

Capítulo 1

Laura Daniela Cárdenas Araya y Juan Sebastián Rojas Cachón

A) Resuma el ciclo de vida de Construcción de un Programa.

El ciclo de vida de un Programa es el proceso que sigue hace mucho porque se identifica la necesidad de crear un software hasta que se ejecuta y se mantiene funcionando y tiene varios ciclos y son:

* Identificación de un Problema

Un cliente o empresa detecta una necesidad o problema que se resuelve con un programa, y también se contacta a un equipo de desarrollo de software para encontrar una solución.

* Análisis y diseño

Se estudia el problema en detalle, identificando qué tiene que hacer el programa, y se definen los requisitos funcionales y no funcionales que es el rendimiento, compatibilidad, seguridad. Se diseña una solución, organizando cómo se estructurará el código y las tecnologías que se usarán.

* Desarrollo o implementación

Según lo leído se escribe el código del programa usando el lenguaje de programación y se crean funciones, interfaces y bases de datos dependiendo del diseño.

* Pruebas y validación

Se realizan diferentes pruebas para afirmar que el programa funciona a la perfección y cumple con los plazos del cliente y se corrigen errores y se mejoran el rendimiento.

* Implementación y despliegue

Se instala el programa en el lugar donde se usará, ejemplo en computadora, un servidor o una aplicación web, y se capacita a los usuarios sobre cómo utilizarlo.

* Mantenimiento y actualización

Se monitorea el uso del programa y se realizan mejoras o correcciones de errores al pasar el tiempo y se agregan nuevas funcionalidades necesarias del cliente.

B) Explique los aspectos que hacen parte del análisis de un problema

Esto sirve para poder desarrollar un excelente programa y tiene 3 aspectos y son:

* Requisitos del usuario

¿Cuáles son las funciones principales que necesita?

¿De qué quiere el cliente que haga el programa?

Si el cliente quiere un software para gestionar empleados se preguntará:

¿De qué necesita controlar horarios, pagos, asistencias?

* Contexto del problema

¿De qué condiciones externas afectan su funcionamiento?

¿De qué entorno funcionará el programa?

Si se crea un sistema para calcular salarios se tienen que conocer las leyes laborales, tipo de empresa, impuestos.

* Requisitos NO funcionales

Restricciones técnicas que afectan bastante al software y son:

Seguridad = ¿el programa necesita proteger datos?

Tiempo de respuesta = ¿debe ser rápido en procesar la información?

Compatibilidad = ¿debe funcionar en windows?

Es como si un banco necesitara que su software sea seguro y rápido para procesar transacciones sin errores.

C) Explique los pasos del proceso de solución de problemas

Según lo leído tiene un proceso Ordenado y son:

* El cliente identifica un problema

Una empresa o persona detecta un problema que necesita una solución por software y contacta a un programador o empresa de desarrollo de software.

* El programador analiza y diseña la solución

Se ejecuta o estudia el problema y se hace un plan para resolverlo y se sigue un proceso estructurado para garantizar que la solución funcione y sea clara.

* El programador implementa y prueba el programa

Se desarrolla el software y se realizan pruebas para asegurar que funcione bien, se ajustan errores para mejorar el rendimiento.

* El programador instala el programa para el usuario

Se instala el software en el sistema del cliente y se capacita al usuario para que lo pueda usar muy bien.

Si una empresa o persona necesita un sistema que controle los inventarios, el programador analiza qué información utilizará, diseña la base de datos, desarrolla el software y finalmente lo instala para que los empleados lo usen sin ningún inconveniente.

• ¿Cuáles son los elementos que se deben entregar a un cliente?

Los elementos son:

* El Programa funcional

El software debe estar listo para que el cliente lo use, puede ser en un computador, servidor o aplicación web.

* Documentación de usuario

Puede entregarse en un manual o una guía para saber como usar el programa y las instrucciones sobre las funciones y claramente la configuración del software.

* Documentación técnica

Se explica el código, la estructura o el diseño del programa que es útil para futuros desarrolladores ya necesiten hacer modificaciones.

* Pruebas y reportes de validación

mostrarán informes de pruebas realizadas para demostrar que el software funcione de forma perfecta y no fallar en ejecuciones, entonces tendría registros de errores corregidos y mejoras implementadas ahí.

* Capacitación o Soporte técnico

Lo mayoría de veces, se ofrece capacitación a los empleados del cliente sobre el uso del programa y se incluye el soporte técnico por si hay problemas.

Digamos que se crea un sistema de facturación para un negocio, y pasesse entregar el software instalado, se envía el manual a los empleados y las pruebas que demuestran que el funcionamiento es excelente para finalizar el documento con detalles técnicos.

e) Elabore la tarea N° 1 (Pag. 5 del texto guía), con el Objeto de identificar los aspectos que forma parte de un problema.

Objetivo identificar los aspectos

El Problema: un banco quiere crear un programa para manejar sus cajeros automáticos. Dicho programa debe permitir retirar el dinero y consultar el saldo de una cuenta. Identifique y discuta los aspectos que constituyen el problema. Si el enunciado no es explícito con respecto a algún punto, intente imaginar para comprender.

Aspecto	Descripción
diente	Un banco que quiere implementar un programa para manejar sus cajeros automáticos
Usuario	Son los clientes del banco que utilizan los cajeros automáticos para retirar dinero y consultar su saldo
Requerimiento funcional	El programa tiene que permitir a los usuarios retirar el dinero de su cuenta, además debe permitir que se pueda consultar el saldo de una cuenta.
Mundo del problema	Para que el sistema funcione bien tiene que estar conectado a la base de datos del banco. Ya que ahí se almacenarán los saldos de las cuentas y tiene que tener medidas de seguridad para evitar robos, como la identidad del usuario, tarjetas y claves y lo más importante, los cajeros deben tener dinero suficiente para que los clientes obtengan el dinero solicitado.
Requerimiento No funcional	Se trata de que el sistema garantice que solo las personas o los dueños de las cuentas accedan a su saldo y retirar dinero y que los cajeros estén activos las 24 hrs en diferentes ubicaciones y por último que las consultas y retirar se procesen rápidamente para evitar demoras y errores.

f) Elabore la tarea N°2 (Pág 13), con el objetivo de identificar los requerimientos funcionales de un problema.

Para el caso de estudio 2. Un simulador bancario, identifique y especifique 3 requerimientos funcionales

<u>Requerimiento funcional 1</u>	<u>Nombre</u>	Consulta de Saldo
	<u>Resumen</u>	Este sistema tendrá que permitir que los usuarios puedan verificar su saldo disponible en su cuenta de manera fácil y sencilla
	<u>Entradas</u>	El número de la cuenta que ya este autenticado
	<u>Resultado</u>	Se tiene que mostrar el saldo actual de la cuenta en pantalla
<u>Requerimiento funcional 2</u>	<u>Nombre</u>	Autenticación de usuario
	<u>Resumen</u>	El sistema tendrá que permitir que los usuarios se identifiquen ingresando su número de cuenta y contraseña para evitar robos.
	<u>Entradas</u>	El número de la cuenta y la contraseña.
	<u>Resultado</u>	Tiene acceso al sistema si los credenciales son correctos y si no es así, se mostrará un mensaje de error.
<u>Requerimiento funcional 3</u>	<u>Nombre</u>	Transferencia de fondos
	<u>Resumen</u>	Donde el sistema tendrá que permitir a los usuarios transferir dinero entre cuentas dentro del banco
	<u>Entradas</u>	Tendrá cuenta de origen, también cuenta de destino y por último monto a transferir
	<u>Resultado</u>	Si los datos son correctos y además hay saldo suficiente, se obtendrá la transferencia y se actualizarán los saldos de ambas cuentas y si no es así, se mostrará un mensaje de error.

g) Elabore la tarea N°3 (Pág.14), con el objetivo de identificar los requerimientos funcionales de un problema

Para el caso de estudio 3, un programa para manejar un triángulo, identifique 4 ejecutivos 3 requerimientos funcionales

Requerimiento funcional 1	Nombre	Saber que tipo de triángulo es
	Resumen	El programa debe identificar un triángulo como equilatero, isóceles o escaleno en función de la longitud de los lados
	Entradas	La longitud de los tres lados del triángulo
	Resultado	Se mostrará en la pantalla el tipo de triángulo que se obtuvo.

Requerimiento funcional 2	Nombre	Calcular ahora el perímetro
	Resumen	El Programa tendrá que calcular el Perímetro del triángulo y lo hará sumando la longitud de los 3 lados
	Entradas	La longitud de los tres lados del triángulo
	Resultado	Se mostrará en la pantalla el Perímetro calculado del triángulo

Requerimiento funcional 3	Nombre	Calcular ahora el área
	Resumen	El Programa tendrá ahora que calcular el área del triángulo y se logrará utilizando la fórmula de Herón
	Entradas	La longitud de los tres lados del triángulo
	Resultado	Se mostrará en la pantalla el área calculada del triángulo propuesto.

Grupo 1

Louis Daniela Cruz Amaya y Juan Sebastian Pineda C.

(H) Elabore la tarea No:4 (Pag 17), con el objetivo de identificar las entidades del mundo del problema.

Entidad	El programa debe incluir un método que permite dibujar y visualizar el triángulo en cuestión. No se especifica el método de ingreso del triángulo, lo más comodo sería ingresar las medidas del mismo.
Entidad	El programa debe incluir un método por el cual calcule el perímetro del triángulo ingresado/dibujado.
Entidad	El programa de incluir un método o proceso por el cual calcule el área del triángulo ingresado/dibujado.

Punto de reflexión: ¿Qué pasa si no identificamos bien las entidades del mundo?

R: Si no se identifican de forma adecuada las entidades del mundo, no se llegaría a un análisis extenso y comprensión precisa del problema en cuestión, esto afectaría en general a la solución a la que se pueda llegar, dando un resultado poco satisfactorio o que no cumple con todos los requerimientos que se especifican

Punto de reflexión: ¿Cómo decidir si se trata efectivamente de una entidad y no sólo de una característica de una entidad?

R: Una entidad es independiente, no dependiente, como por ejemplo un nombre de un cliente, tienen un atributo único: Una función, una representación numérica o un concepto (estando de cuenta, valor, ETC)

① Elabore la tarea, N°. 5 (pág. 20) con el objetivo de identificar las entidades de un caso de estudio.

Atributo	Valores Posibles	Diagrama UML
Nombre	cadena de caracteres	cuenta de ahorros Nombre
Apellido	cadena de caracteres	Apellido
Cédula	valores positivos	Cédula

Atributo	Valores Posibles	Diagrama UML
Depositar	Valores enteros positivos	Cuenta de ahorro corriente
Retirar	Valores enteros positivos	Depositor Apellido
Interés	Valor fijo en 0	Interés

Atributos	Valores Posibles	Diagramo UML
Depositar	Valores enteros positivos	cuenta de ahorro
Retirar	Valores enteros positivos	Depositor Neto
Interés	Aumento mensual del 0,6%	Interés

Atributo	Valores Posibles	Diccionario UML
Invertir	Valor entero positivo	COT
Interes	Valor entero positivo	Invertir Interes Cierre de cuenta
Cierre de cuenta	Si o No	
Atributo	Valores Posibles	Diccionario UML
Avanzar 1 mes	Si o No	Mes
Resultado de Movimientos	Valores enteros positivos	Avanza, 1 mes n. de Movimientos Res. de inversiones
Rendimientos de Inversiones	Valores enteros positivos	

⑤ Elabore la tarea No. 6 (pág. 23), con el objetivo de reflexionar sobre el nivel de precisión de un algoritmo

En general, es un algoritmo que le sirve útil a la mayoría de viajeros, pero no contempla casos específicos, como por ejemplo, una persona necesita ir de estación A a B, pero no hay linea directa entre A y B, por lo que el usuario tendrá que ir a una estación intermedia extra.

⑥ Estudie los siguientes aspectos del ejemplo seleccionado: Enunciado, Requerimientos funcionales (casos de uso) y el modelo (clases del proyecto). A continuación, redacte el enunciado del problema y el nombre de cada uno de los requerimientos funcionales del proyecto

Enunciado:

Del empleado se maneja la siguiente información:

- Nombre

- Apellido
- Género (femenino e masculino)
- Fecha de nacimiento
- Foto
- Fecha de ingreso a la empresa
- Salario básico

Si aplicación permite visualizar la información, y hacer los siguientes cálculos:

- Edad del empleado, utilizando la fecha de nacimiento
- Antigüedad del empleado, utilizando fecha de ingreso
- Prestaciones a las que tiene derecho el empleado. Para este cálculo se debe usar la siguiente fórmula:

$$\text{Prestaciones} = (\text{antigüedad} * \text{salario}) / 12$$

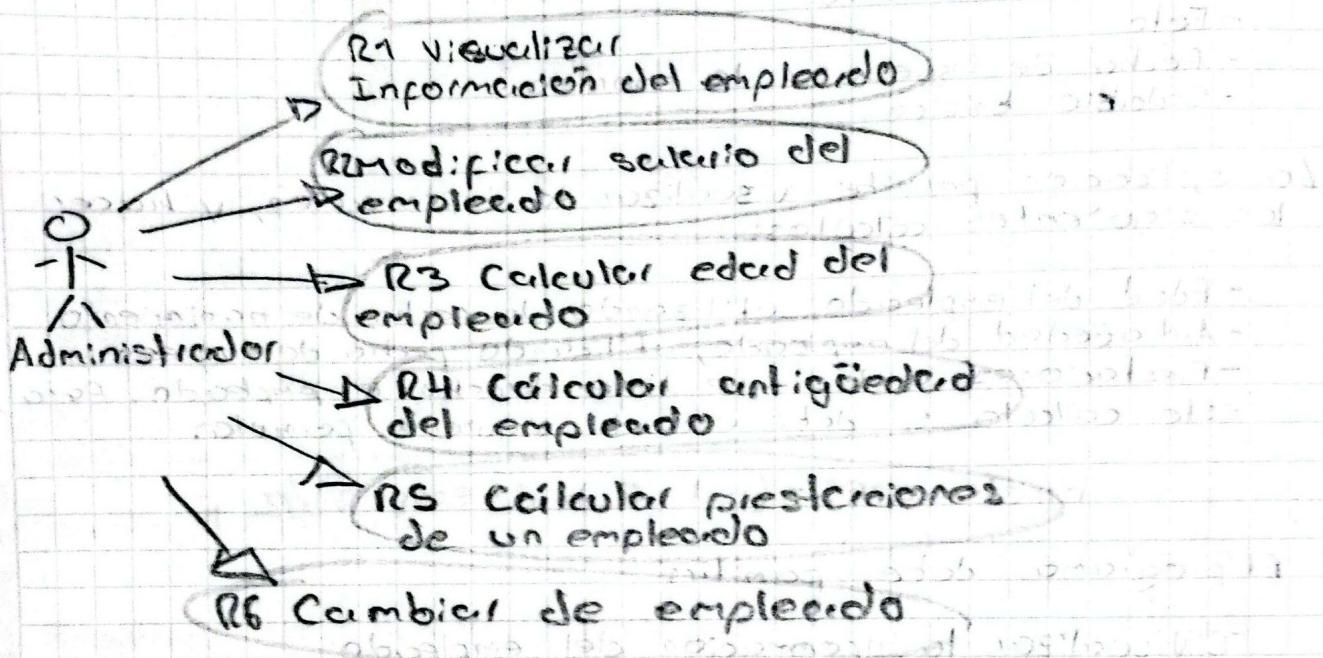
El programa debe permitir:

1. Visualizar la información del empleado
2. Modificar el salario del empleado
3. Calcular edad del empleado
4. Calcular la antigüedad del empleado en la empresa
5. Calcular prestaciones del empleado
6. Cambiar el empleado.

Requerimientos

- R1: "Visualizar la información del empleado"
- R2: "Modificar salario del empleado"
- R3: "Calcular la edad del empleado"
- R4: "Calcular la antigüedad del empleado"
- R5: "Calcular las prestaciones de servicio"
- R6: "Cambiar el empleado"

D) Dibuja el respectivo diagrama de casos de uso del ejemplo elegido



M) Observa nuevamente el modelo conceptual del caso y escribe el nombre de cada clase identificando variables y funciones:

Clase: Empleado

Variables:

- Nombre
- Genero
- Salario
- Apellido
- Imagen

Funciones:

- | | |
|---|--|
| <ul style="list-style-type: none">- Dar nombre- Dar genero- Dar fecha de Nac.- Dar fecha de Ingr.- Dar imagen- Cambiar empleado- Dar fecha actual | <ul style="list-style-type: none">- Dar salario- Dar apellido- Calcular edad- Calcular Antiguedad- Calcular Prestaciones- Cambiar Salario |
|---|--|

Clase: Fecha

VARIABLES:

- dia
- mes
- año (año)

Funciones:

- Fecha
- Dar Año(año)
- Dar dia
- Dar diferencia en meses
- Dar Mes

(N) Debes plantear 2 ideas de proyectos (problemas solubles y cígoritmicos)

Proyecto #1: "Gestión de inventario para una tienda de ropa"

R1: Nombre: Agregar nuevo producto

Resumen: Permite registrar nuevos productos en el inventario

Entrada: Nombre del producto, código, cantidad, proveedor

Salida: Se confirman los datos del producto y se confirma el registro

R2: Nombre: Actualizar stock

Resumen: Permite actualizar la cantidad de un producto en el inventario

Entrada: Código de producto, cantidad a agregar o quitar

Salida: Confirmación de cambio y muestra del nuevo stock

R3: Nombre: Generar reporte de inventario

Resumen: Permite consultar un resumen del inventario y varios datos relevantes

Entrada: Opción de filtrado según los diversos elementos

Salida: Reporte con los filtros usados

R4: Nombre: Control de pedidos a proveedores

Resumen: Permite registrar pedidos según artículo, fechas, cantidad y precio.

Entrada: código, cantidad a pedir y precio

Salida: confirmación de registro

Proyecto #2 "Gestor de tareas para equipos de progreso"

R1: Nombre: Crear tarea nueva

Resumen: Permite asignar una tarea del proyecto

Entrada: título, descripción, fecha límite, encargado

Salida: Confirmación de creación de la tarea

R2: Nombre: Asignar tarea

Resumen: permite cambiar o agregar encargados a una tarea

Entrada: ID de tarea, miembro al que se asigna

Salida: confirmación de asignación

R3: Nombre: Marcar tarea como completa

Resumen: Permite marcar una tarea como completa

Entrada: ID de tarea

Salida: Confirma completación de tarea, y la tache de una lista

R4: Nombre: seguimiento del progreso

Resumen: Permite ~~ver~~ visualizar el progreso del proyecto

Entrada: Ninguna

Salida: Gráfica o tabla que muestra el progreso general