**Desarrollo e Implementación de un Sistema de Servicio Técnico para Celulares Basado en Arquitectura en Capas**

L. VELASQUEZ, M. NUÑEZ, J. ESPINALES, O. YAGUAL, E. ROBLES, G. QUILLAMBAQUI

1Facultad de Ciencias Matemáticas y Físicas, Universidad de Guayaquil, Guayaquil, Ecuador

Emails: [leslyvelasquezanc@ug.edu.ec](mailto:leslyvelasquezanc@ug.edu.ec), [jeniffer.espinaleslin@ug.edu.ec](mailto:je%6e%69f%66%65%72.%65%73%70i%6e%61le%73%6ci%6e@%75g%2e%65%64u.%65%63), [miguel.nunezfue@ug.edu.ec](mailto:miguel.nunezfue@ug.edu.ec), [emily.roblesalv@ug.edu.ec](mailto:e%6dil%79%2er%6f%62%6c%65%73%61%6c%76@%75%67%2e%65%64%75.ec), [[guillermo.quilambaquimur@ug.edu.ec](mailto:g%75%69%6c%6ce%72%6d%6f.%71%75%69%6c%61%6d%62aqu%69%6d%75r@%75%67%2e%65%64u%2e%65%63)c](mailto:%6a%6fs%68%75%61%2ev%69%6cl%61%63%72es%62%6f%74@u%67.%65%64%75%2e%65c), [omar.yagualzam@ug.edu.ec](mailto:%6fmar.%79ag%75%61%6c%7aa%6d@%75%67%2e%65%64%75.ec)

RESUMEN Este artículo describe el desarrollo de un sistema de gestión de servicio técnico para celulares, diseñado para talleres de reparación. Utilizando una arquitectura en capas, el sistema se divide en: capa de presentación, capa de negocio y capa de datos, garantizando modularidad y mantenibilidad. La base de datos fue desarrollada en SQL Server, mientras que la aplicación se implementó en C# utilizando Visual Studio. Los principales retos incluyeron la integración entre capas y la gestión eficiente de datos. Se realizaron pruebas unitarias e integrales para asegurar la calidad del sistema, proporcionando una solución robusta y eficiente que optimiza procesos como la gestión de clientes, dispositivos, órdenes de reparación, repuestos y servicios.

PALABRAS CLAVES Arquitectura en Capas, Desarrollo de Software, Gestión de Reparaciones, SQL Server, Visual Studio, C#.

I. INTRODUCCIÓN

En la era digital, los talleres de reparación de dispositivos móviles enfrentan el reto de gestionar eficientemente las órdenes de reparación, inventarios y datos de clientes. Este proyecto aborda dicha necesidad mediante un sistema que automatiza y organiza estos procesos.

Nuestro objetivo principal es proporcionar una herramienta intuitiva y escalable que simplifique las operaciones diarias de los talleres, mejorando su eficiencia y reduciendo errores. Este artículo detalla la arquitectura, desarrollo e implementación del sistema, así como los resultados obtenidos.

II. TRABAJOS RELACIONADOS

Existen diversos estudios que abordan el desarrollo de sistemas de gestión utilizando arquitecturas en capas. Se describe un sistema para la gestión de reparaciones utilizando MVC, destacando la importancia de separar la lógica de negocio de la interfaz de usuario. Por otro lado, se presenta un enfoque basado en sistemas web para mejorar la accesibilidad.

Nuestro sistema opta por una aplicación de escritorio, priorizando la seguridad y el control local de los datos.

III. DESCRIPCIÓN DEL SISTEMA

#### **A. Arquitectura del Sistema**

El sistema está dividido en tres capas:

1. **Capa de Presentación:**
   * Diseñada en C# con Windows Forms.
   * Proporciona una interfaz intuitiva para gestionar clientes, dispositivos y órdenes.
2. **Capa de Negocio:**
   * Implementa la lógica del sistema, incluyendo validaciones y reglas de negocio.
   * Gestiona operaciones como el registro de órdenes y cálculo de costos.
3. **Capa de Datos:**
   * Basada en SQL Server.
   * Incluye tablas, vistas y procedimientos almacenados para manejar las operaciones CRUD.

#### **B. Tecnologías Utilizadas**

1. **Lenguaje de Programación:** C#
2. **Base de Datos:** SQL Server 2019
3. **IDE:** Visual Studio 2019
4. **Control de Versiones:** GitHub

IV. DESARROLLO

#### **A. Fase de Diseño**

Se realizaron diagramas de entidad-relación y prototipos de interfaz para definir la estructura del sistema. Este paso aseguró la correcta implementación de las relaciones entre las tablas y una experiencia de usuario clara.

#### **B. Implementación por Capas**

**Capa de Presentación:**

* + Incluye formularios para registrar clientes, dispositivos y órdenes de reparación.
  + Se aplicó un diseño responsivo para facilitar su uso.

**Capa de Negocio:**

* + Implementa validaciones como la consistencia de datos.
  + Maneja la autorización de órdenes y asignación de técnicos.

**Capa de Datos:**

* + Utiliza procedimientos almacenados para optimizar consultas y mejorar la seguridad.

#### **C. Pruebas**

Se realizaron pruebas unitarias con NUnit y pruebas de integración para garantizar la estabilidad del sistema. Las pruebas verificaron la interacción entre capas y la correcta ejecución de las operaciones.

V. RESULTADOS Y DISCUSIÓN

El sistema ha demostrado ser eficiente en la gestión de tareas críticas como:

1. Registro y consulta rápida de clientes y dispositivos.
2. Gestión de inventarios de repuestos.
3. Generación de reportes financieros y de operación.

#### **A. Beneficios**

1. Reducción del tiempo de procesamiento de órdenes.
2. Mejora en la organización de la información.
3. Incremento en la satisfacción del cliente.

#### **B. Retos Superados**

**Integración entre capas:** Solucionado mediante el uso de interfaces claras.

**Optimización de consultas:** Implementando índices en las tablas más consultadas.

VI. DESAFÍOS Y SOLUCIONES

El desarrollo del sistema de servicio técnico para celulares representa un avance significativo en la gestión de talleres de reparación. Su arquitectura en capas garantiza flexibilidad y escalabilidad, mientras que el enfoque en la experiencia del usuario asegura su adopción efectiva.

Las lecciones aprendidas durante este proyecto refuerzan la importancia de una planificación adecuada, la colaboración en equipo y las pruebas exhaustivas para el éxito de proyectos similares.

VII. LECCIONES APRENDIDAS

Durante la implementación del sistema, identificamos varias lecciones clave que serán valiosas para futuros proyectos:

**- Planificación es Fundamental:** Un diseño detallado inicial ayuda a anticipar problemas y evitar retrasos durante el desarrollo.

**-Colaboración en Equipo:** La comunicación constante entre los miembros del equipo fue esencial para garantizar la integración efectiva de las capas y componentes del sistema.

**-Pruebas Exhaustivas:** Incorporar pruebas unitarias y de integración desde las primeras fases asegura un producto final más estable y confiable.

**-Flexibilidad Ante Cambios:** Ser capaces de adaptarnos a nuevos requerimientos o solucionar problemas imprevistos fue un factor decisivo para cumplir los objetivos.

**-Documentación Continua:** Mantener actualizada la documentación técnica y de usuario facilita el mantenimiento y la transferencia de conocimientos.

VIII. CONCLUSIÓN

El desarrollo del sistema de servicio técnico para celulares representa un avance significativo en la gestión de talleres de reparación. Su arquitectura en capas garantiza flexibilidad y escalabilidad, mientras que el enfoque en la experiencia del usuario asegura su adopción efectiva.

Las lecciones aprendidas durante este proyecto refuerzan la importancia de una planificación adecuada, la colaboración en equipo y las pruebas exhaustivas para el éxito de proyectos similares.

IX. AGRADECIMIENTOS

Queremos expresar nuestro agradecimiento a nuestro profesor Parrales Franklin por sus enseñanzas para empezar y terminar con éxito el desarrollo del proyecto.

Agradecemos también a nuestros compañeros de equipo por su tiempo y colaboración durante todo el proyecto. Sus trabajos fueron válidos para la construcción del sistema, y el ambiente de cooperación y apoyo mutuo apoyó a la finalización del proyecto.

X. REFERENCIAS

1. [ProyectoMobile/ at main · Mikesaika/ProyectoMobile](https://github.com/Mikesaika/ProyectoMobile/tree/main)[Documentación oficial de C#](https://docs.microsoft.com/en-us/dotnet/csharp/)

1. [Documentación oficial de SQL Server](https://docs.microsoft.com/en-us/sql/sql-server/sql-server-2022" \t "_new)

1. [Documentación oficial de Visual Studio](https://docs.microsoft.com/en-us/visualstudio/" \t "_new)

1. [Documentación oficial de GitHub](https://docs.github.com/en" \t "_new)