

**Departamento de Ciencias de la  
Computación(DCCO)**  
**Carrera de Ingeniería de Software**

**Curso de Fundamentos de la Ingeniería de  
Software**

**Trabajo**

Presentado por: Gongora Llumiquinga Lucas Nicolas,  
Manosalvas Clavijo Gabriel Alexander, Molina Quevedo  
Jairo Estiven, Velez Valencia Yandri Josue (Grupo 5)

Ciudad: Quito

Fecha: 19 de noviembre del 2024

|   |    |
|---|----|
| 1. Introducción.....                      | 5  |
| 2. Planteamiento del trabajo.....         | 5  |
| 3. Sistema de Objetivos.....              | 5  |
| 4. Alcance.....                           | 6  |
| 5. Marco Teórico.....                     | 6  |
| 6. Ideas a Defender.....                  | 7  |
| 7. Resultados Esperados.....              | 8  |
| 8. Viabilidad.....                        | 9  |
| 9. Planificación para el Cronograma:..... | 11 |
| 10. Bibliografía.....                     | 11 |

# 1. Introducción

El software tiene como objetivo el desarrollo de un sistema orientado a la gestión administrativa de los pagos de los residentes en el condominio “La Primavera”. La aplicación está dirigida exclusivamente al administrador del condominio José Leiton.

Actualmente, el administrador de condominio enfrenta el desafío de llevar un registro de las alcúotas de los residentes, además, debe realizar reportes que justifique o valide su labor como administrador, lo cual se realiza de manera manual o mediante sistemas poco integrados como Excel, lo que aumenta la probabilidad de errores y retrasa los procesos administrativos.

El sistema a desarrollar proporcionará una solución automatizada que permitirá al administrador registrar, visualizar y actualizar los pagos de manera rápida y precisa. Además, ofrecerá funciones como el seguimiento del estado de los pagos, la generación de reportes de pagos y la gestión de alertas para pagos pendientes. La aplicación será intuitiva, accesible desde computadoras y enfocada en facilitar las tareas del administrador sin requerir intervención directa de los residentes.

## 2. Planteamiento del trabajo

### 2.1 Formulación del problema

El administrador del condominio “La Primavera”, José Leiton, enfrenta dificultades en la gestión de los pagos de los residentes, debido a la falta de un sistema que automatice el registro, control y seguimiento de los pagos. Esto genera retrasos y errores en los registros, afectando tanto la organización interna del condominio como la transparencia con los residentes generando desconfianza. La ausencia de una herramienta específica incrementa la carga administrativa, lo que limita la eficiencia operativa y puede ocasionar problemas como confusiones de pagos.

### 2.2 Justificación

La implementación de un sistema para la gestión de pagos en el condominio “La Primavera” es fundamental para optimizar las acciones para el registro de pagos y mejorar la eficiencia operativa del administrador. Esto no solo reducirá la carga de trabajo del administrador, sino que también aumentará la confianza de los residentes con el administrador ya que el cálculo y registro de pagos será de forma automatizada y contribuirá a la correcta gestión del flujo de caja del condominio.

## 3. Sistema de Objetivos

### 3.1. Objetivo General

Desarrollar un software de administración de pagos de alcúotas del condominio

“La Primavera”, para facilitar el registro, seguimiento y actualización de los pagos de los residentes, utilizando el marco de trabajo ágil SCRUM.

### 3.2. Objetivos Específicos

- Realizar la matriz de historias de usuario y obtener los requisitos funcionales y no funcionales.
- Elaborar un plan de pruebas y reporte de errores.
- Generar reportes de pagos y el estado de pago de las alícuotas de los residentes.
- Implementar un sistema de alertas para notificar al administrador sobre pagos vencidos o pendientes.

## 4. Alcance

El proyecto se enfocará en el desarrollo de una aplicación de escritorio para el administrador del condominio, que permita gestionar exclusivamente los pagos de los residentes. La aplicación incluirá funcionalidades para el registro, seguimiento y control de pagos, la generación de reportes de pagos, y la gestión de alertas para pagos pendientes. No se contempla la integración directa con sistemas de pago externos ni la interacción de los residentes con la plataforma; su uso será exclusivo para el administrador. La aplicación estará disponible solo en laptop o pc.

## 5. Marco Teórico

### ¿Qué es y para qué sirve un IDE?

Un entorno de desarrollo integrado (IDE) es un sistema de software para el diseño de aplicaciones que combina herramientas del desarrollador comunes en una sola interfaz gráfica de usuario (GUI). Generalmente, un IDE cuenta con las siguientes características:

- Editor de código fuente
- Automatización de las compilaciones locales
- Depurador

Nuestro proyecto será realizado en Codeblocks y NetBeans.

### Code::Blocks

Code::Blocks es un IDE (Integrated Development Environment, o entorno de desarrollo integrado) para C, C++ y Fortran. Es un entorno totalmente configurable,

orientado a satisfacer las necesidades más exigentes de los usuarios. Construido sobre un marco para plugins, puede extenderse por medio de diferentes plugins, incorporando cualquier tipo de funcionalidad mediante la instalación o codificación de un plugin específico (por ejemplo, la funcionalidad de compilación y debugging).

## NetBeans IDE

NetBeans IDE es un entorno de desarrollo integrado, gratuito y de código abierto para el desarrollo de aplicaciones en los sistemas operativos Windows, Mac, Linux y Solaris.

El IDE simplifica el desarrollo de aplicaciones web, corporativas, de escritorio y móviles que utilizan plataformas Java y HTML5. El IDE también ofrece soporte para el desarrollo de aplicaciones PHP y C/C++.

### 5.1 Metodología (Marco de trabajo 5W+2H)

| Pregunta   | Descripción  |
|------------|--|
| ¿Qué?      | Crear un sistema para la administración financiera de un condominio  |
| ¿Para qué? | Para facilitar la administración financiera del condominio   |
| ¿Quién?    | Grupo 5 de la materia fundamento de ingeniería de software   |
| ¿Cuándo?   | En el periodo actual de clases   |
| ¿Dónde?    | El sistema será desarrollado y probado primero en clases, y posteriormente en la PC o laptop del administrador |
| ¿Cómo?     | Utilizando el marco de trabajo ágil SCRUM.   |
| ¿Cuánto?   | El tiempo que dure el presente periodo académico   |

Tabla 1. Metodología

## 6. Ideas a Defender

- **Reducción de carga administrativa:** Automatizar tareas repetitivas, reducir errores humanos y permitir que los administradores se concentren en tareas de valor añadido.

- **Automatización de pagos y finanzas:** Facilitar el registro de pagos, automatizar recordatorios de vencimientos y generar informes financieros claros y accesibles.
- **Optimización de tareas administrativas:** Gestionar incidencias, actas, reuniones y contratos de forma digital y automatizada.
- **Accesibilidad y facilidad de uso:** Interfaz intuitiva para el usuario.
- **Cumplimiento y seguridad:** Implementación de autenticación para evitar modificaciones no deseadas por terceros en el sistema.
- **Participación y transparencia:** Registro y acceso a historiales de decisiones, actas y pagos.
- **Escalabilidad y mantenimiento:** Adaptabilidad al incremento de residentes en la comunidad y facilidad de actualización sin complicaciones.
- **Soporte y formación:** Ofrecer soporte técnico y formación online para facilitar el uso y aprovechamiento del sistema.

## 7. Resultados Esperados

### **Eficiencia operativa:**

Se espera reducir el tiempo invertido en tareas administrativas repetitivas (como la gestión de pagos, la emisión de recordatorios, y la generación de informes financieros). Esto permitirá que el administrador dedique más tiempo a tareas de valor añadido, como la resolución de incidencias y la comunicación de los pagos a los residentes.

### **Reducción de errores humanos:**

Se anticipa una disminución en los errores relacionados con la facturación y los pagos, debido a la automatización del proceso de cobro, la generación automática de facturas.

### **Transparencia y participación:**

Se espera un incremento en la confiabilidad de los residentes en el administrador debido a los reportes que genere el sistema y permite crear pruebas que respalden el trabajo realizado por el administrador.

### **Aumento de la rentabilidad:**

La automatización de tareas administrativas reducirá los costos operativos ,

contribuyendo a una mejora en la rentabilidad del administrador y un ahorro en costos de gestión.

## 8. Viabilidad

| Recursos              | Descripción  | Costo estimado    |
|-----------------------|--|-------------------|
| Hardware              | Equipos de computo (PCs de desarrollo y servidores)                      | \$2.000,00        |
| Software              | Licencias de software (Office 365)                                       | \$1.000,00        |
| Recursos Humanos      | Sueldo para los desarrolladores  | \$4.000,00        |
| Capacitación          | Capacitación para el administrador del condominio en el uso del software | \$500,00          |
| Fondos imprevistos    | Fondo para imprevistos (10% del presupuesto total)                       | \$750,00          |
| <b>Total Estimado</b> |  | <b>\$8.250,00</b> |

Tabla 2. presupuesto del proyecto

### 8.1 Humana

#### 8.1.1 Tutor Empresarial

Ing. Jenny Ruiz

Supervisar la viabilidad comercial y las necesidades administrativas del proyecto, asegurando que el software desarrollado cumpla con los requisitos de los administradores de los condominios.

#### 8.1.2 Tutor Académico

Ing. Jenny Ruiz

Supervisar la viabilidad comercial y las necesidades administrativas del proyecto, asegurando que el software desarrollado cumpla con los requisitos de los administradores de los condominios.

### 8.1.3 Estudiantes

- Gongora Lucas
- Manosalvas Gabriel
- Molina Jairo
- Velez Yandry

## 8.2 Tecnológica

### 8.2.1 Hardware

- Se requerirán equipos de cómputo con capacidad para desarrollar y probar la aplicación. Los equipos de escritorio serán suficientes para realizar el desarrollo del software. Además, se necesita un servidor de bajo costo para pruebas y almacenamiento de la base de datos.

### 8.2.2 Software

- **Entorno de Desarrollo:** Se utilizarán herramientas como NetBeans o Code::Blocks.
- **Base de Datos:** La base de datos se gestionará con MySQL o PostgreSQL, ya que son herramientas de código abierto y con buena escalabilidad.
- **Lenguajes de Programación:** Se emplearán C# o Java para la construcción de la aplicación de escritorio, dependiendo de la disponibilidad de recursos y conocimientos del equipo.



## 9. Planificación para el Cronograma:

| Fase                             | Descripción   | Duración Estimada | Fecha de Inicio | Fecha de Fin |
|----------------------------------|---|-------------------|-----------------|--------------|
| 1. Análisis y Planificación      | Definición de requerimientos, análisis de necesidades y planificación del proyecto. | 1 semanas         | 18/11/2024      | 24/11/2024   |
| 2. Diseño de la Aplicación       | Diseño de la arquitectura del software, interfaces y base de datos.                 | 2 semanas         | 25/11/2024      | 08/12/2024   |
| 3. Desarrollo del Sistema        | Programación de la aplicación: backend, frontend, integración de la base de datos.  | 3 semanas         | 09/12/2024      | 29/12/2024   |
| 4. Pruebas Internas              | Pruebas de funcionalidad, detección de errores y corrección de fallos.              | 2 semanas         | 30/12/2024      | 12/01/2025   |
| 5. Capacitación al Administrador | Capacitación sobre el uso de la aplicación para los administradores del condominio. | 1 semana          | 13/01/2025      | 19/01/2025   |
| 6. Implementación y Entrega      | Implementación final en el entorno real y entrega del sistema al cliente.           | 1 semana          | 20/01/2025      | 26/01/2025   |
| 7. Seguimiento y Ajustes         | Ajustes finales tras feedback y posibles correcciones o mejoras.                    | 2 semanas         | 27/01/2025      | 09/02/2025   |

## 10. Bibliografía

Götz, T. (2019). *Integrated Development Environment (IDE)*. Webopedia.

Recuperado el 18 de noviembre de 2024, de

<https://www.webopedia.com/>

Code::Blocks. (n.d.). *Code::Blocks: A free C, C++ and Fortran IDE*. Recuperado

el 18 de noviembre de 2024, de <https://www.codeblocks.org/>

Apache NetBeans. (2024). *NetBeans IDE: Open-source integrated*

*development environment*. Apache Software Foundation. Recuperado el

18 de noviembre de 2024, de <https://netbeans.apache.org/>

Martínez, L. (2022). *El análisis estratégico con el modelo 5W+2H: ¿Qué es y*

*cómo aplicarlo?*. Estrategia y Gestión. Recuperado el 18 de noviembre

de 2024, de <https://www.estrategiaygestion.com>

## **Anexos.**

### **Anexo I. Crono**

| <b>Fase</b>                        | <b>Descripción</b>   |
|------------------------------------|--|
| <b>1. Análisis y Planificación</b> | <b>Definición de requerimientos, análisis de necesidades y planificación del proyecto.</b> |
| <b>2. Diseño de la Aplicación</b>  | <b>Diseño de la arquitectura del software, interfaces y base de datos.</b>                 |
| <b>3. Desarrollo del Sistema</b>   | <b>Programación de la aplicación: backend, fronted, integración de base de datos.</b>      |
| <b>4. Pruebas Internas</b>         | <b>Pruebas de funcionalidad, detección de errores y corrección de fallos.</b>              |

|   |  |
|---|--|
| <b>5. Capacitación al administrador</b> | <b>Capacitación sobre el uso de la aplicación para los administradores del condominio.</b> |
| <b>6. Implementación y Entrega</b>      | <b>Implementación final en el entorno real y entrega al sistema cliente.</b>               |
| <b>7. Seguimiento y Ajustes</b>         | <b>Ajustes finales tras feedback y posibles correcciones o mejoras.</b>                    |

## **Anexo II. Crono Matriz de identificación de requisitos**

| ITEM      | PROBLEMA  | QUE (NECESIDAD)   | PARA QUÉ (SOLUCIÓN)  | PARA QUIEN (USUARIO) | COMO (DESCRIPCIÓN DE TAREAS)   | HECHO POR (PROG. RESP.) | CUANTO TIEMPO (ESTIMADO EN HRS) | FECHA DE ENTREGA | PROBANDO | STATUS    | PRUEBA (COMO SE VERIFICA)                      | COMENTARIOS | NOMBRE DE HISTORIA      |
|-----------|---|---|--|----------------------|--|-------------------------|---------------------------------|------------------|----------|-----------|--|-------------|-------------------------|
| CT-RF-001 | El administrador tiene problemas al gestionar pagos de forma manual.          | Gestionar los pagos de forma eficiente.                         | Automatizar el registro y control de pagos                           | Administrador        | Desarrollar funciones para registrar y gestionar pagos en el sistema.          | Equipo 5                | 20                              |                  | Alta     | Pendiente | Pagos registrados correctamente.               | S/C         | Gestionar pagos         |
| CT-RF-002 | Falta de alertas automáticas para pagos vencidos.                             | Implementar alertas automáticas para pagos vencidos.            | Notificar al administrador sobre pagos pendientes.                   | Administrador        | Crear un sistema de notificaciones basado en fechas límite.                    | Equipo 5                | 15                              |                  | Alta     | Pendiente | Alertas enviadas para pagos vencidos.          | S/C         | Notificaciones de pagos |
| CT-RF-003 | Dificultad en generar reportes financieros claros y precisos.                 | Facilitar la generación de reportes de pagos.                   | Generar reportes detallados y accesibles para el residente           | Administrador        | Implementar herramientas para generar reportes personalizados.                 | Equipo 5                | 25                              |                  | Alta     | Pendiente | Reportes financieros generados sin errores.    | S/C         | Generación de reportes  |
| CT-RF-004 | No existe un sistema claro para calcular y aplicar multas por pagos atrasados | Establecer un sistema automático para calcular y aplicar multas | Automatizar el cálculo y la aplicación de multas por pagos atrasados | Administrador        | Desarrollar una función para que el sistema calcule y registre automáticamente | Equipo 5                | 18                              |                  | Alta     | Pendiente | Cálculo correcto de multas por pagos atrasados | S/C         | Cálculo de multas       |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|
|  |  |  |  | las multas por<br>pagos atrasados<br>según el<br>porcentaje<br>establecido |  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|

### Anexo III. Historia de Usuario

El cliente José Leiton, quien desempeña el rol de administrador del condominio, nos expresó durante una conversación que sus tareas podrían ser mucho más eficientes y sencillas si contara con una herramienta adecuada que optimiza sus funciones diarias.