

**Renata Carolina Castro Olmos**

**S22019640**

**Luis Gerardo León Salamanca**

**S22002209**

**Daniel Gutiérrez Contreras**

**S22019646**

**José Alexis González Chávez**

**S22002189**

**Othon Lozano Vidal**

**S22019632**

**Sistemas Operativos**

**M.C. Josué Shinoe Munguía Tiburcio**

**Proyecto Final**

**Mini Sistema Operativo Académico**

**Veracruz, 30 de mayo de 2025**

**ÍNDICE**

[***1. NOMBRE Y LOGO DEL SISTEMA OPERATIVO 3***](#_heading=h.vpi790isl3a7)

[**1.1 Nombre del Sistema Operativo 3**](#_heading=h.1h3gpvm6b86g)

[**1.2 Significado y Justificación del Nombre 3**](#_heading=h.uduelotbuiz3)

[**1.3 Versión Inicial 3**](#_heading=h.vwqhyk9aiwr)

[**1.4 Logo del Sistema Operativo 3**](#_heading=h.n41ifgeyn1c4)

[***2. TIPO DE SISTEMA OPERATIVO 4***](#_heading=h.de647v9nlmw8)

[**2.1 Clasificación Principal 4**](#_heading=h.jbq81vqf72q1)

[**2.2 Características Específicas 4**](#_heading=h.gvai1u746q3y)

[**2.3 Comparación con Tipos de SO Reales 5**](#_heading=h.26wq96z59hwn)

[**2.4 Justificación del Tipo Seleccionado 5**](#_heading=h.ami5a1pnht7j)

[***3. DIAGRAMA DE ARQUITECTURA GENERAL 6***](#_heading=h.srxdrtc0rjeo)

[**3.1 Arquitectura Global del Sistema 6**](#_heading=h.bnltqfhhrb32)

[**3.2 Componentes Detallados del Sistema 7**](#_heading=h.pni01fvo9trk)

[**3.3 Flujo de Datos del Sistema 8**](#_heading=h.skm8lxqrarsn)

[***4. JUSTIFICACIÓN DEL DISEÑO Y ENFOQUE PEDAGÓGICO 9***](#_heading=h.20bt86ljytz1)

[**4.1 Objetivos Educativos Específicos 9**](#_heading=h.1wn9dnqqfbw0)

[**4.2 Enfoque Pedagógico Aplicado 9**](#_heading=h.q2yyl3asig9i)

[***5. BITÁCORA DE USO DE IA 11***](#_heading=h.j98xwubgc9ac)

[**5.1 Información General del Uso de IA 11**](#_heading=h.m90e49ud1as)

[**5.2 Registro Detallado de Consultas 11**](#_heading=h.ruv36wmlucxd)

# 1. NOMBRE Y LOGO DEL SISTEMA OPERATIVO

## 1.1 Nombre del Sistema Operativo

QueSO

## 1.2 Significado y Justificación del Nombre

- "Que": Representa la pregunta “Qué” importante en el educativo del sistema

- "SO": Acrónimo de Sistema Operativo

- Combinación: Refleja un juego de palabras al igual el uso de la palabra "qué" la cual tiene relevancia en el contexto académico ya que permite establecer objetivos claros en el aprendizaje y asegurar la adquisición de saberes significativos

## 1.3 Versión Inicial

QueSO v1.0 "Pioneer"

- Nombre clave "Pioneer" simboliza el primer paso en el desarrollo de sistemas operativos educativos

## 1.4 Logo del Sistema Operativo



# 2. TIPO DE SISTEMA OPERATIVO

## 2.1 Clasificación Principal

Sistema Operativo Educativo Simulado

## 2.2 Características Específicas

2.2.1 Por su Propósito

- Tipo: Sistema Educativo/Académico

- Objetivo: Demostración y aprendizaje de conceptos fundamentales de SO

- Alcance: Prototipo funcional para fines pedagógicos

2.2.2 Por su Estructura

- Arquitectura: Monolítica Simplificada

- Núcleo: Integrado con módulos bien definidos

- Modularidad: Alta separación de responsabilidades

2.2.3 Por el Número de Usuarios

- Tipo: Multiusuario Básico

- Usuarios soportados: Administrador e Invitado

- Gestión: Sistema de autenticación con roles diferenciados

2.2.4 Por el Número de Procesos

- Tipo: Multitarea Simulada

- Planificación: Algoritmos educativos (FIFO, Round Robin, Prioridades)

- Concurrencia: Simulación de ejecución paralela

2.2.5 Por la Interfaz de Usuario

- Tipo: Híbrido (CLI + GUI)

- Terminal: Línea de comandos funcional

- Gráfica: Interfaz visual para monitoreo y gestión

2.2.6 Por el Manejo de Recursos

- Memoria: Gestión simulada con paginación básica

- Almacenamiento: Sistema de archivos jerárquico virtual

- CPU: Planificación por tiempo compartido simulado

## 2.3 Comparación con Tipos de SO Reales

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caracteristicas | EduOS | Windows | Linux | macOS |
| Proposito | Educativo | Comercial | Libre/Comercial | Comercial |
| Arquitectura | Monolitica Simple | Híbrida | Mololítica | Híbrida |
| Usuarios | Básico (2 tipos) | Completo | Completo | Completo |
| Hardware | Simulado | Real | Real | Real |
| Complejidad | Baja (educativa) | Alta | Alta | Alta |
| Portabilidad | Java (multiplataforma) | x86/x64 | Múltiple | Apple Silicon / Intel |

## 2.4 Justificación del Tipo Seleccionado

2.4.1 Ventajas del Enfoque Educativo

1. Simplicidad Conceptual: Permite enfocarse en fundamentos sin complejidades innecesarias

2. Transparencia: Todos los procesos internos son visibles y explicables

3. Modificabilidad: Fácil adaptación para diferentes escenarios educativos

4. Seguridad: Entorno controlado sin riesgos para el sistema host

2.4.2 Beneficios de la Simulación

1. Control Total: Manipulación completa del comportamiento del sistema

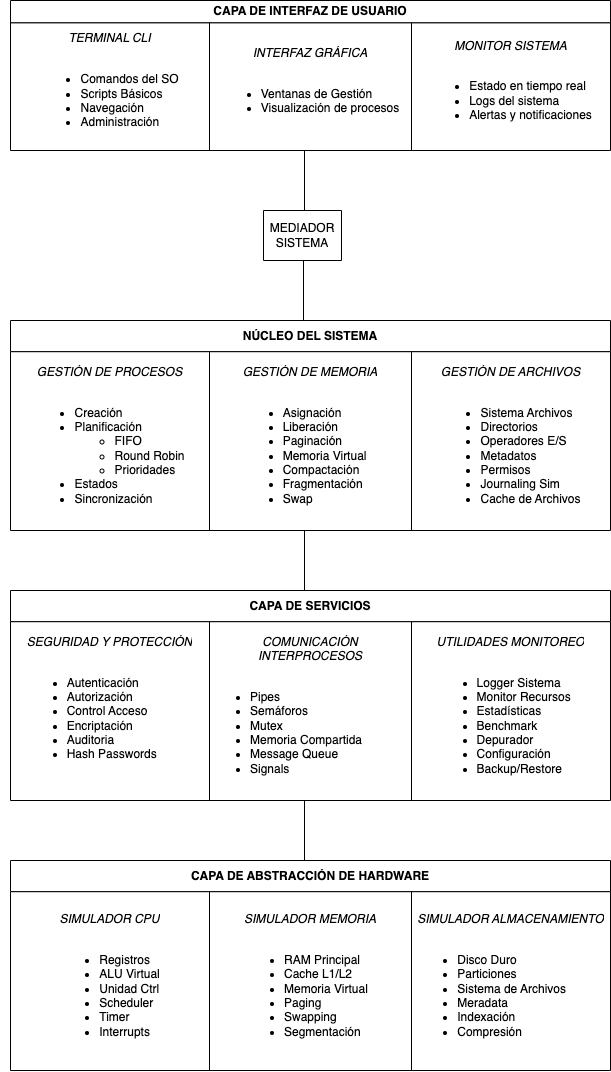
2. Depuración: Capacidad de pausar, analizar y modificar estados

3. Experimentación: Prueba de diferentes algoritmos y configuraciones

4. Reproducibilidad: Resultados consistentes para evaluación académica

# 3. DIAGRAMA DE ARQUITECTURA GENERAL

## 3.1 Arquitectura Global del Sistema



## 3.2 Componentes Detallados del Sistema

3.2.1 Capa de Interfaz de Usuario

// Componentes principales:

- InterfazUsuario.java // CLI principal

- MonitorSistema.java // GUI de monitoreo

- GestorComandos.java // Procesador de comandos

- VisualizadorProcesos.java // GUI de procesos

3.2.2 Núcleo del Sistema

// Gestión de Procesos:

- GestorProcesos.java // Controlador principal

- Proceso.java // Estructura de proceso

- Planificador.java // Algoritmos de planificación

- EstadosProceso.java // Máquina de estados

// Gestión de Memoria:

- GestorMemoria.java // Controlador de memoria

- BloqueMemoria.java // Estructura de bloques

- Paginador.java // Sistema de paginación

- MemoriaVirtual.java // Gestión virtual

// Sistema de E/S y Archivos:

- SistemaArchivos.java // Controlador de archivos

- DirectorioSimulado.java // Estructura de directorios

- ArchivoSimulado.java // Estructura de archivos

- GestorES.java // Operaciones E/S

3.2.3 Capa de Servicios

// Seguridad:

- SeguridadManager.java // Controlador seguridad

- Usuario.java // Estructura de usuario

- ControlAcceso.java // Permisos y autorización

- Encriptador.java // Funciones criptográficas

// Comunicación:

- ComunicacionIP.java // IPC básico

- Semaforo.java // Sincronización

- Cola.java // Colas de mensajes

// Utilidades:

- Logger.java // Sistema de logs

- Monitor.java // Monitoreo de recursos

- Configuracion.java // Gestión de configuración

## 3.3 Flujo de Datos del Sistema

3.3.1 Flujo de Creación de Proceso

Usuario → CLI → GestorProcesos → Planificador → GestorMemoria → Proceso Creado

↓

Logger ← Monitor ← EstadosProceso ← MemoriaAsignada ← BloqueMemoria

3.3.2 Flujo de Operación de Archivo

Usuario → Comando → SistemaArchivos → DirectorioSimulado → ArchivoSimulado

↓

SeguridadManager → ControlAcceso → VerificarPermisos → Operación Autorizada

3.3.3 Flujo de Autenticación

Usuario → Credenciales → SeguridadManager → Hash → ValidaciónBD → SesionCreada

↓

Logger → AuditoriaSeguridad → PermisosSistema → AccesoAutorizado

# 4. JUSTIFICACIÓN DEL DISEÑO Y ENFOQUE PEDAGÓGICO

## 4.1 Objetivos Educativos Específicos

4.1.1 Objetivos de Aprendizaje Conceptual

1. Comprensión de Componentes

- Identificar y explicar los módulos principales de un SO

- Entender las interacciones entre gestión de procesos, memoria y archivos

- Reconocer la importancia de la seguridad en sistemas operativos

2. Comprensión de Procesos

- Visualizar estados de procesos y transiciones

- Comparar algoritmos de planificación

- Analizar el impacto de diferentes políticas de gestión

3. Comprensión de Memoria

- Entender técnicas de asignación de memoria

- Visualizar fragmentación y compactación

- Experimentar con memoria virtual y paginación

4.1.2 Objetivos de Aprendizaje Procedimental

1. Implementación de Algoritmos

- Codificar planificadores de procesos

- Implementar algoritmos de gestión de memoria

- Desarrollar sistemas de archivos básicos

2. Resolución de Problemas

- Identificar y resolver deadlocks

- Optimizar rendimiento del sistema

- Depurar problemas de concurrencia

## 4.2 Enfoque Pedagógico Aplicado

4.2.1 Constructivismo Educativo

Principio: Los estudiantes construyen conocimiento a través de la experiencia práctica.

Aplicación en QueSO:

- Construcción gradual: Cada fase añade complejidad al sistema base

- Experimentación activa: Posibilidad de modificar parámetros y observar efectos

# 5. BITÁCORA DE USO DE IA

## 5.1 Información General del Uso de IA

Herramienta de IA seleccionada: Claude (Anthropic)

Versión: Claude Sonnet 4

Período de uso: 16 de junio de 2025

Responsable del registro: Othon Lozano Vidal

## 5.2 Registro Detallado de Consultas

5.2.1 Consulta 1: Arquitectura del Sistema

Pregunta realizada:

"Necesito diseñar la arquitectura de un sistema operativo educativo en Java. ¿Cuáles son los componentes principales que debería incluir y cómo estructurarlos de manera modular?"

Respuesta obtenida (resumen):

- Sugerencia de arquitectura en capas

- Identificación de módulos principales: procesos, memoria, archivos, seguridad

- Recomendación de patrones de diseño como Singleton y Observer

- Estructura de clases en Java

5.2.2 Consulta 2: Enfoque Pedagógico

Pregunta realizada:

"¿Qué enfoques pedagógicos son más efectivos para enseñar sistemas operativos a través de un proyecto práctico? ¿Cómo puedo justificar educativamente el diseño de mi sistema?"

Respuesta obtenida (resumen):

- Recomendación de constructivismo y ABP

- Sugerencias sobre progresión de complejidad

- Ideas para evaluación formativa y sumativa

- Estrategias de aprendizaje visual

5.2.3 Consulta 3: Implementación Técnica

Pregunta realizada:

"¿Cómo puedo implementar un simulador de gestión de procesos en Java que sea educativo pero funcional? Necesito ejemplos de código para estructuras de datos y algoritmos básicos."

Respuesta obtenida (resumen):

- Estructura de clases para procesos

- Implementación de estados de proceso

- Algoritmos básicos de planificación

- Patrones para gestión de memoria

5.2.4 Consulta 4: Documentación y Diagramas

Pregunta realizada:

"¿Cómo puedo crear diagramas de arquitectura claros y documentación técnica efectiva para un proyecto académico de sistemas operativos?"

Respuesta obtenida (resumen):

- Formatos de diagramas de arquitectura

- Estructura de documentación técnica

- Mejores prácticas para documentación académica

- Herramientas recomendadas

Link de la Conversación Básica:  
  
<https://claude.ai/share/e346e9d9-c8f1-4ad4-9336-66505303a147>