### **Proyecto: Reescritura en Programación Orientada a Objetos (POO)**

**Objetivo:** Reescribir el sistema actual (gestión de estacionamiento) usando principios de Programación Orientada a Objetos y dividir el trabajo de forma equitativa entre los integrantes del equipo.

### **🔧 Módulos Principales Identificados en POO**

1. **Usuario y Autenticación**

* Registro de usuario
* Inicio y cierre de sesión
* Recuperación y actualización de contraseña

1. **Vehículos**

* Registro y edición de vehículos
* Visualización de historial de vehículos

1. **Reservas**

* Crear, editar, eliminar reservas
* Consulta de disponibilidad
* Recordatorios y notificaciones

1. **Administración y Reportes**

* Panel de administrador
* Estadísticas de uso
* Generación de reportes PDF

### **👥 Asignación de Roles**

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Rol Principal** | **Responsabilidades POO** |
| **Rodrigo** | 🧑‍💻 Gestión de Usuarios | Clases Usuario, Autenticador, lógica de login/registro. |
| **DiegoOli** | 🚗 Módulo de Vehículos | Clases Vehiculo, HistorialVehiculo, validaciones. |
| **Daniel** | 🗓️ Gestión de Reservas | Clases Reserva, Calendario, EspacioEstacionamiento. |
| **DiegoVas** | 📊 Admin y Reportes | Clases Administrador, Reporte, Estadistica, PDFGen. |

### **💡 Recomendaciones para la Migración a POO**

* Utilizar clases para representar entidades clave: Usuario, Vehiculo, Reserva, etc.
* Encapsular lógica dentro de métodos apropiados.
* Dividir lógicamente el backend (lógica del sistema) del frontend (interfaz).
* Establecer interfaces o APIs entre los módulos para integrar el sistema.

### **📄 Tecnologías sugeridas para reimplementación (opcional)**

* Lenguaje: Python (usando Flask o Django), o PHP orientado a objetos
* Base de datos: MySQL o PostgreSQL
* Frontend: HTML, CSS, JS con Bootstrap (o framework frontend si se desea)

### **✅ Siguientes pasos sugeridos**

1. Cada integrante define las clases principales de su módulo.
2. Reunión para definir interfaces entre módulos.
3. Implementación de forma modular y pruebas individuales.
4. Integración de módulos y pruebas finales.

💭 *Documento generado para la organización del proyecto basado en la estructura original del sistema web de estacionamiento.*