroOEE:

- Área de producción (FK a Area)
- Fecha y turno del registro
- Usuario que registra (FK a Usuario)
- Plan de producción establecido
- Producción real alcanzada
- Horario de inicio y fin de operación
- · Observaciones del turno

#### Campos específicos para áreas de empaque:

- Formato de producto
- Producción en kilogramos

#### Campos específicos para áreas de prensa:

Lectura inicial del contador

- Lectura final del contador
- Número de paradas
- Motivo de las paradas

#### Campos calculados automáticamente:

- Disponibilidad (porcentaje)
- Rendimiento (porcentaje)
- Calidad (porcentaje, default 100%)
- OEE final (porcentaje)

#### 2. Lógica de Cálculo OEE Implementada:

#### Fórmula de Disponibilidad:

- (Horas reales de operación / Horas planificadas) × 100
- Maneja turnos que cruzan medianoche
- Límite máximo del 100%

#### Fórmula de Rendimiento:

- Para Prensas: (Producción real KG / Capacidad teórica × Horas) × 100
- Para Empaque: (Producción real bultos / Plan de producción) x 100

#### Fórmula de Calidad:

- Inicialmente 100%, ajustable manualmente
- (Producción conforme / Producción total) × 100

#### **Fórmula OEE Final:**

(Disponibilidad × Rendimiento × Calidad) / 10,000

#### 3. API REST Implementada:

#### **Endpoints de Áreas:**

- GET /api/areas/ Listar todas las áreas activas
- GET /api/areas/{id}/ Detalle de área específica
- GET /api/areas/por\_tipo/ Áreas agrupadas por tipo (empaque/prensa)
- POST/PUT/DELETE /api/areas/ CRUD completo (admin only)

#### **Endpoints de Autenticación:**

- POST /api/auth/login/ Login con username/password, retorna token
- POST /api/auth/logout/ Logout y eliminación de token

#### **Endpoints de Usuarios:**

- GET /api/usuarios/ Listar usuarios activos
- GET /api/usuarios/{id}/ Detalle de usuario específico
- POST/PUT/DELETE /api/usuarios/ CRUD completo (admin only)

#### **Endpoints de Registros:**

- GET /api/registros/ Listar registros con paginación
- POST /api/registros/ Crear nuevo registro con cálculo automático OEE
- GET /api/registros/{id}/ Detalle de registro específico
- PUT/DELETE /api/registros/{id}/ Modificar/eliminar registro
- GET /api/registros/dashboard/ KPIs agregados para dashboard

#### 4. Datos Iniciales Configurados:

- 5 áreas de producción con capacidades reales
- Usuario administrador (admin/admin123)
- Usuario operador de prueba (operador1/oper123)
- Estructura de permisos por roles

#### 5. Seguridad Implementada:

- Autenticación por tokens
- Permisos por roles de usuario
- CORS configurado para desarrollo
- Validaciones de integridad de datos

# 🔄 ESTADO ACTUAL - EN DESARROLLO

# 1

# Frontend Vue.js (0% Completado)

- Setup inicial pendiente
- Configuración TypeScript pendiente

# 📝 FUNCIONALIDADES PENDIENTES - CRÍTICAS

### 🚨 Prioridad 1 (Semanas 1-2):

#### Frontend Vue.js Básico:

- Setup inicial del proyecto Vue.js 3 + TypeScript
- 2. Configuración de router y structure de componentes
- 3. Servicios de API para comunicación con backend
- 4. Sistema de autenticación frontend
- 5. Formulario de login funcional

#### Módulo de Registro de Datos:

- 1. Formulario inteligente que adapte campos según tipo de área
- 2. Validaciones en tiempo real
- Previsualización de cálculo OEE antes de guardar
- 4. Manejo de errores y confirmaciones

#### Dashboard Básico:

- 1. KPIs principales (OEE promedio, disponibilidad, rendimiento, calidad)
- 2. Estado actual por área con indicadores visuales
- 3. Tabla de registros recientes
- 4. Alertas básicas para OEE crítico (<70%)

# Prioridad 2 (Semanas 3-4):

#### Módulo de Reportes:

- 1. Filtros por fecha, área y turno
- 2. Gráficas de tendencias históricas
- 3. Comparativo entre áreas y turnos
- 4. Análisis de pérdidas y causas raíz
- 5. Exportación a PDF y Excel

#### Panel de Administración:

- 1. Gestión completa de usuarios (CRUD)
- 2. Configuración de parámetros del sistema
- 3. Logs de actividad y auditoría
- 4. Estadísticas de uso del sistema
- 5. Backup y restauración de datos

# Prioridad 3 (Semanas 5-6):

#### Funcionalidades Avanzadas:

- 1. Sistema de alertas automáticas por email/SMS
- 2. Dashboard en tiempo real con WebSockets
- 3. Gráficas interactivas avanzadas (Plotly.js)
- 4. Análisis predictivo de tendencias
- 5. Integración con sistemas externos (ERP)

#### **Optimización Mobile/Tablet:**

- 1. PWA (Progressive Web App)
- 2. Interface optimizada para tablets industriales
- 3. Modo offline con sincronización
- 4. Escaneo de códigos QR/barras

# **III** CASOS DE USO DETALLADOS

# **CU-001: Registro de Datos de Empaque**

**Actor:** Operador de empaque

Precondición: Usuario autenticado y asignado a área de empaque

Flujo Principal:

- Operador selecciona su área de trabajo
- 2. Sistema presenta formulario específico para empaque
- 3. Operador ingresa: fecha, turno, plan de producción, producción real, formato de producto, horarios
- 4. Sistema calcula automáticamente disponibilidad, rendimiento y OEE
- 5. Sistema muestra previsualización de cálculos
- 6. Operador confirma y guarda registro

7. Sistema almacena datos y actualiza dashboard en tiempo real

#### Reglas de Negocio:

- No permitir duplicados para misma área/fecha/turno
- Validar que producción real no exceda 120% del plan
- Calcular disponibilidad basada en horas reales vs 8 horas estándar
- Rendimiento = (Producción real / Plan) × 100

### **CU-002: Registro de Datos de Prensa**

Actor: Operador de prensa

Precondición: Usuario autenticado y asignado a área de prensa

Flujo Principal:

- 1. Operador selecciona área de prensa
- 2. Sistema presenta formulario específico para prensa
- 3. Operador ingresa: fecha, turno, plan KG, lecturas inicial/final, horas de operación
- 4. Sistema calcula producción real (lectura final inicial)
- 5. Sistema calcula rendimiento vs capacidad teórica del equipo
- 6. Sistema calcula OEE final
- 7. Operador revisa cálculos y guarda

#### Reglas de Negocio:

- Lectura final debe ser mayor a lectura inicial
- Horas de operación no pueden exceder horas del turno
- Rendimiento = (Producción real KG / Capacidad teórica × Horas) × 100
- Capacidad teórica: Prensa Cobra 1600 KG/H, Prensa P-10 1100 KG/H

#### CU-003: Monitoreo de Dashboard

**Actor:** Supervisor/Gerente

**Precondición:** Usuario autenticado con permisos de visualización

Flujo Principal:

- Usuario accede al dashboard principal
- 2. Sistema muestra KPIs en tiempo real
- 3. Sistema presenta estado actual de todas las áreas

- 4. Sistema genera alertas visuales para OEE crítico
- 5. Usuario puede hacer drill-down por área específica
- 6. Sistema muestra tendencias y comparativos

#### Reglas de Negocio:

- OEE ≥85% = Verde (Excelente)
- OEE 70-84% = Amarillo (Aceptable)
- OEE <70% = Rojo (Crítico)
- Actualización automática cada 5 minutos

### **CU-004: Generación de Reportes**

**Actor:** Supervisor/Gerente

Precondición: Datos históricos disponibles

Flujo Principal:

- 1. Usuario selecciona tipo de reporte
- 2. Usuario define filtros (fechas, áreas, turnos)
- 3. Sistema procesa datos y genera análisis
- 4. Sistema presenta reporte con gráficas interactivas
- 5. Usuario puede exportar en múltiples formatos
- 6. Sistema registra la generación del reporte

#### Tipos de Reporte:

- Diario: OEE por turno y área
- Semanal: Tendencias y comparativas
- Mensual: Análisis ejecutivo con recomendaciones
- Análisis de pérdidas: Identificación de problemas recurrentes

# **© OBJETIVOS DE NEGOCIO**

# Inmediatos (3 meses):

- Reemplazar completamente el sistema Excel actual
- Reducir tiempo de cálculo de OEE de 2 horas a 5 minutos
- Eliminar errores humanos en cálculos

Visibilidad en tiempo real del estado de planta

# Mediano Plazo (6 meses):

- Incrementar OEE promedio planta del 78% actual al 85%
- Identificar y reducir principales causas de pérdidas
- Implementar sistema de alertas preventivas
- Certificación en mejores prácticas industriales

### Largo Plazo (12 meses):

- Integración con sistemas ERP existentes
- Análisis predictivo y mantenimiento preventivo
- Benchmarking con plantas similares
- ROI estimado del 15-20% anual

# MÉTRICAS DE ÉXITO

#### **Técnicas:**

- Disponibilidad del sistema: 99.5%
- Tiempo de respuesta API: <200ms
- Tiempo carga dashboard: <2 segundos</li>
- Cero pérdida de datos

## **Operacionales:**

- 100% adopción por parte de operadores
- Reducción 95% en errores de cálculo
- Tiempo de registro por turno: <5 minutos</li>
- Generación automática de reportes diarios

# Negocio:

- Incremento OEE promedio: +7 puntos porcentuales
- Reducción tiempo administrativo: 80%
- Identificación proactiva de problemas: 90%
- Ahorro anual estimado: \$150,000 USD

# **RIESGOS Y CONSIDERACIONES**

#### **Técnicos:**

- Migración de datos históricos desde Excel
- Integración con sistemas legacy existentes
- Escalabilidad para futuras plantas
- Backup y recuperación ante desastres

# **Operacionales:**

- Resistencia al cambio por parte de operadores
- Capacitación necesaria en nueva tecnología
- Dependencia de conectividad de red
- Mantenimiento y actualizaciones del sistema

## **Negocio:**

- Inversión inicial en hardware (tablets industriales)
- Tiempo de implementación y estabilización
- Costo de soporte técnico continuo
- ROI dependiente de mejoras en OEE

# PRÓXIMOS PASOS INMEDIATOS

# Paso 1: Setup Frontend Vue.js

- Crear proyecto Vue.js 3 con TypeScript
- Configurar routing y estructura de componentes
- Implementar servicios de API

#### Paso 2: Autenticación Frontend

- Formulario de login funcional
- Manejo de tokens y sesiones
- Protección de rutas por roles

## Paso 3: Formulario de Registro

· Interface adaptativa por tipo de área

- · Validaciones en tiempo real
- Integración con API de backend

#### Paso 4: Dashboard Básico

- KPIs principales con datos reales
- Estado por área con indicadores visuales
- Actualización automática de datos

# **CONTACTOS Y REFERENCIAS**

### **Stakeholders Clave:**

- Gerencia de Operaciones: Aprobación funcionalidades
- Supervisores de Turno: Validación de procesos
- Operadores: Testing de usabilidad
- IT: Infraestructura y deployment

#### **Documentación Técnica:**

- Archivo Excel original: "INDICADORES 6 oee.xlsx"
- Prototipos UI/UX: 6 componentes React desarrollados como referencia
- API Documentation: Auto-generada por Django REST Framework
- Till Última Actualización: 8 de Enero 2025
- **Próxima Revisión:** Al completar frontend básico (estimado 15 de Enero 2025)
- **Estado:** Backend Core completo, iniciando desarrollo Frontend Vue.js