

Software de Gestión de Proyectos de Vivienda Obras Vanguard

Housing Project Management Software Obras Vanguard

Javier Antonio Ballesteros Hoyos, Esteban David Lozano Rodriguez, Mayron Santiago Serrano Hernandez

Artículo recibido en noviembre de 2024; artículo aceptado en noviembre de 2024.

Ballesteros Hoyos, J. A., Lozano Rodríguez, E. D. Serrano Hernandez, M. S. (2024). Proyecto integrador [Obras Vanguard]. Volumen 1, 1-10.

Resumen

Este artículo presenta el desarrollo de un proyecto integrador que consolida las competencias adquiridas en el programa de Ingeniería de Sistemas, enfocándose en la conceptualización, diseño, implementación y evaluación de un software innovador para la gestión de pagos en proyectos de vivienda. El sistema ofrece herramientas para la administración integral de torres, apartamentos, clientes y pagos, incorporando funcionalidades clave como la generación de reportes personalizados y la gestión de roles de usuario (administrador y asesor).

Durante el desarrollo del proyecto, se emplearon metodologías fundamentales de la ingeniería de software. En particular, se adoptó la metodología en cascada, conocida por su enfoque secuencial y lineal, la cual garantizó una planificación estructurada que permitió cumplir con los requisitos del cliente y las especificaciones definidas desde las fases iniciales. Este enfoque fue complementado con un diseño técnico robusto que incluye diagramas UML, modelado relacional de bases de datos y principios de programación SOLID, asegurando que el sistema sea escalable, mantenible y adaptable a futuras necesidades.

El impacto del proyecto es significativo en el ámbito de la construcción de viviendas, al optimizar procesos críticos como la administración de proyectos, la venta de propiedades y el registro de pagos. Gracias a su diseño intuitivo y su funcionalidad avanzada, el software permite a los usuarios, tanto administradores como asesores, gestionar información clave de manera ágil, segura y confiable, promoviendo la eficiencia operativa. Además, la generación automatizada de reportes proporciona datos precisos para la toma de decisiones estratégicas, mejorando la gestión administrativa en el sector de la vivienda.

Palabras clave: gestión de pago, desarrollo de software, SOLID, vivienda, base de datos.

Abstract

This article presents the development of an integrative project that consolidates the skills acquired in the Systems Engineering program, focusing on the conceptualization, design, implementation, and evaluation of innovative software for payment management in housing projects. The system offers tools for the comprehensive management of towers, apartments, clients, and payments, incorporating key features such as customized report generation and user role management (administrator and advisor).

During the project development, fundamental software engineering methodologies were employed. Specifically, the waterfall methodology was adopted, known for its sequential and linear approach, which ensured a structured planning process that met the client's requirements and the specifications defined from the initial stages. This methodology was complemented by a robust technical design that includes UML diagrams, relational database modeling, and SOLID programming principles, ensuring that the system is scalable, maintainable, and adaptable to future needs.

The project's impact is significant in the housing construction sector, as it optimizes critical processes such as project management, property sales, and payment registration. Thanks to its intuitive design and advanced functionality, the software allows users, both administrators and advisors, to manage key information efficiently, securely, and reliably, fostering operational efficiency. Moreover, the automated report generation feature provides accurate data for strategic decision-making, enhancing administrative management in the housing sector.

Keywords: Payment Management, software development, SOLID, dwelling, database

Introducción

El acceso a la vivienda representa un desafío prioritario para las constructoras, ya que se enfrentan a la necesidad de gestionar procesos complejos relacionados con la venta, el cobro y la administración de proyectos habitacionales. La falta de herramientas tecnológicas específicas y eficientes en este sector a menudo dificulta el manejo integral de clientes, pagos y propiedades, lo que puede derivar en errores administrativos, demoras en las operaciones y una limitada capacidad para evaluar el rendimiento del negocio.

En respuesta a esta problemática, este estudio plantea el diseño y desarrollo de un sistema integral para la gestión de proyectos de vivienda. Este software está diseñado para atender las necesidades particulares del sector constructor, destacándose por su enfoque en la eficiencia operativa, la seguridad de los datos y la generación de reportes estratégicos. A diferencia de otras herramientas existentes en el mercado, esta solución combina una integración robusta de funcionalidades administrativas con la capacidad de personalizar reportes que ofrecen información clave para la toma de decisiones estratégicas.

Los objetivos principales de este proyecto incluyen: (1) diseñar una plataforma tecnológica que permita gestionar de manera ágil y centralizada los datos relacionados con clientes, ventas y pagos, reduciendo tiempos de procesamiento y minimizando errores humanos; (2) implementar principios de programación SOLID para garantizar que el sistema sea escalable, mantenible y adaptable a futuros requerimientos; y (3) desarrollar un módulo avanzado de reportes automatizados que brinde a las constructoras herramientas para evaluar el rendimiento operativo, identificar áreas de mejora y planificar estrategias con base en datos confiables y actualizados.

Al ofrecer una solución que optimiza la gestión de los procesos internos, promueve la eficiencia administrativa y mejora la experiencia tanto de los usuarios internos como de los clientes finales, este sistema tiene el

potencial de transformar la manera en que las constructoras operan, permitiéndoles adaptarse a un entorno competitivo y en constante cambio.

Metodología

Tipo de estudio

Este proyecto se basa en un enfoque no experimental, lo que implica que no se manipularon variables de forma directa, sino que se analizaron las condiciones existentes en los procesos actuales de gestión de proyectos de vivienda para identificar oportunidades de mejora. El diseño es de tipo descriptivo, ya que busca detallar de manera precisa las características y necesidades del sector inmobiliario en relación con la gestión de ventas y pagos, así como los requerimientos específicos de la compañía.

La investigación sigue un enfoque cuantitativo, centrado en la recopilación y análisis de datos objetivos para fundamentar las decisiones de diseño y desarrollo del software. Este enfoque permitió evaluar métricas relacionadas con la eficiencia de los procesos actuales, los tiempos de ejecución de tareas administrativas, y las necesidades de los usuarios finales, como administradores, asesores y clientes.

El estudio está orientado hacia la creación de un producto funcional, lo que significa que el principal objetivo es diseñar e implementar un sistema que no solo cumpla con los requerimientos técnicos, sino que también sea útil y práctico en el contexto empresarial. Esto incluye características como la optimización de la gestión de datos, la generación automática de reportes, y la implementación de un sistema de roles y seguridad para garantizar el control de acceso.

Herramientas tecnológicas

Para garantizar un desarrollo eficiente y estructurado del proyecto, se emplearon diversas herramientas

tecnológicas que facilitaron la implementación, gestión y control de calidad del sistema. Estas herramientas incluyen:

- **Lenguaje de programación:** Se utilizó Java como lenguaje principal para la implementación del sistema debido a su robustez, portabilidad y amplio soporte para herramientas de desarrollo. Java permitió construir una arquitectura modular basada en principios de programación orientada a objetos, lo que favoreció la escalabilidad y el mantenimiento del sistema.
- **Gestión de bases de datos:** El sistema de gestión de bases de datos Oracle fue elegido por su capacidad para manejar grandes volúmenes de información de manera segura y eficiente. Oracle ofrece características avanzadas como, integridad referencial y soporte para consultas complejas, lo que permitió estructurar los datos de clientes, proyectos y pagos de forma óptima.
- **Versionamiento:** Para la gestión del código fuente, se utilizó GitHub como plataforma de control de versiones. Esto permitió el seguimiento de cambios, la colaboración en equipo y la recuperación de versiones previas en caso de ser necesario. GitHub también facilitó la integración de ramas para desarrollar nuevas funcionalidades y corregir errores sin afectar la estabilidad del proyecto principal.

Requerimientos/Requisitos

El sistema propuesto integra funcionalidades clave y mecanismos de control que garantizan una gestión eficiente de la información y un acceso seguro. Los principales requisitos se detallan a continuación:

Gestión de Proyectos: Registro, actualización y consulta de proyectos de vivienda.

Gestión de Torres y Apartamentos: Organización estructurada de torres y apartamentos asociados a cada proyecto, incluyendo detalles como tipo de unidad, área y valor.

Gestión de Clientes: Registro de clientes, actualización de datos y consulta de historiales de compra.

Gestión de Ventas y Pagos: Registro y seguimiento de ventas realizadas, incluyendo detalles de escrituras y matrículas, así como el registro de pagos efectuados y pendientes.

Roles de Usuario

Administrador: Gestión completa de proyectos de vivienda.

Administración de torres y apartamentos, incluyendo altas, modificaciones y bajas..

Asesor: Registro y gestión de ventas, incluyendo registro de clientes.

Registro de pagos realizados por los clientes y seguimiento de pagos pendientes.

Generación de Reportes

Pagos Realizados: Visualización detallada de los pagos completados por cliente, torre y proyecto.

Pagos Pendientes: Identificación de clientes con deudas pendientes, mostrando fechas límite y montos adeudados.

Reportes Específicos por Cliente: Generación de historiales detallados que incluyan compras realizadas, pagos efectuados y estado actual de cuentas.

Procedimiento

El desarrollo del proyecto se llevó a cabo el modelo de cascada, un enfoque estructurado que divide el proceso en fases secuenciales, donde cada etapa depende de la finalización exitosa de la anterior. Este método permitió garantizar un avance ordenado y lógico en el diseño y desarrollo del sistema. Las fases implementadas fueron las siguientes:

- **Análisis:** Durante esta fase, se realizó un levantamiento exhaustivo de los requisitos del sistema mediante entrevistas, análisis de documentación y reuniones con los principales interesados. Se identificaron las principales necesidades del sector constructor, como la gestión eficiente de proyectos, clientes y pagos, y la generación de reportes personalizados. Los requisitos se clasifican en funcionales y no funcionales, asegurando que el sistema cumpliera tanto con las expectativas operativas como con los estándares de rendimiento y usabilidad.
- **Diseño:** Una vez definidos los requisitos y requerimientos, se procedió al diseño del sistema. Se elaboraron diagramas UML, como el diagrama de clases, el diagrama de casos de uso y el diagrama de secuencia, para modelar las interacciones y estructuras principales del sistema. Además, se diseñaron los modelos entidad-relación y relacional para la base de datos, asegurando una estructura óptima que permitiera almacenar y gestionar los datos de

manera eficiente. Durante esta fase, también se definieron las interfaces gráficas del usuario, enfocándose en una experiencia intuitiva y funcional.

- **Desarrollo:** En esta etapa, se implementan las funcionalidades definidas en el diseño utilizando el lenguaje de programación Java, en conjunto con la herramienta de desarrollo NetBeans. La arquitectura del sistema se dividió en capas (igu, lógica y persistencia), lo que permitió una organización modular del código y facilitó el mantenimiento. Se implementaron módulos específicos para la gestión de proyectos, clientes, ventas y pagos, así como funcionalidades adicionales como la generación automática de reportes y la seguridad basada en roles.
- **Pruebas:** La fase de pruebas incluyó la verificación y validación del sistema para garantizar que cumpliera con los requisitos establecidos. Se realizaron pruebas unitarias para asegurar el correcto funcionamiento de cada componente del sistema, pruebas de integración para verificar la interacción entre módulos, y pruebas funcionales para confirmar que el sistema cumpliera con los objetivos del proyecto.
- **Implementación:** Finalmente, se procedió al despliegue del sistema en un entorno controlado para su evaluación por parte de los usuarios finales. Durante esta fase, se capacitó al personal de la constructora en el uso del software y se recopiló retroalimentación para posibles mejoras. El sistema fue configurado para operar con una base de datos centralizada, asegurando la integridad de los datos y su acceso en tiempo real.

El uso de este enfoque estructurado permitió cumplir ciertos objetivos planteados como era de esperarse y

así asegurar que el producto que se tienen finalmente pueda cumplir con las necesidades del cliente y una ayuda hacia la gestión manual, esto quiere decir que no solamente del cliente sino también del empleado con esto se puede decir que se tiene un programa donde se facilitara la comprensión de este y la mejora para la compañía.

Resultados

El sistema desarrollado cumplió con los requerimientos del cliente, logrando:

- El diseño del sistema se enfoca en ofrecer una interfaz amigable y fácil de usar, reduciendo la curva de aprendizaje para los usuarios. Esto

permite que los empleados, independientemente de su nivel técnico, puedan adaptarse rápidamente al software, optimizando la productividad desde las primeras etapas de implementación.

- Con un sistema de gestión de roles bien definido, se asegura que cada usuario tenga acceso únicamente a las funcionalidades y datos relevantes para su función. Esto protege la información sensible y reduce riesgos asociados a accesos no autorizados o uso indebido del sistema.
- El sistema incluye un módulo para la creación de reportes financieros detallados, como estados de cuenta, análisis de pagos pendientes y flujos de caja. Esto no solo minimiza los errores humanos, sino que también ahorra tiempo y permite a las constructoras tomar decisiones informadas de manera más ágil.
- El sistema centraliza y organiza toda la información relacionada con los clientes, proyectos y pagos, lo que permite un seguimiento eficiente de cada proceso. Esto no solo mejora la precisión de los datos, sino que también facilita la consulta y actualización de información en tiempo real, garantizando una administración más efectiva.

Referencias

BankTrack. (s. f.). Software de gestión de cobros. Recuperado de <https://banktrack.com/blog/software-gestion-cobros>

Appvizer. (s. f.). Procesamiento de pagos: todo lo que necesitas saber. Recuperado de <https://www.appvizer.es/contabilidad-finanzas/procesamiento-de-pagos>

IBM. (s. f.). Creating a Java project. Recuperado de <https://www.ibm.com/docs/es/integration-bus/10.0?topic=node-creating-java-project>

HubSpot. (s. f.). Cómo programar con Java. Recuperado de

<https://blog.hubspot.es/website/como-programar-con-java>

GanttPRO. (s. f.). ¿Qué es el método de cascada?
Recuperado de
<https://blog.ganttpro.com/es/metodologia-de-cascada/#:~:text=ágiles%20y%20cascada-,¿Qué%20es%20el%20método%20de%20cascada%3F,se%20haya%20completado%20la%20anterior.>

Apache NetBeans. (s. f.). Connecting to Oracle
database. Recuperado de
<https://netbeans.apache.org/tutorial/main/kb/docs/ide/oracle-db/>

SafetyCulture. (s. f.). Metodología en cascada: guía y
ejemplos. Recuperado de
<https://safetyculture.com/es/temas/metodologia-en-cascada/>