# DOCUMENTO DEL ARTICULO DE INVESTIGACION DEL SISTEMA DE REGISTRO DE MANTENIMIENTO

# DE CELULARES

**Autores:**

Carrión Loaiza Marc Anthony

Carranza Freijo Bryan Alexander

Chiriguaya Vásquez Héctor Jeremias

Hidalgo Moran Ronny Alexander

Morales Espinales Terry Alejandro

Tabla de contenido

[DOCUMENTO DEL ARTICULO DE INVESTIGACION DEL SISTEMA DE REGISTRO DE MANTENIMIENTO 1](#_Toc202120547)

[DE CELULARES 1](#_Toc202120548)

[Resumen 3](#_Toc202120549)

[Abstract 4](#_Toc202120550)

[Introducción 5](#_Toc202120551)

[Objetivo General 6](#_Toc202120552)

[Objetivos Específicos: 6](#_Toc202120553)

[Metodología 7](#_Toc202120554)

[Metodología ágil SCRUM 7](#_Toc202120555)

[Visión del proyecto 7](#_Toc202120556)

[Planificación del proyecto 7](#_Toc202120557)

[Equipo SCRUM: 7](#_Toc202120558)

[Sprint 0: Preparación 8](#_Toc202120559)

[Sprint 1: Desarrollo Inicial 8](#_Toc202120560)

[Sprint 2: Entrega del Prototipo Funcional 8](#_Toc202120561)

[Sprint 3: Entrega Final del Sistema 9](#_Toc202120562)

[Materiales y Métodos 10](#_Toc202120563)

[Arquitectura del Sistema 10](#_Toc202120564)

[Componentes del sistema 10](#_Toc202120565)

[1. Recursos Estáticos (app/View) 10](#_Toc202120566)

[2. Vista (VIEW) 10](#_Toc202120567)

[3. Controladores (Controllers) 10](#_Toc202120568)

[4. Modelos (Models) 11](#_Toc202120569)

[5. Configuración (Config) 11](#_Toc202120570)

[Herramientas y Tecnologías Utilizadas 11](#_Toc202120571)

[Resultados 12](#_Toc202120572)

[Conclusiones 12](#_Toc202120573)

[Anexos 13](#_Toc202120574)

# Resumen

El presente artículo describe el desarrollo e implementación de un sistema informático de registro de mantenimiento de dispositivos móviles, orientado a optimizar la gestión de reparaciones, el control de datos de clientes y técnicos, y el seguimiento del historial de servicios en talleres técnicos. A través de un enfoque ágil basado en la metodología SCRUM, se identificaron las necesidades funcionales del entorno de trabajo y se diseñó una plataforma digital eficiente, intuitiva y adaptable. El sistema permite registrar dispositivos, visualizar historiales de mantenimiento, administrar la información de clientes y técnicos, y automatizar procesos que anteriormente se realizaban de forma manual. Como resultado, se logró mejorar significativamente la organización interna del taller, reducir errores administrativos y aumentar la satisfacción de los usuarios.

**Palabras clave:** sistema de registro, mantenimiento de celulares, gestión de reparaciones, datos de clientes, SCRUM.

# Abstract

This article presents the development and implementation of an information system for recording mobile device maintenance, aimed at optimizing repair management, customer and technician data control, and service history tracking in technical workshops. Using an agile approach based on the SCRUM methodology, the project identified functional needs and delivered an efficient, user-friendly, and adaptable digital platform. The system allows users to register devices, view maintenance histories, manage client and technician information, and automate processes previously done manually. As a result, the internal organization of the workshop was significantly improved, administrative errors were reduced, and user satisfaction increased.

**Keywords:** registration system, mobile maintenance, repair management, customer data, SCRUM.

# Introducción

En la actualidad, el mantenimiento y reparación de dispositivos móviles se ha convertido en un servicio esencial, impulsado por el constante crecimiento del mercado de teléfonos celulares y la dependencia que las personas tienen de estos dispositivos para actividades personales, laborales y comerciales. La alta demanda de servicios técnicos especializados ha generado la aparición de talleres de reparación, muchos de los cuales operan sin un sistema adecuado de gestión de la información.

Durante el desarrollo de este proyecto, se identificó un taller técnico que presenta serias limitaciones en la administración de los datos relacionados con los dispositivos ingresados para mantenimiento, los clientes que solicitan el servicio y los técnicos encargados de realizar las reparaciones. Estas deficiencias no solo afectan la eficiencia operativa del negocio, sino también la satisfacción del cliente y el ambiente laboral del personal.

Frente a esta problemática, se plantea la necesidad de diseñar e implementar un sistema de registro de mantenimiento de celulares que permita organizar y automatizar el flujo de información del taller. El sistema propuesto busca optimizar el control de ingresos y egresos de equipos, facilitar el seguimiento del estado de las reparaciones, gestionar adecuadamente los datos de los clientes y asignar las tareas a los técnicos de manera más eficiente. De esta manera, se espera mejorar la trazabilidad de los procesos, reducir errores administrativos y ofrecer un servicio más transparente y confiable a los usuarios.

# Objetivo General

Desarrollar un sistema informático para el registro y gestión del mantenimiento de dispositivos móviles, que optimice la administración de reparaciones, la organización de datos de clientes y técnicos, y el control eficiente de la información en talleres de servicio técnico.

# Objetivos Específicos:

1. Diseñar la estructura funcional del sistema, incorporando los módulos necesarios para el registro de equipos, el historial de reparaciones, la gestión de clientes y la asignación de técnicos.
2. Implementar una plataforma digital eficiente y de fácil uso que permita registrar, consultar y actualizar los datos asociados al mantenimiento de dispositivos móviles.
3. Evaluar el desempeño del sistema mediante pruebas funcionales que permitan validar su efectividad en la optimización de la gestión interna y la mejora de la satisfacción del cliente.

# Metodología

## Metodología ágil SCRUM

Para el desarrollo del sistema de registro de mantenimiento de dispositivos móviles, se utilizó la metodología ágil SCRUM, debido a su enfoque iterativo, incremental y centrado en el cliente, que permite entregar valor de forma continua y adaptarse a cambios durante el proceso.

### Visión del proyecto

El objetivo general del proyecto es desarrollar un sistema de registro de mantenimiento que permita gestionar de forma ágil y eficiente las reparaciones realizadas, los datos de los clientes, los técnicos asignados y el historial de servicios. Con ello se busca mejorar la organización interna del taller y optimizar la experiencia de usuario.

### Planificación del proyecto

Se elaboró un backlog con las funcionalidades principales del sistema, incluyendo:

* Registro de nuevos dispositivos.
* Visualización del historial de reparaciones.
* Gestión de técnicos asignados.
* Registro de clientes.

Este backlog es dinámico y se prioriza en cada sprint de acuerdo con el valor aportado y la retroalimentación de los interesados.

**Identificación de Stakeholders**

Se identificaron como partes interesadas al personal técnico, al equipo administrativo y a los usuarios finales. Su retroalimentación fue esencial para la validación de requisitos y mejora continua del sistema.

### Equipo SCRUM:

El equipo fue organizado con los siguientes roles:

* **Product Owner:** Representante de los usuarios, responsable de priorizar los requerimientos y validar las funcionalidades entregadas.
* **Scrum Master:** Encargado de garantizar el cumplimiento de los principios de SCRUM y eliminar obstáculos que afecten al equipo.
* **Equipo de Desarrollo:** Responsable del diseño, codificación, pruebas y documentación del sistema.

## Sprint 0: Preparación

El Sprint 0 tuvo como propósito preparar el entorno de desarrollo y definir la base técnica del proyecto. Las actividades incluyeron:

* **Selección de tecnologías:** C#, SQL Server y Visual Studio.
* Definición de la arquitectura del sistema y patrones de diseño.
* Redacción inicial de historias de usuario, como el registro de una reparación o la asignación de un técnico.
* Elaboración de un cronograma estimado y plan de trabajo.

## Sprint 1: Desarrollo Inicial

Durante el primer sprint se trabajó en los módulos más críticos del sistema. Las actividades realizadas fueron:

* **Reuniones diarias (Daily Scrum):** Breves encuentros para compartir avances, identificar obstáculos y planificar el día.
* **Desarrollo priorizado:** Se implementaron funcionalidades esenciales como el registro de dispositivos y de técnicos.
* **Revisión del Sprint:** Al finalizar el sprint, se presentó el avance al Product Owner para recibir retroalimentación inmediata.
* **Retrospectiva del Sprint:** El equipo reflexionó sobre el desempeño, identificó oportunidades de mejora y ajustó su estrategia para el siguiente sprint.

## Sprint 2: Entrega del Prototipo Funcional

En esta etapa se entregó un prototipo con todas las funcionalidades principales activas, incluyendo el registro de dispositivos, historial de reparaciones, gestión de técnicos, y administración de clientes. El objetivo fue validar el sistema con usuarios reales y recopilar información para posibles mejoras.

**Actividades realizadas:**

* **Despliegue del prototipo:** Se entregó una versión funcional del sistema en un entorno de pruebas, accesible para los usuarios finales.
* **Pruebas por usuarios:** Técnicos y personal administrativo utilizaron el sistema para realizar registros reales y comprobar el flujo de trabajo.
* **Recopilación de retroalimentación:** A través de encuestas y entrevistas breves, se identificaron aspectos positivos del sistema y sugerencias para mejorar la usabilidad, la velocidad de respuesta y la visualización de datos.
* **Análisis:** El equipo de desarrollo revisó los comentarios recibidos y priorizó ajustes para incluirlos en el siguiente sprint.

Esta fase fue esencial para garantizar que el sistema respondiera adecuadamente a las necesidades reales de los usuarios antes de su implementación final.

## Sprint 3: Entrega Final del Sistema

El Sprint 3 tuvo como objetivo realizar los últimos ajustes, implementar las mejoras obtenidas de la retroalimentación del Sprint anterior y entregar la versión final del sistema, lista para su uso.

* **Ajustes finales:** Se corrigieron observaciones relacionadas con la navegación, diseño de interfaz y flujos de validación.
* **Pruebas de aceptación:** Se ejecutaron pruebas de funcionamiento completas para verificar la estabilidad del sistema y su cumplimiento con los requisitos definidos.
* **Documentación técnica y manual de usuario:** Se elaboró la documentación necesaria para facilitar el mantenimiento del sistema y la capacitación del personal.
* **Entrega formal del sistema:** Se presentó la versión final al Product Owner y a los stakeholders, quedando lista para su implementación en el taller técnico.

Con este sprint se culmina el ciclo de desarrollo ágil, logrando un producto funcional, probado y alineado con las necesidades prácticas del entorno en el que será utilizado.

# Materiales y Métodos

En esta sección se ofrece una visión general de la estructura y el diseño del sistema desarrollado por los estudiantes a cargo del proyecto, la estructura se puede encontrar libremente en el repositorio de GitHub: <https://github.com/BrianAbH/WebRegistroGrupoE/tree/main>.

# Arquitectura del Sistema

El sistema fue desarrollado con el patrón de arquitectura Modelo-Vista-Controlador (MVC), permitiendo una separación lógica de responsabilidades entre la capa de presentación (View), la lógica de negocio (Controller) y el acceso a datos (Model). Esta estructura facilita la escalabilidad, el mantenimiento del código y la reutilización de componentes.

Cada componente se describe en las líneas siguientes, agrupándolo por paquetes:

# Componentes del sistema

## Recursos Estáticos (app/View)

* Contiene carpetas para CSS, JavaScript (js), imágenes (img), y archivos de Bootstrap para el diseño visual del sistema.
* Incluye una carpeta helpers para funciones reutilizables del front-end.

## Vista (VIEW)

Conformada por archivos .php que representan las interfaces del sistema:

* **vistaCliente.php:** Gestión de clientes.
* **vistaTecnico.php:** Interacción con los datos de los técnicos.
* **vistaDispositivo.php:** Información de los dispositivos móviles.
* **vistaReparacion.php:** Historial de reparaciones.

## Controladores (Controllers)

Son responsables de manejar la lógica entre el usuario y los modelos:

* **ClienteController.php:** Administra acciones relacionadas con los clientes.
* **TecnicoController.php:** Administra acciones de técnicos.
* **DispositivoController.php:** Procesa los datos de dispositivos móviles.
* **ReparacionController.php:** Gestiona las operaciones sobre el historial de reparaciones.

## Modelos (Models)

Manejan el acceso a datos y la lógica de base de datos:

* **ClienteModels.php:** Acceso a datos de clientes.
* **TecnicoModels.php:** Manejo de datos de técnicos.
* **DispositvoModels.php:** Operaciones sobre los dispositivos móviles.
* **ReparacionModels.php:** Registros de reparaciones realizadas.

## Configuración (Config)

Contiene database.php, encargado de establecer la conexión con la base de datos. Este archivo es esencial para vincular los modelos con el sistema de almacenamiento de datos.

# Herramientas y Tecnologías Utilizadas

* **Lenguaje de programación:** PHP
* **Gestión de base de datos:** MySQL, administrado a través de phpMyAdmin.
* **Entorno de desarrollo:** Visual Studio Code
* **Framework de diseño:** Bootstrap para el diseño responsivo de las interfaces.
* **Servidor local:** XAMPP, utilizado para simular un servidor Apache y ejecutar el sistema localmente.

# Resultados

Como resultado del desarrollo del sistema de registro de mantenimiento, se logró implementar una plataforma funcional con las siguientes características principales:

* Registro y visualización de dispositivos ingresados para reparación.
* Administración de información de clientes y técnicos.
* Consulta del historial de reparaciones por dispositivo o por cliente.
* Interfaz intuitiva con diseño responsivo, desarrollada con Bootstrap.

Durante la prueba con usuarios (realizada en el Sprint 2), se observó una mejora en la organización del taller técnico, con una reducción en el tiempo destinado a la búsqueda manual de información. Los usuarios destacaron la facilidad de uso del sistema y sugirieron pequeñas mejoras, como reorganización de menús y alertas visuales, las cuales fueron consideradas para la versión final.

# Conclusiones

La implementación del sistema de registro de mantenimiento de celulares permitió atender de manera efectiva las necesidades de gestión identificadas en el taller técnico. Se logró automatizar procesos clave, reducir errores en el manejo de la información y mejorar la trazabilidad de las reparaciones.

El uso del enfoque ágil SCRUM fue fundamental para adaptar el desarrollo a los requerimientos reales de los usuarios, permitiendo incorporar retroalimentación continua y mejorar la calidad del producto final.

Se concluye que el sistema desarrollado representa una herramienta eficiente para pequeños y medianos negocios del sector técnico, con potencial de escalabilidad e integración con otras plataformas, como sistemas de facturación o control de inventarios.

# Anexos

Diagrama

El contenido generado por IA puede ser incorrecto.Para más detalles sobre cómo utilizar el sistema, puede consultar el manual de usuario que se encuentra en su repositorio de GitHub <https://github.com/BrianAbH/WebRegistroGrupoE/tree/main>.

1 Presentación de los componentes del sistema desarrollado.

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

2 Interfaz de inicio de sesión de usuario

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

3 Interfaz de ingreso de datos de los clientes

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

4 Interfaz de usuario para verificar los datos del técnico

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

5 Interfaz de usuario para verificar los datos de los clientes

Imagen que contiene Diagrama

El contenido generado por IA puede ser incorrecto.

6 Interfaz de usuario para verificar los datos de los dispositivos móviles