**UNIVERSIDAD DE GUAYAQUIL**  
**Facultad de Ciencias Matemáticas y Físicas**  
**Carrera: Ingeniería en Software**  
**Asignatura: Construcción de Software**

**Diseño Detallado del Sistema: Gestión de Reparaciones**

**Integrantes del grupo:**

* Joffre Yagual

**Fecha:** 22 de junio de 2025

**1. Introducción**

El presente documento describe el diseño detallado del sistema web "Gestión de Reparaciones", desarrollado como parte del proyecto final de la asignatura Construcción de Software. Este sistema permite registrar clientes, gestionar reparaciones, visualizar listados y controlar el acceso mediante autenticación. Se ha diseñado utilizando buenas prácticas de desarrollo y principios de modularidad para facilitar su mantenimiento.

**2. Descripción general del sistema**

El sistema permite gestionar reparaciones de dispositivos móviles, registrar información de los clientes y llevar el seguimiento de los trabajos. Utiliza tecnologías estándar de desarrollo web:

* Lenguaje de programación: PHP
* Base de datos: PostgreSQL
* Administración de BD: pgAdmin4
* Interfaz: HTML + CSS básico
* Servidor local: XAMPP

**3. Arquitectura del sistema**

* **Cliente:** Navegador web
* **Servidor:** Apache (XAMPP)
* **Back-end:** PHP (sin frameworks)
* **Base de datos:** PostgreSQL

Aunque no se sigue un patrón MVC formal, la lógica se encuentra separada en archivos organizados según su funcionalidad.

**4. Estructura de archivos del proyecto**

PROYECTO-CSOFT/

│

├── conexion.php

├── login.php

├── menu.php

├── registro\_cliente.php

├── registro\_reparacion.php

├── listar\_reparaciones.php

├── css/

│ └── estilos.css

└── README.md

## ****5. Diseño de la base de datos****

**Base de datos:** cso

### Tabla clientes

| **Campo** | **Tipo** | **Descripción** |
| --- | --- | --- |
| id\_cliente | SERIAL (PK) | Identificador único |
| cedula | VARCHAR(10) | Cédula del cliente |
| nombre | VARCHAR | Nombre completo |
| telefono | VARCHAR | Teléfono de contacto |
| correo | VARCHAR | Correo electrónico |

### Tabla reparaciones

| **Campo** | **Tipo** | **Descripción** |
| --- | --- | --- |
| id\_reparacion | SERIAL (PK) | ID único de la reparación |
| id\_cliente | INT (FK) | Relación con el cliente |
| descripcion | TEXT | Descripción del trabajo a realizar |
| costo | NUMERIC | Costo estimado |
| estado | VARCHAR | Estado actual (En proceso, Completada) |
| fecha\_reparacion | TIMESTAMP | Fecha de registro de la reparación |

### Tabla usuarios

| **Campo** | **Tipo** | **Descripción** |
| --- | --- | --- |
| id\_usuario | SERIAL (PK) | ID de usuario |
| usuario | VARCHAR | Nombre de usuario (ej. admin) |
| clave | VARCHAR | Contraseña (encriptada con crypt()) |

## ****6. Lógica de funcionamiento por módulo****

### a) Login

* Verifica que el usuario y contraseña coincidan con los registrados en la base de datos.
* Usa sesiones PHP para mantener el acceso autorizado.
* Redirige a dashboard o menú si el login es correcto.

### b) Registro de clientes

* Permite registrar nuevos clientes con validaciones de cédula, correo y teléfono.
* Verifica que la cédula no esté duplicada.
* Muestra mensajes de confirmación o error según el resultado.

### c) Registro de reparaciones

* Permite seleccionar un cliente existente.
* Registra la reparación con su costo, descripción y estado inicial “En proceso”.
* Maneja errores en la entrada de datos.

### d) Listado de reparaciones

* Muestra todas las reparaciones registradas, con la opción de filtrarlas por estado.
* Permite al usuario ver fácilmente el historial y progreso de cada reparación.

## ****7. Tolerancia a fallos implementada****

El sistema incluye mecanismos básicos de tolerancia a fallos:

* Uso de try-catch con PDO para capturar errores en consultas SQL.
* Validación de datos antes de insertar en la base.
* Mensajes de error personalizados para evitar mostrar información técnica al usuario.
* Separación de lógica de negocio y presentación en archivos distintos.
* Registro de errores en el log del servidor (error\_log()), sin interrumpir la experiencia del usuario.

## ****8. Conclusiones****

El sistema de gestión de reparaciones cumple con los requerimientos básicos para el registro, consulta y control de clientes y trabajos técnicos. La estructura modular facilita su mantenimiento y posibles mejoras futuras.  
Gracias a la implementación de validaciones y manejo de errores, el sistema ofrece una experiencia estable para el usuario final.