

TRABAJO LOGICA COMPUTACION

JUAN SEBASTIAN CABRERA MARQUEZ

JOHAN DAVID GUERRA TIQUE

Guia logica Computacional

A) Resumen es ciclo de vida de construcccion de un Programa

Planificacion: Se define los objetivos del software, se identifican los requisitos y se crea un Plan y un Presupuesto. Esta fase es crucial para afianzar que todos los implicados esten alineados.

Analisis de Requisitos: Se recopilan y documentan los requisitos del software. Es fundamental saber que necesita el cliente y usuario y como debe funcionar el sistema para satisfacer esas necesidades.

Diseño: Basado en los requisitos, se crea el diseño del software. Esto incluye arquitectura del sistema, la interfaz del usuario y los detalles tecnicos que guiaran el desarrollo.

Desarrollo: Los Programadores escriben el codigo segun el diseño previamente establecido. Aqui se construye el software real.

Pruebas: Antes de lanzar el software, se hacen pruebas para verificar y corregir errores. Esta fase asegura que el software funcione correctamente y cumpla con los requisitos.

Implementacion: Una vez probado, el software se pone en funcionamiento. Esta fase incluye la instalacion de software en el entorno del usuario.

Mantenimiento: Mejoras y optimizaciones del software.

B) Aspectos del Analisis del Problema

Identificacion del Problema: Es crucial definir claramente cual es el problema. Esto incluye entender el problema y su naturaleza, alcance y las partes involucradas.

Recoleccion de datos: Recopilar toda la informacion relevante que pueda ayudar a entender mejor el problema. Esto incluye datos cualitativos y cuantitativos.

Analisis de causas: Determinar las causas principales del problema. Esto puede implicar el uso de herramientas como el diagrama de causa y efecto.

Evaluacion de impacto: Analizar como el problema afecta a la organizacion o el contexto en el que se presenta esa situacion.

Generacion de hipotesis: Proponer posibles soluciones.

Validacion de hipotesis: Probar la hipotesis para ver si realmente ~~nece~~
Resuelve el problema o necesita ajustes

c) Etapas del Proceso de Solucion del Problema

Definicion del Problema: Clarificar y dividir el problema de manera precisa.

Analisis del Problema: Desglosar el problema en partes mas pequenas y analizar cada una de ellas.

Generacion de Soluciones: Búsqueda de ideas.

Evaluacion de Soluciones: Evaluar las soluciones propuestas en terminos de viabilidad, costo, tiempo y recursos.

Seleccionar la mejor opcion.

*Proyectos de construcción o diseño

1. Planos finales
2. Especificaciones técnicas
3. Informe de inspección
4. Certificado de finalización
5. Garantía de calidad

x Otros

1. Presentaciones o informe de avances
2. Documentación de reuniones
3. Correo electrónico de confirmación
4. Informe de problemas resueltos
5. Documentación de seguimiento

* Documentos:

1. Informe de entrega (Delivery Report)
2. Documentación técnica (Manual de usuario, guía de instalación, etc...)
3. Contrato de acuerdo firmado
4. Factura o recibo de pago
5. Certificado de garantía (si aplica)

* Productos digitales:

1. Archivos de diseño (imágenes, logos, etc...)
2. Código de fuente (si se trata de desarrollo de software)
3. Bases de datos o contenido
4. Documentos de configuración
5. Archivos de instalación (si corresponde)

* Servicios

1. Informe de servicios prestados
2. Detalle de horas trabajadas
3. Resultados de pruebas o análisis
4. Recomendaciones para futuras mejoras
5. Documentación de procesos

c) Tarea #1 (pag 5)

1. Definición del problema: Describe claramente el problema que se necesita resolver.
2. Contexto: Explica el entorno en el que ocurre el problema.
3. Restricciones: Enumera limitaciones que deben considerarse.
4. Objetivos: Define que se espera lograr al resolver el problema.
5. Recursos disponibles: Identifica los recursos que tienes para abordar el problema.

9) Tarea #3

Para el caso de estudio 3, un programa para
Manejar un triángulo identifique y clasifique
3 requerimientos funcionales

Requerimiento #1

Nombre: Verificación de tipo de triángulo

Resumen: El sistema debe permitir al usuario
ingresar los longitudes de los
3 lados del triángulo

Entradas: longitud A
 longitud B
 longitud C

Resultados: El sistema determina y muestra
las características del triángulo

D M A

Programa funcional #2

Nombre: Área del área del triángulo

Resumen: el programa debe calcular el área del triángulo

Entradas: • Base del triángulo
• Altura del triángulo

Resultado: Se mostrará el área del triángulo en unidades cuadradas usando la fórmula correspondiente según las entradas proporcionadas

Programa funcional #3

Nombre: Perímetro del triángulo

Resumen: ingresar 3 las longitudes de 3 lados forman un triángulo rectángulo

Entradas: longitud A
longitud B
longitud C

Resultado: se mostrará si las longitudes ingresadas forman

H) tarea #4

Entidad	Nombre	Descripción
1	triángulo	es la entidad principal del problema que se presenta
2	lado	cada uno de los 3 segmentos que conforman un triángulo
3	ángulo	es el espacio entre 2 lados
4	area	es una entidad abstracta que representa la medida de la superficie

i) tarea #5

¿que pasa si no identificamos bien los entigados del texto?

si identificamos erróneamente los entigados del problema se pueden generar reglas conceptuales incorrectas, colados equivocados, confusiones entre los entigados y complejidad innecesaria del solución

CDT		
Atributo	Nombres Posibles	
# Certificado	→	Cadena de Caracter numerico
Monto Invertido	→	# decimal
Tasa de Interes	→	# decimal
Plazo	→	# decimal/entero
Fecha de Vencimiento	→	Fecha

Diagrama UML
CPT
o # Certificado
o Monto Invertido
o Tasa de Interes
o Plazo
o Fecha de Vencimiento

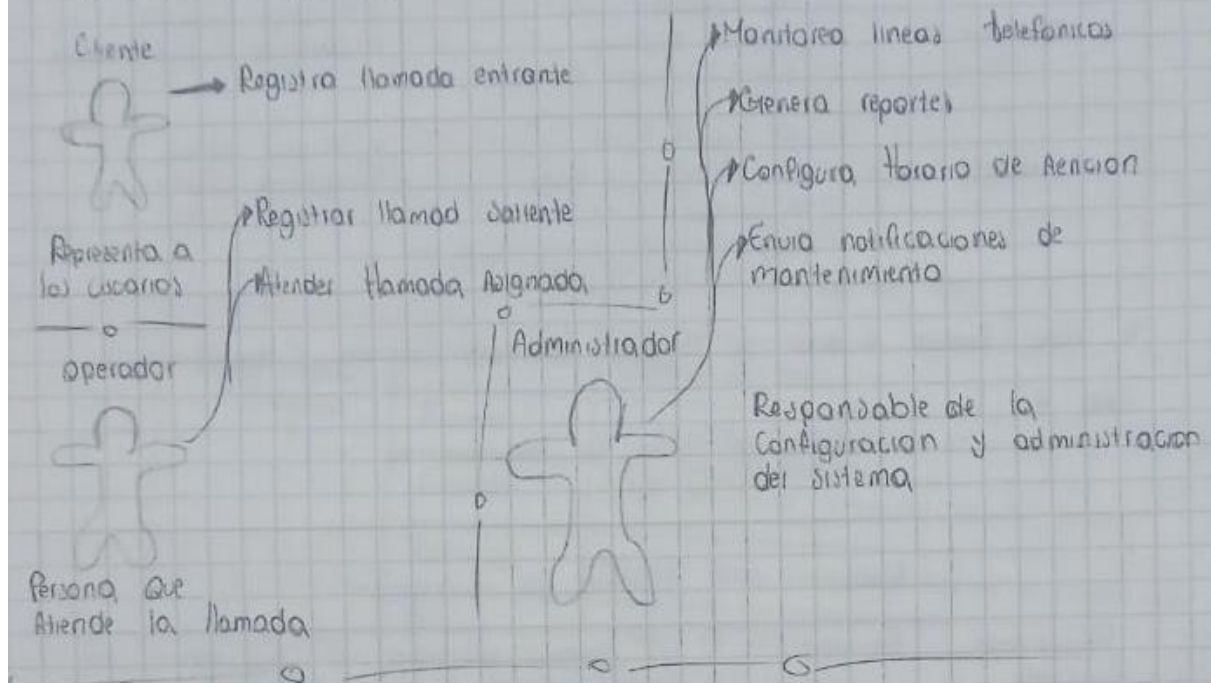
K. Estudiar los siguientes aspectos del ejemplo seleccionado: Enunciado, Requerimiento Funcionales (Casos de uso) y el modelo (Clases del Proyecto). Ar. A continuación redacta el enunciado del Problema y el nombre de cada uno de los Procedimientos.

La empresa "X" quiere implementar una aplicación para gestionar sus 3 líneas telefónicas con el objetivo de mejorar la organización y el manejo de las llamadas entrantes y salientes. También debe generar reportes de uso y hacer un seguimiento del estado de estas.

Requerimientos Funcionales:

- Registrar llamada entrante: La aplicación debe permitir registrar llamadas entrantes guardando la información relevante.
- Registrar llamada saliente: La APP debe permitir registrar llamadas realizadas por los operadores.
- Asignar llamada: Debe permitir asignar automáticamente la llamada a algún operador disponible.
- Monitorear llamada: Debe mostrar el estado de las llamadas.
- Generar reportes: Reportes periódicos bajo demanda que incluya el uso de la línea.
- Horarios de Atención: Debe permitir configurar horarios de atención de cada línea.

1. Dibja el respectivo diagrama de casos de uso del ejemplo elegido (consulta "Requerimiento funcionales" en un Archivo).



Me Observa nuevamente el modelo Conceptual del caso y escribe el nombre de cada uno de las clases identificando cada una de sus variables y funciones

	Atributos	funciones
Línea telefónica	<ul style="list-style-type: none"> o # línea & string o Estado & string o Duración llamada & float o Tipo & string 	<ul style="list-style-type: none"> o Registrar llamada o Actualizar estado
Llamada	<ul style="list-style-type: none"> o # teléfono & string o fecha & Date o hora inicio & Time o Duración & float o Tipo & string 	<ul style="list-style-type: none"> o iniciar llamada o finalizar llamada

	Atributos	funciones
Operador	<ul style="list-style-type: none"> o ID Operador & int o Nombre & String o disponibilidad & Boolean 	<ul style="list-style-type: none"> o Asignar llamada o Cambiar disponibilidad
Reporte	<ul style="list-style-type: none"> o ID Reporte & int o fecha Generacion & fecha o # Llamadas & int o tiempo Promedio & int / float 	<ul style="list-style-type: none"> o Generar reporte o Exportar reporte

n. Debes Plantear 2 ideas de Proyecto (Problemas solubles y algorítmicos)

Idea 1

Sistema de gestión de inventario en tiempo real

n	Nombre	Monitoreo de inventario
Requerimiento funcional 1	Resumen	El sistema realizará un seguimiento en tiempo real de los niveles de inventario en diferentes ubicaciones
	Entradas	Datos de inventario actual, Ubicación de Almacenamiento
	Resultado	Inf. Actualizado sobre los niveles de inventario a tiempo real
Requerimiento funcional 2	Nombre	Notificaciones de Reabastecimiento
	Resumen	El sistema envía alertas cuando los niveles de inventario caigan por debajo de un umbral
	Entradas	Niveles de inventario, Umbrales de Reabastecimiento
Requerimiento funcional 3	Resultado	Notificación Automática para reabastecer
	Nombre	Optimización de Almacenamiento
	Resumen	El sistema sugiere la mejor disposición de los productos en el almacén para optimizar el espacio
	Entradas	Historial de ventas, datos de mercado
	Resultado	Predicción de demanda y inventario óptimo

Requerimiento funcional 4	Nombre	Análisis de demanda
	Resumen	Predicción de la demanda futura de Productos para optimizar el inventario
	Entrada	Historial de ventas, Datos de Mercado
	Resultado	Predicción de demanda para optimizar Ventas

idea 2

Sistema de monitoreo de Calidad de Aire

Requerimiento funcional 1	Nombre	Contorno de datos
	Resumen	El sistema recopila datos de calidad de Aire de los Sensores
	Entrada	Datos del sensor, Ubicación de los Sensores
	Resultado	Base de datos con Información A tiempo real
Requerimiento funcional 2	Nombre	Análisis de datos
	Resumen	Analiza los datos recopilados y genera visualizaciones del estado del Aire
	Entrada	Calidad del Aire, Herramienta de visualización
	Resultado	Gráficas de la calidad del Aire
Requerimiento funcional 3	Nombre	Alerta de Notificación
	Resumen	El sistema Alerta cuando el Aire este Muy Sucio
	Entrada	Calidad Aire, Umbrales Contaminación
	Resultado	Notifica a tiempo real el estado del Aire
Requerimiento funcional 4	Nombre	Predicción Calidad del Aire
	Resumen	Recopila datos Históricos del Aire y Predice la calidad del Aire
	Entrada	Datos Históricos, Factores externos
	Resultado	Predicción de Calidad del Aire para tomar acciones


```
#include <iostream>

using namespace std;

int main() {
    string matricula[4];
    double capacidadCarga[4];
    double consumoGasolina[4];
    double cargaActual[4];
    for (int i = 0; i < 4; i++) {
        cout << "Ingrese los datos del camion " << i + 1 << endl;

        cout << "Matricula: ";
        cin >> matricula[i];

        cout << "Capacidad de carga (kg): ";
        cin >> capacidadCarga[i];

        cout << "Consumo de gasolina (galones/km): ";
        cin >> consumoGasolina[i];

        cout << "Carga actual (kg): ";
        cin >> cargaActual[i];

        cout << endl;
    }
}
```

```
cout << "Informacion de los camiones:" << endl;

for (int i = 0; i < 4; i++) {

    cout << "Camion " << i + 1 << ": " << endl;

    cout << "Matricula: " << matricula[i] << endl;

    cout << "Capacidad de carga: " << capacidadCarga[i] << " kg" << endl;

    cout << "Consumo de gasolina: " << consumoGasolina[i] << " galones/km" <<
endl;

    cout << "Carga actual: " << cargaActual[i] << " kg" << endl;

    cout << endl;

}

return 0;

}
```