

Capítulo 4

Etapas del desarrollo

Este capítulo documenta la aplicación del modelo presentado anteriormente, para el caso de la detección y clasificación de eventos sísmicos sobre señales digitales. El problema presenta las condiciones que permiten categorizarlo en el dominio del modelo: entrada de datos, análisis o procesamiento, salida y respuestas.

El modelo descrito, constituye una base conceptual que establece métodos y definiciones para problemas del tipo de detección y clasificación de señales digitales. Como ciclo de vida del producto software, el modelo establece los pasos para realizar el análisis y diseño global del problema, permitiendo la definición y ejecución de las etapas descritas para construir la aplicación de software. Como soporte del sistema crítico ofrece la recopilación de criterios y factores que guían las fases de construcción, vinculando todo el proceso de desarrollo con el concepto de calidad de un producto de software.

4.1. Planeación

4.1.1. Factibilidad

4.1.1.1. Técnica

De acuerdo a la revisión bibliográfica, se conoce la existencia de métodos y procedimientos que pueden ayudar al alcance de las metas del proyecto. A fin de contemplar más alternativas de solución se buscarán nuevos mecanismos que definan los procesos de construcción de la herramienta de software.

4.1.1.2. Económica

El **OSSO** proveerá los recursos humanos, equipos y telecomunicaciones para el desarrollo del proyecto.

4.1.1.3. Operacional

Los ítem 1.1.4 y 1.1.5 del Capítulo 1 aportan elementos que sustentan la importancia del problema y algunas alternativas de solución existentes.

4.1.2. Actividades

Siguiendo la identificación de actividades del modelo se tiene:

Aplicación del modelo:

- Revisión de Análisis global: a realizar en la descripción del sistema actual.
- Revisión del Diseño global: Arquitectura general e identificación del componente crítico.

Diseño de etapas:

- Análisis detallado: Identificado como Etapa 0.
- Identificación de etapas: A partir de la etapa 1 hasta integración de componente y pruebas.

Diagnóstico y Mejoras:

- Aplicación de planes de prueba del sistema.
- Procesos posteriores al desarrollo para evaluar validez de los procesos.

4.1.3. Sistema de calidad

Se consideran los siguientes elementos para uso durante el diseño y codificación:

- Uso del ciclo de vida de entrega por etapas, según la descripción del modelo.
- Uso de patrones de diseño GRASP y conceptos de diseño Orientado a Objetos.
- Uso de un sistema para manejo de código (CVS).
- Generación de archivos de registros del sistema (*LOGs*).
- Codificación y asignación de errores y alertas sobre el sistema. Favorece en la detección de fallos presentados durante la ejecución del sistema.
- Formatos para aplicación del plan de pruebas de diseño y funcionalidad

4.2. Descripción del sistema actual

Para tener una correcta identificación de las variables y condiciones del sistema, se propone la descripción formal de las características propias del sistema y sus necesidades.

4.2.1. Análisis detallado (Etapa 0)

Siguiendo los métodos propuestos en [CL99], se realizará el análisis detallado del proyecto.

4.2.1.1. Definición del problema

Como parte del proceso de observación sísmológica, los casos de mayor interés son la ocurrencia de cambios o fenómenos sobre las señales; estos casos pueden presentarse en cualquier momento y tener diversas causas de origen. Se desea conocer cuando las señales detectadas son producidas por movimientos del subsuelo (eventos sísmicos) y buscar un esquema de clasificación basado en las características de la señal. La definición más completa de este punto se encuentra en los ítem 1.1 y 1.2 del Capítulo 1.

4.2.1.2. Listado detallado de requerimientos

1. Lectura de datos de entrada en formato WVM.
2. Lectura y conversión de datos desde archivos almacenados previamente.
3. Mecanismo de visualización de los datos actualmente en uso.

4. Mecanismo de configuración de códigos de estaciones y un coeficiente que indique calidad de la señal (valor numérico denominado peso).
5. Registro de funcionamiento del sistema mediante LOGs (mensajes a consola).
6. Determinar la ocurrencia de eventos de interés y establecer un mecanismo de clasificación.
7. Registrar e informar la ocurrencia de eventualidades sobre las señales.

4.2.1.3. Descripción Casos de uso

Como metáfora de uso del sistema, se pueden considerar los siguientes elementos.

Ver Figura 4.1.

Se presentan dos actores:

- **Usuario:** puede visualizar información suministrada por el sistema de adquisición y los parámetros de configuración utilizados.
- **Operador:** Visualiza y monitorea los estados de operación del sistema. Los posibles errores en la entrada de datos, en la ejecución de los módulos internos, o en las salidas del sistema serán reportados al registro de mensajes del sistema. El operador debe realizar alguna acción para restaurar la operación completa de la herramienta.

Cinco casos de uso:

- **Visualización y configuración:** Visualizar la señal de las estaciones sísmológicas. Realizar configuración para la operación del sistema.

- **Captura de señal:** Leer la señal del sistema de adquisición digital. Se utilizará el protocolo de datos nativo del sistema de adquisición. Se registran los estados de funcionamiento de los componentes asociados.
- **Detección:** Indica si un conjunto de señales muestran comportamientos o presencia de eventualidades.
- **Clasificación:** de acuerdo a la salida otorgada por la detección, la clasificación dará lugar a determinar la clase de comportamiento de la señal entrante.
- **Alerta:** respuesta del sistema de acuerdo a la salida del caso de detección.

4.2.1.4. Modelo Conceptual

Como una primera aproximación al diseño detallado, los objetos funcionales sobre el sistema se indican a continuación. Se muestra en este modelo conceptual la interrelación entre los componentes. Figura 4.2.

4.2.1.5. Diagramas de Secuencia del Sistema

Los diagramas de secuencia muestran una generalización del paso de mensajes entre los actores y casos de uso detectados. Estos diagramas ayudan a comprender el funcionamiento y la interacción de componentes del sistema.

- Caso de uso Visualización y configuración. Figura 4.3.
- Caso de uso Captura de señal. Figura 4.4.

S-DCE : Casos de uso

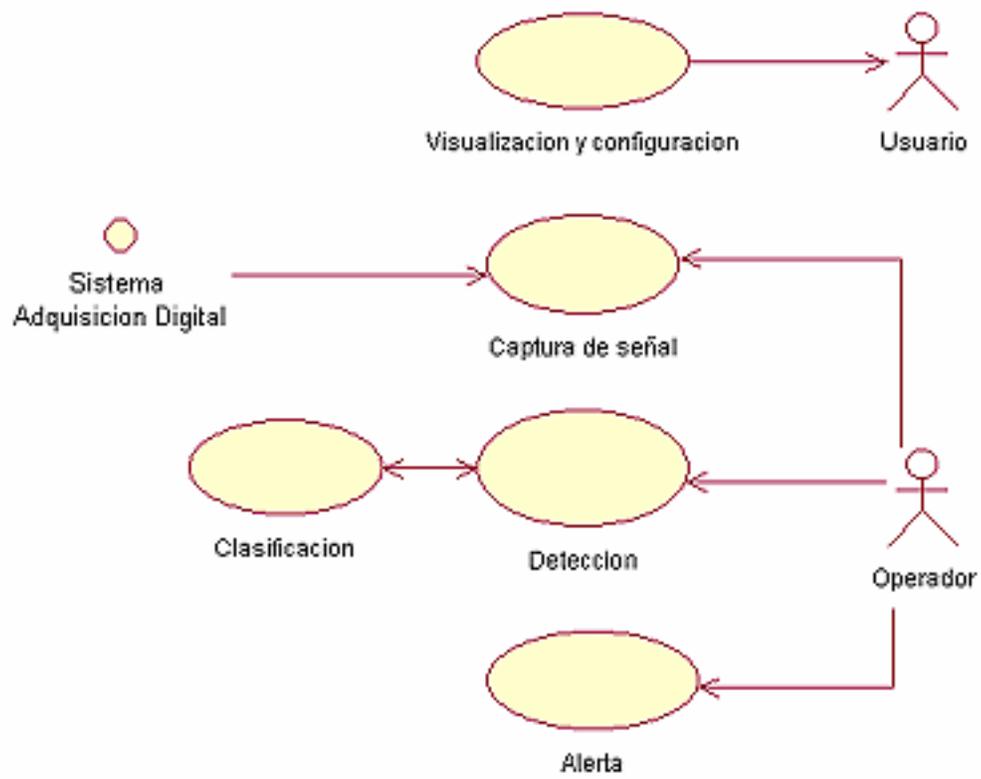


Figura 4.1: Casos de uso

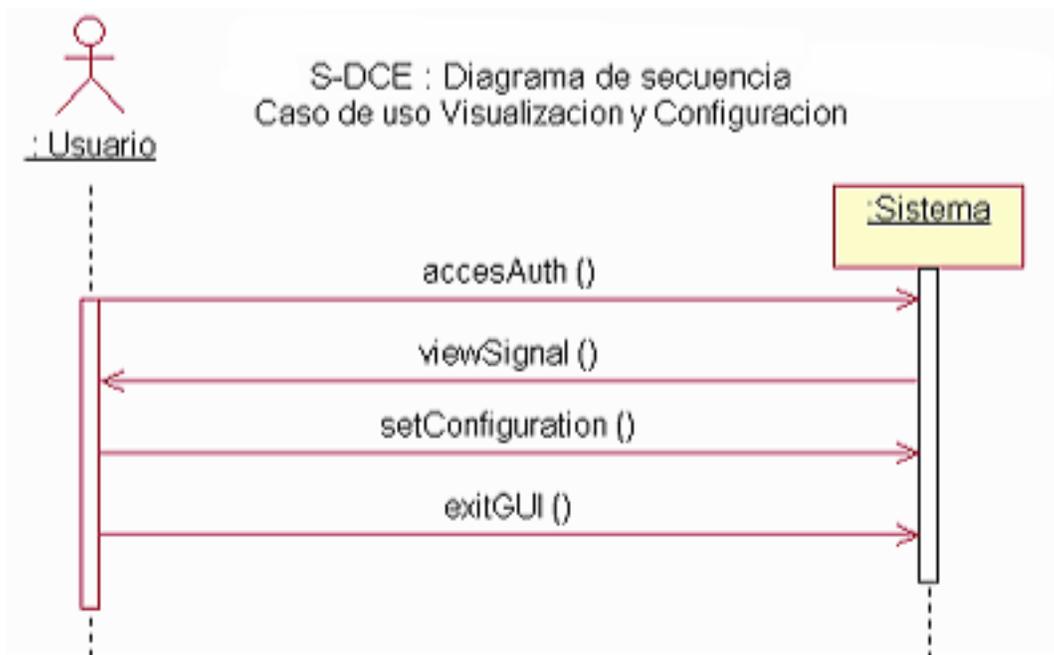


Figura 4.3: DSS - Visualización y configuración

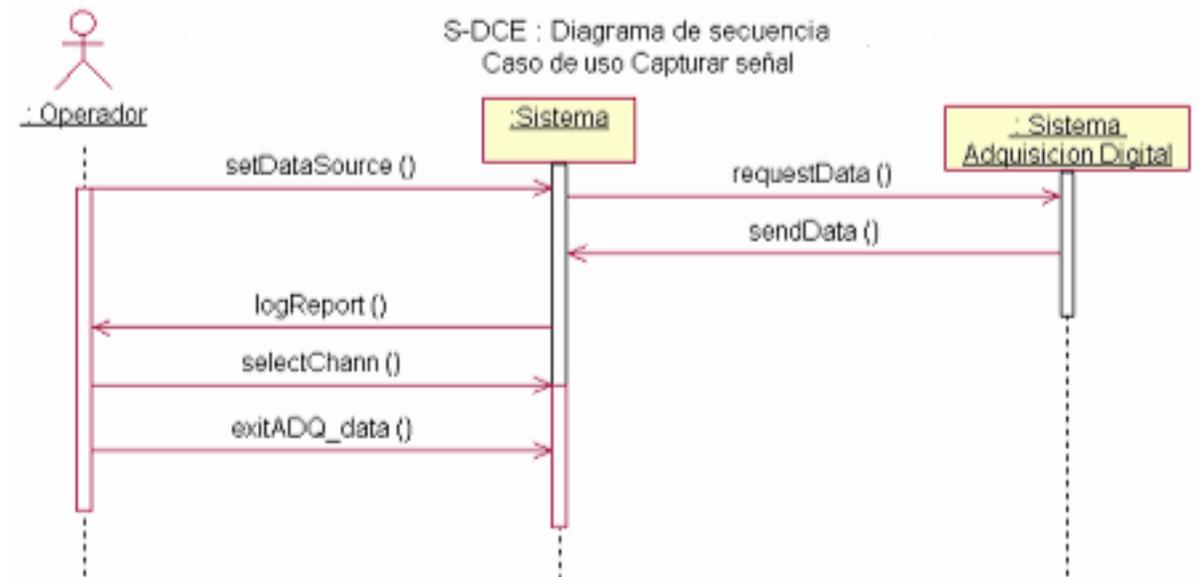


Figura 4.4: DSS - Captura de señal

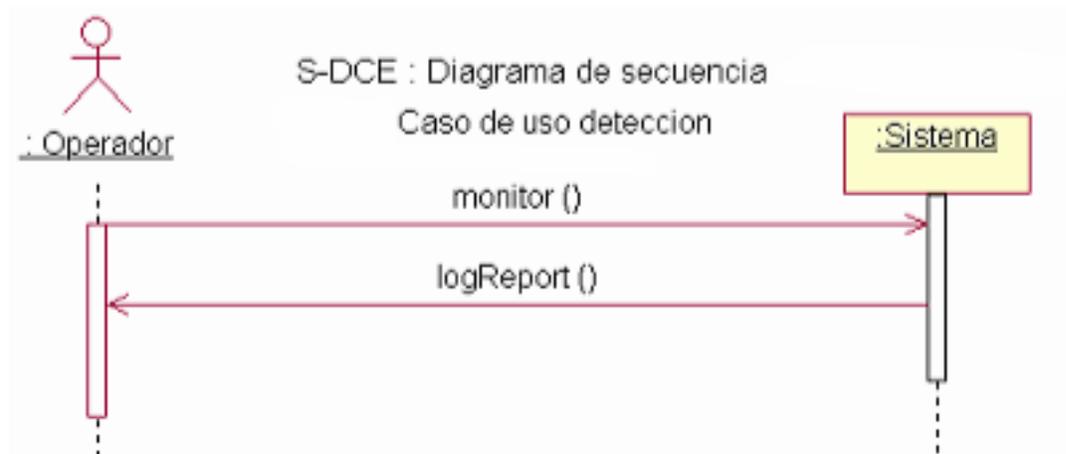


Figura 4.5: DSS - Detección

- Caso de uso Detección. Figura 4.5.
- Caso de uso Clasificación. Figura 4.6.

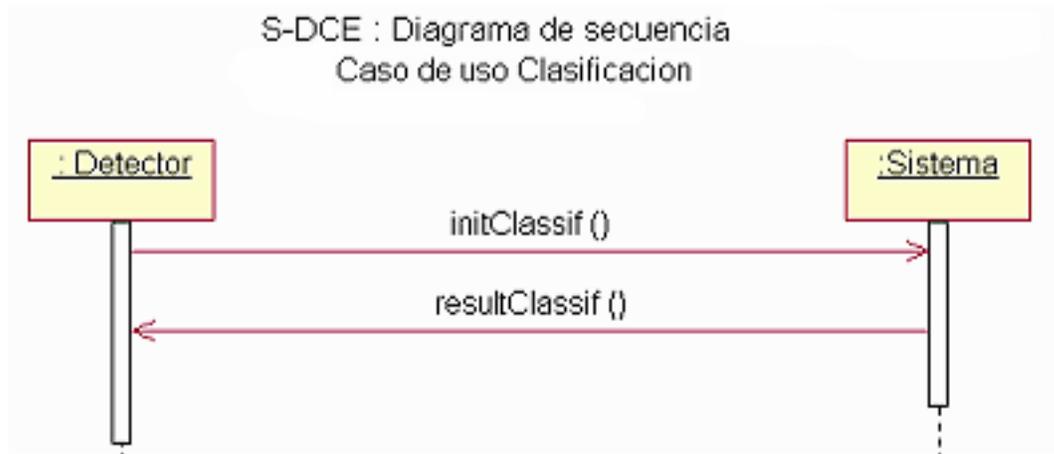


Figura 4.6: DSS - Clasificación

- Caso de uso emitir alerta. Figura 4.7.

4.2.2. Definición de las etapas

4.2.2.1. Listado de componentes

De acuerdo a la estructura general del modelo y con el análisis detallado del sistema a tratar, se pueden deducir algunos componentes funcionales:

- Entrada de datos
- Detección de cambios

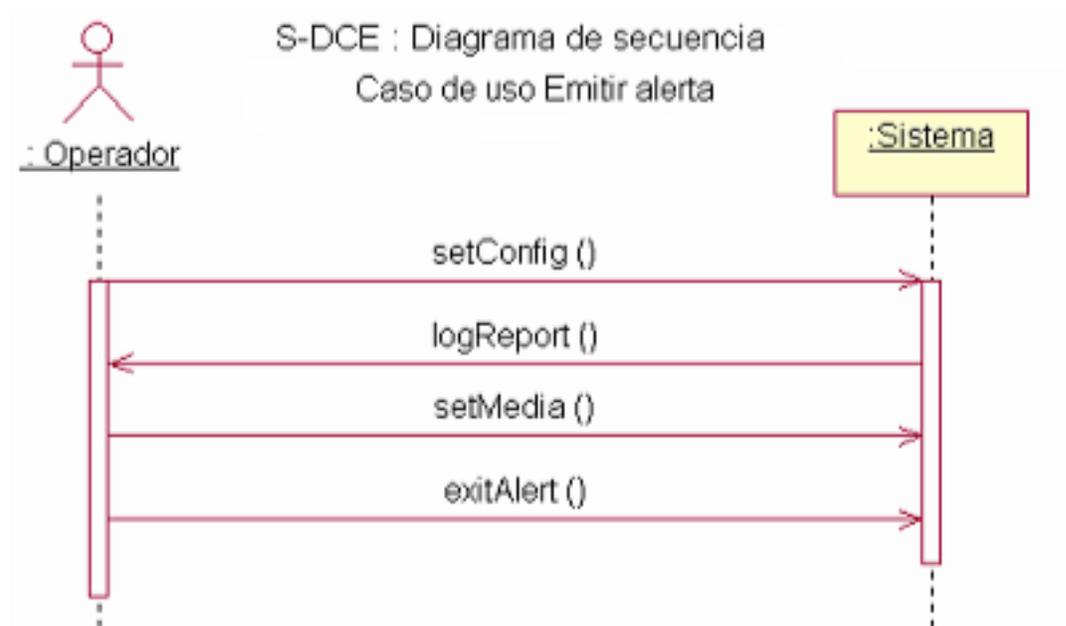


Figura 4.7: DSS - Emisión de alerta

- Clasificación de eventualidades
- Alerta

Adicionalmente, se requieren algunos componentes ocultos para el usuario, pero de uso en la aplicación:

- Protocolos y funciones para comunicaciones.
- Protocolos y funciones para manejo de las estructuras de datos.
- Sistema de registro de mensajes en consola.

4.2.2.2. Ordenamiento

Se puede determinar el orden de construcción de los componentes, buscando reducir el riesgo potencial de la entrega por etapas: sensibilidad en cambios de requerimientos sobre etapas tempranas. Se designarán como etapas tempranas las que ofrezcan mayor independencia de diseño y que definan protocolos y estructuras de información necesarias en etapas posteriores. De acuerdo a estos factores, el orden de las etapas será :

- **Etapa 1:** Comunicaciones y registro
Corresponde a las estructuras y mecanismos de software que permiten realizar la comunicación entre los módulos internos, el registro de mensajes en consola (LOG) y la manipulación de estructuras de datos.
- **Etapa 2:** Entrada de datos
Define el mecanismo de entrada de datos, estableciendo comunicación con

la fuente que suministra los datos, convirtiéndolos en la estructura interna adecuada.

- **Etapa 3:** Detector de cambios

Este componente comprende la implementación del sistema que permite detectar los cambios sobre las señales de entrada.

- **Etapa 4:** Clasificador de eventualidades

Este componente, aunado al trabajo del detector realiza la clasificación de las señales ingresadas.

- **Etapa 5:** Sistema de Alerta

La alerta constituye el actuador o respuesta del sistema. Puede ser vista como un mecanismo de comunicación hacia otros sistemas o un mecanismo de retroalimentación.

- **Etapa 6:** Integración de componentes

El modelo sugiere la inclusión de una última etapa para la integración de componentes y pruebas. Esto permite documentar y aplicar de manera homogénea planes de prueba.

4.2.3. Arquitectura de la herramienta

4.2.3.1. S-DCE : Señales - Detector y Clasificador de Eventualidades

El sistema S-DCE corresponde a la implementación del diseño modular y Orientado a Objetos. Este sistema se encargará de tomar la señal de una fuente de adquisición de datos, detectar y clasificar las señales de entrada de acuerdo a los criterios definidos internamente y emitir una alerta, según sea necesario. Paralelamente

proporciona un entorno para la visualización y configuración del detector, indicando parámetros generales para el funcionamiento del sistema. Figura 4.8.

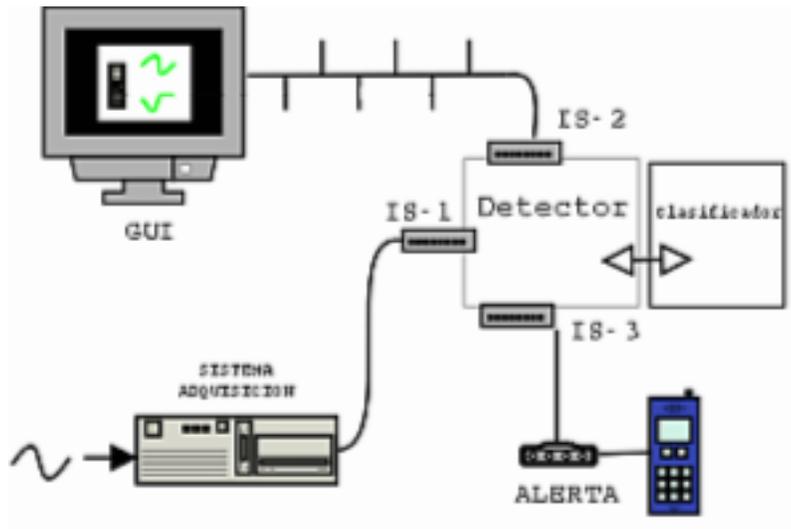


Figura 4.8: Arquitectura de componentes

4.2.3.2. Componentes

GUI: Interfaz de usuario.

Sistema adquisición: fuente de señales digitales.

Alerta: Actuador o respuesta.

Detector: Mecanismo para la detección de cambios abruptos.

Clasificador: Mecanismo para la clasificador de señales.

IS-1: Interfaz del Sistema 1 para el módulo de adquisición de datos.

IS-2: Interfaz del Sistema 2 para el módulo de visualización y configuración.

IS-3: Interfaz del Sistema 3 para el módulo de alerta.

4.2.3.3. Características generales

- *Plataforma* : GNU/Linux
- *Lenguajes de programación* : GNU/C y GNU/C++, Java, Perl.
- *Bibliotecas del sistema* : proporcionadas por las librerías propias del sistema operativo
 - Comunicaciones por sockets.
 - Manejo de memoria, cadenas, operaciones matemáticas.
 - Registro de mensajes a consola (syslog).

4.2.4. Sección crítica

Las condiciones de estabilidad del sistema están dadas en gran medida por la gestión realizada en el control de calidad del producto de software. Este control se aplica durante los planes de prueba, especialmente en los componentes críticos.

Para la aplicación puntual, los componentes críticos serán denominados núcleo, ya que en cierta medida los otros módulos trabajan alrededor de éste. El núcleo se define según los objetivos primarios del sistema, en este caso realizar la detección de cambios sobre el sistema. Cualquier fallo sobre los mecanismos de entrada o salida de datos deberán ser contemplados por el núcleo.

Los alcances de la sección crítica del sistema, se encuentran limitados a las condiciones de funcionamiento del sistema operativo y por ende al desempeño del hardware de la máquina.

4.3. Etapa 1: Comunicaciones y control

4.3.1. Descripción

Esta etapa busca crear los métodos y funciones para :

- Realizar la transmisión de datos entre los módulos, utilizando sockets (Mas información A.1.3).
- Realizar el registro de mensajes en consola, utilizando **syslog** (*Linux system logging utilities*).
- Realizar el manejo de las estructuras de datos descritas en A.1.1 y A.1.2.

4.3.2. Protocolos

Las especificaciones de las estructuras de datos utilizadas se encuentran consignados en el Anexo A, A.1.1 y A.1.2.

4.3.3. Codificación

La codificación de estos componentes comprenden la generación de librerías y clases que sirvan para realizar :

1. Operaciones de transmisión de información localmente o por red de datos utilizando un modelo cliente servidor.
2. Registro de mensajes del sistema.
3. Funciones para el manejo de las estructuras DSRCP y XMLDSRC.

4.3.4. Pruebas

Para esta etapa se utilizarán las siguientes fichas de prueba:

Caso de prueba :	Transmisión de datos
Objetivo :	Realizar transmisión de datos mediante el modelo de sockets
Configuración :	PC i386 con sistema operativo Linux
Entradas :	Archivos planos y binarios.
Salidas :	Copia de los archivos enviados.
Descripción :	Se realiza la transmisión de archivos planos y binarios.
Resultados :	La transmisión de datos utilizando sockets se realizó con éxito.

Cuadro 4.1: Prueba: Transmisión de datos

Caso de prueba :	Registro de mensajes
Objetivo :	Realizar el registro de mensajes en <i>Consola</i>
Configuración :	PC i386 con sistema operativo Linux
Entradas :	Entrada del mensaje a registrar.
Salidas :	Formato de registro en <i>Consola</i> .
Descripción :	Se realiza el registro de mensajes en Consola, utilizando <i>syslog</i> .
Resultados :	Los mensajes son registrados con éxito.

Cuadro 4.2: Prueba: Registro de mensajes

4.4. Etapa 2: Entrada y visualización de señales

4.4.1. Descripción

4.4.1.1. Enlace al sistema de adquisición

Recepción y conversión del formato de los datos del sistema de adquisición. Este componente actúa como un servidor de datos para los otros módulos. Los datos quedan disponibles bajo la estructura XMLDSRC.

4.4.1.2. Cliente de interfaz gráfica

Visualiza las señales recibidas, desplegando las ondas para cada canal de datos. También provee un mecanismo simple para configurar la información de las estaciones y su nivel de calidad mediante un parámetro denominado *peso*.

4.4.2. Diseño detallado

4.4.2.1. Enlace al sistema de adquisición

El diagrama de colaboración para este componente se indica en la Figura 4.9.

4.4.2.2. Cliente de interfaz gráfica

El diagrama de colaboración para este componente se indica en la Figura 4.10.

4.4.3. Protocolos

Los detalles de los protocolos y estructuras de datos se encuentran en el Anexo A, ítem A.1.1 y A.1.4.

4.4.4. Codificación

4.4.4.1. Enlace al sistema de adquisición

El enlace al sistema de adquisición permite la conexión a una fuente de datos, correspondiente a archivos previamente almacenados. Adicionalmente estos datos se podrán acceder mediante un servidor vía Sockets utilizando *Threads* (hilos) que proveerá esta información a los otros módulos. Este módulo se encuentra implementado en /C/C++.

4.4.4.2. Cliente de interfaz gráfica

Como herramienta para visualizar la información adquirida, se cuenta con un cliente del enlace al sistema de adquisición. Este componente permite realizar una configuración simple de los canales de datos. Este cliente se implementa en Java y la visualización se realiza mediante *Applets*.

4.4.5. Pruebas

Se propone la siguiente ficha de prueba para esta etapa:

4.5. Etapa 3: Núcleo - Detector de cambios

4.5.1. Descripción

El detector de cambios corresponde al núcleo o sección crítica. En general este sistema es responsable de indicar donde ocurre un cambio importante sobre la señal,

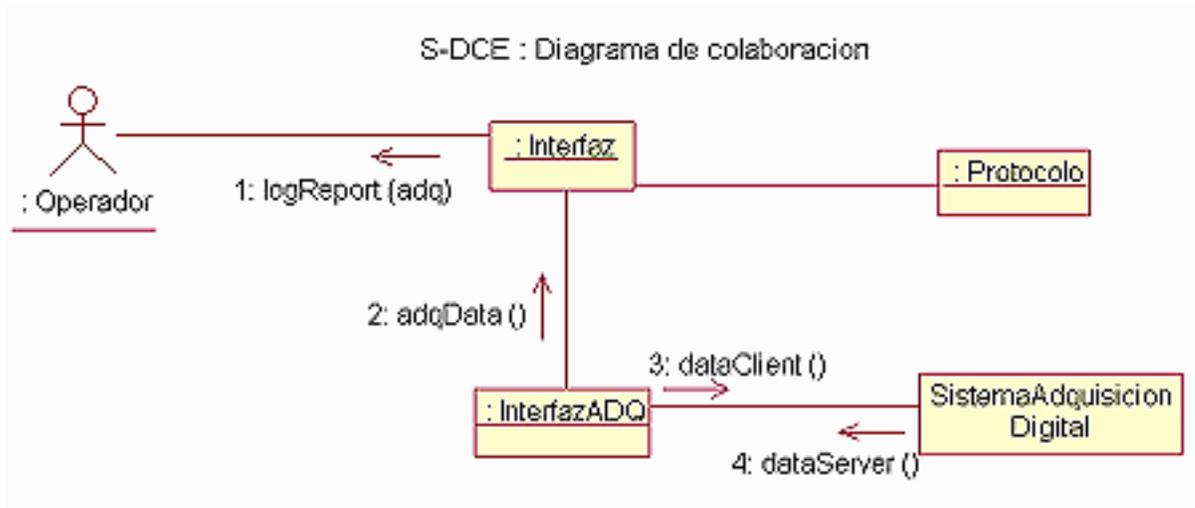


Figura 4.9: DColaboración - Captura de señal

Caso de prueba :	Enlace y visualización de datos
Objetivo :	Lee y visualiza los datos del sistema de adquisición.
Configuración :	PC i386 con sistema operativo Linux
Entradas :	Archivos en formato WVM.
Salidas :	Señal visualizada mediante aplicación.
Descripción :	Se realiza la lectura de archivos WVM. Se provee un mecanismo para suministrar los datos a otros módulos.
Resultados :	Los datos se leyeron y visualizaron correctamente.

Cuadro 4.3: Prueba: Enlace y visualización de datos

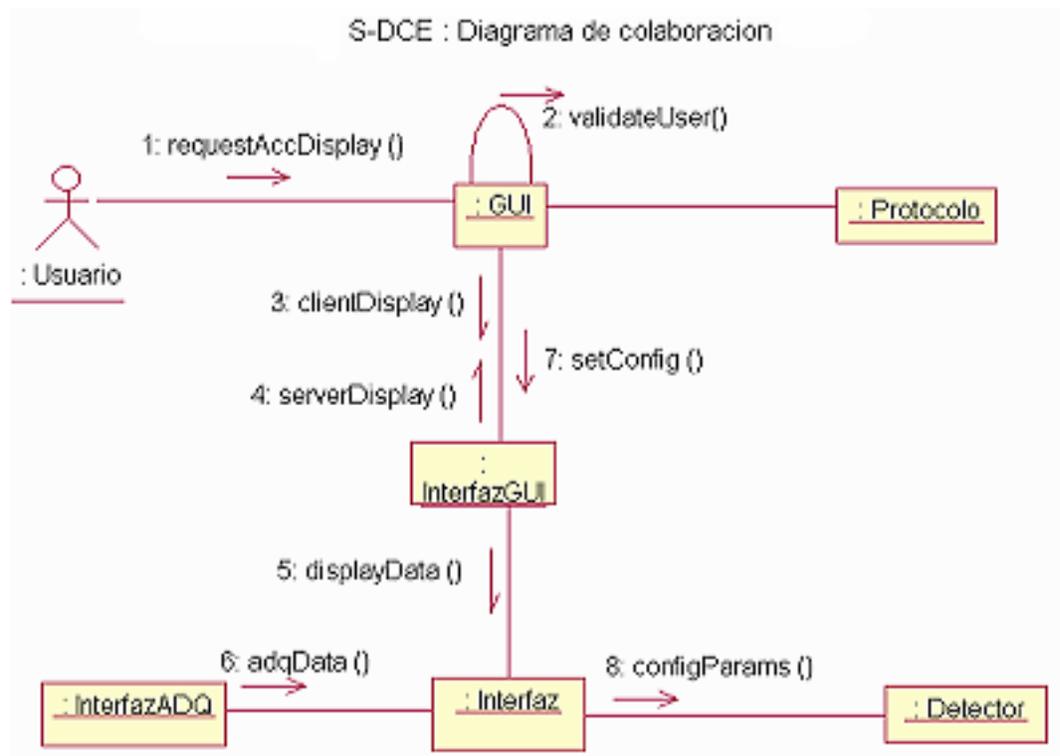


Figura 4.10: DColaboración - Visualización y configuración

bien sea que se presente atenuado o abrupto. Una vez se determine que ha ocurrido un cambio, se procede a llamar al módulo de clasificación, el cual opera en colaboración con este componente.

4.5.2. Diseño detallado

El diagrama de colaboración para este componente se indica en la Figura 4.11.

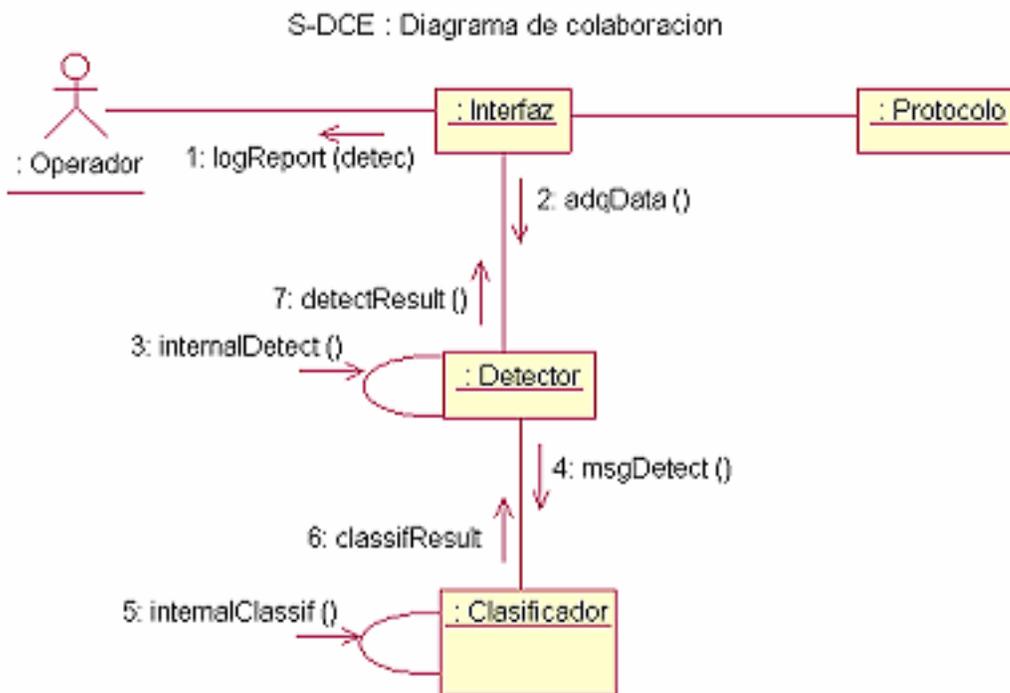


Figura 4.11: DColaboración - Detección y Clasificación

4.5.3. Protocolos

Los protocolos de entrada y salida de datos se encuentran definidos en el Anexo A, ítem A.1.

4.5.4. Codificación

Este es un cliente de datos que implementa la estructura XMLDSRC. Codifica el algoritmo STA/LTA (descrito en el Capítulo 2, ítem 2.6.1), aplicando algunos cambios para extraer características de las señales, basado en el estudio de [MDH02]. Luego, se realiza el llamado al proceso de clasificación.

4.5.5. Pruebas

Dado que este componente corresponde a la parte crítica del sistema, se define y aplica un plan exhaustivo para la detección de señales. Las fichas de prueba a utilizar son:

Caso de prueba :	Núcleo - Rendimiento del sistema
Objetivo :	Medir los niveles de carga y uso de recursos sobre la máquina.
Configuración :	PC i386 con sistema operativo Linux
Entradas :	Módulo de detección de cambios
Salidas :	Reporte sobre niveles de carga en el sistema.
Descripción :	Se revisa el consumo de recursos que presenta el sistema.
Resultados :	La máquina no presenta niveles altos de consumo de recursos, excepto para arreglos de datos muy extensos.

Cuadro 4.4: Prueba: Núcleo - Rendimiento del sistema

Caso de prueba :	Núcleo - Conexiones de entrada y salida
Objetivo :	Comprobar las funciones que realizan la entrada y salida de datos
Configuración :	PC i386 con sistema operativo Linux
Entradas :	Módulo de detección de cambios
Salidas :	Reporte de funcionamiento de las conexiones.
Descripción :	Se realiza supervisión en la operación de las funciones de entrada / salida de mensajes y datos.
Resultados :	Las conexiones de entrada y salida responden correctamente a los llamados. Cuando los módulos no se encuentran disponibles, el sistema reporta este suceso.

Cuadro 4.5: Prueba: Núcleo - Conexiones de entrada y salida

Caso de prueba :	Núcleo - Excepciones del detector
Objetivo :	Comprobar la respuesta del sistema en situaciones de error o no disponibilidad de recursos.
Configuración :	PC i386 con sistema operativo Linux
Entradas :	Módulo de detección de cambios
Salidas :	Reporte con excepciones del sistema
Descripción :	Se realiza la comprobación de respuesta del sistema ante entradas erróneas o fallos en el acceso a otros componentes.
Resultados :	El sistema realiza correctamente el registro ante los errores descritos.

Cuadro 4.6: Prueba: Núcleo - Excepciones del detector

4.6. Etapa 4: Clasificador de eventualidades

4.6.1. Descripción

Una vez las señales han sido declaradas como *eventualidad*, es de utilidad saber si corresponden a algún tipo de clasificación. Sin embargo, dada la naturaleza de las señales y la cantidad de variantes que las afectan, puede obtenerse, para alcances de este prototipo, una caracterización general, que resulta útil para identificar el tipo de señal.

A nivel de diseño este componente no tiene grandes requerimientos: básicamente el núcleo del sistema entrega cierta información ya procesada; el clasificador determinará si la eventualidad se encuentra en uno de los grupos de clasificación y retornará una respuesta al detector. Para la implementación de éste módulo, es posible utilizar una gran variedad de técnicas desde algoritmos basados en reglas, estadísticos, técnicas de Inteligencia Artificial o de métodos numéricos. Para clasificar eventos sísmicos existen diversos mecanismos, entre los cuales se destacan técnicas de Inteligencia Artificial 2.7, o aquellos basados en aplicación de filtros para transformar la señal a una equivalente que favorezca un determinado procesamiento. Este módulo implementará un filtro, un algoritmo para extraer características y aplicará un conjunto de reglas, con el fin de simplificar las operaciones.

4.6.2. Codificación

Para el caso de aplicación, el clasificador aplica un filtro sobre la señal, extrae un grupo de características relacionadas con la eventualidad detectada y determina

la clase a la que pertenece la señal basándose en algunas reglas derivadas de la configuración local de la red de estaciones del OSSO y algunos procedimientos simplificados de localización del evento.

Este módulo realiza la aplicación del filtro Discrete Wavelet Transform con Daubechies - 4 para extracción de características, basado en la experimentación de [GEM98]. Para realizar la clasificación se aplican un conjunto de reglas utilizadas por el Área de Sismología de la red del Sur Occidente del OSSO. Las características utilizadas para la clasificación son:

- Tiempo y amplitud del inicio del evento (T_p y A_p).
- Tiempo y amplitud máxima (T_s y A_s).
- Tiempo finalización del evento (T_c y A_c).

4.6.3. Pruebas

Se propone la siguiente ficha de prueba para esta etapa:

4.7. Etapa 5: Sistema de alerta

4.7.1. Descripción

Actuando como sistema de control, la alerta corresponde a la respuesta del sistema frente a la ocurrencia de eventualidades, de acuerdo al módulo de detección. Este módulo se encarga de recibir y enviar las alertas generadas por la detección y clasificación de eventualidades. La alerta puede tener múltiples variantes de acuerdo

Caso de prueba :	Núcleo - Funcionamiento del detector
Objetivo :	Comprobar si el detector funciona correctamente.
Configuración :	PC i386 con sistema operativo Linux
Entradas :	Módulo de detección de cambios y Módulo de conexión de entrada de datos.
Salidas :	Ocurrencia de cambios o eventualidades sobre las señales.
Descripción :	Se comprueba el nivel de funcionamiento de la detección de señales.
Resultados :	Se obtiene la caracterización de las señales donde ocurren cambios de interés (presencia de <i>eventualidades</i>).

Cuadro 4.7: Prueba: Núcleo - Funcionamiento del detector

Caso de prueba :	Clasificación de eventualidades
Objetivo :	Verificar el funcionamiento de la clasificación
Configuración :	PC i386 con sistema operativo Linux
Entradas :	Caracterización de las señales
Salidas :	Tabla de clasificación de las señales
Descripción :	Se realiza la verificación en el correcto funcionamiento de la clasificación de las señales.
Resultados :	Se retorna un registro con los resultados de la clasificación.

Cuadro 4.8: Prueba: Clasificación de eventualidades

a las necesidades o configuración del sistema. Para el caso de implementación, la función *setMedia()* soporta en envío de la alerta de texto por correo electrónico.

4.7.2. Diseño detallado

El diagrama de colaboración para este componente se indica en la Figura 4.12.

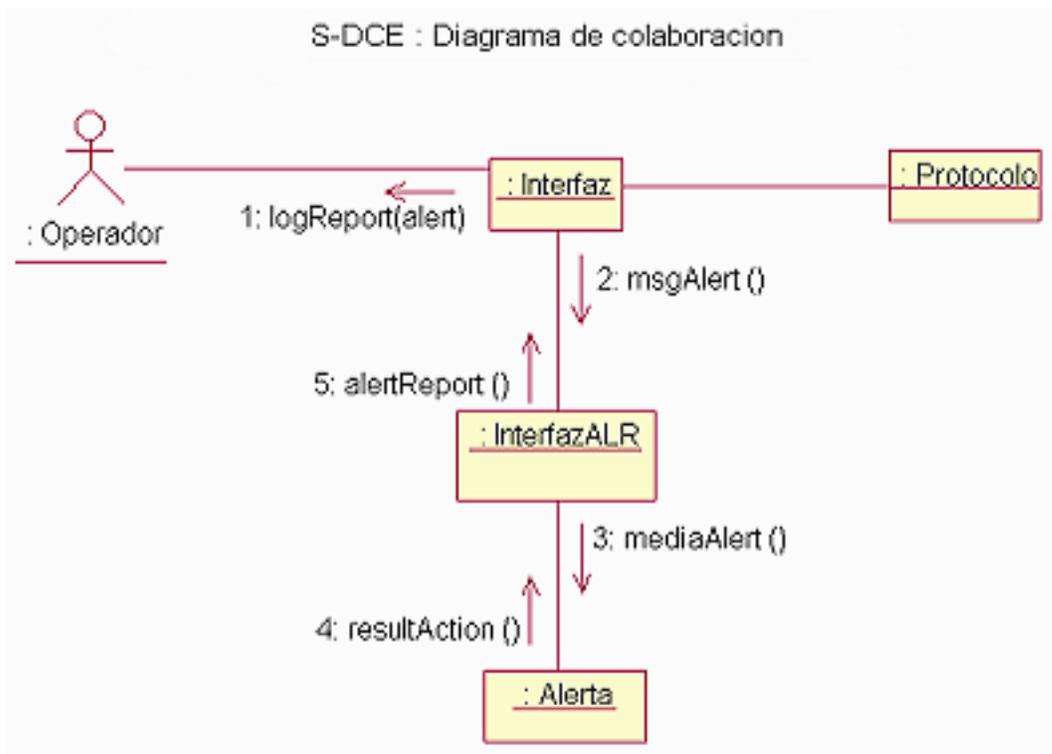


Figura 4.12: DColaboración - Emisión de alerta

4.7.3. Protocolos

De manera general, los tipos de alerta a ser manejados por el sistema corresponden

a:

- *Alerta S-DCE*: Fallos importantes sobre la funcionalidad del sistema se catalogan con este tipo de clasificación.
- *Alerta Técnica*: Errores externos a S-DCE que indiquen imposibilidad de funcionamiento en las operaciones.
- *Alerta de Eventualidad*: Ocurrencia de eventos sísmicos o de otra índole y que requieran atención, de acuerdo a lo establecido en la configuración de este módulo.

Estas alertas poseen clasificaciones internas dinámicas, que van cambiando de acuerdo a las condiciones y características resultantes en los módulos de detección y clasificación.

4.7.4. Codificación

Para la implementación de este componente se generará un reporte de eventualidad.

Los servicios que realiza la alerta son:

- Se provee un servidor que atiende los llamados del detector.
- Se realiza el registro de un reporte de alerta.
- Se ejecuta la acción de alerta de acuerdo a la configuración específica.

4.7.5. Pruebas

Se proponen la siguientes fichas de prueba para esta etapa:

Caso de prueba :	Servidor de alertas
Objetivo :	Verificar el funcionamiento del servicio de recepción de alertas
Configuración :	PC i386 con sistema operativo Linux
Entradas :	Mensajes de alerta de prueba desde el detector
Salidas :	Paso del mensaje al envío de alertas.
Descripción :	El servidor debe recibir y entregar el mensaje para que sea enviado.
Resultados :	El servidor entrega correctamente el mensaje.

Cuadro 4.9: Prueba: Servidor de alertas

Caso de prueba :	Envío del mensaje de alerta
Objetivo :	Realizar envío del mensaje de alerta.
Configuración :	PC i386 con sistema operativo Linux
Entradas :	Mensaje desde el servicio de recepción de alertas.
Salidas :	Envío del reporte de eventualidad.
Descripción :	Se debe recibir y construir el reporte de eventualidad.
Resultados :	En reporte es enviado correctamente.

Cuadro 4.10: Prueba: Envío del mensaje de alerta

4.8. Etapa 6: Integración de componentes y pruebas

4.8.1. Descripción

Como etapa final del ciclo de vida de desarrollo se realiza la integración de los módulos antes desarrollados. Adicionalmente se complementa y aplica el plan de pruebas definido por el modelo utilizado.

4.8.2. Diseño detallado

El Diagrama de clases, indica como se realiza la interconexión y paso de mensajes entre los módulos y clases del sistema, Figura 4.13.

4.8.3. Pruebas

Se describen a continuación las pruebas a realizar para esta etapa. Los resultados de las mismas se especifican en el siguiente Capítulo, con el fin de dar más detalle sobre los datos, valores obtenidos y valores esperados.

4.8.3.1. Pruebas de caja blanca

Estas pruebas, se enfocan principalmente en analizar el funcionamiento de la sección crítica del sistema integrado ahora con el resto del sistema.

4.8.3.2. Pruebas de caja negra

Contemplando el sistema en conjunto, estas pruebas consisten en ejecutar de manera continua introduciendo señales conocidas de las cuales se conocen sus carac-

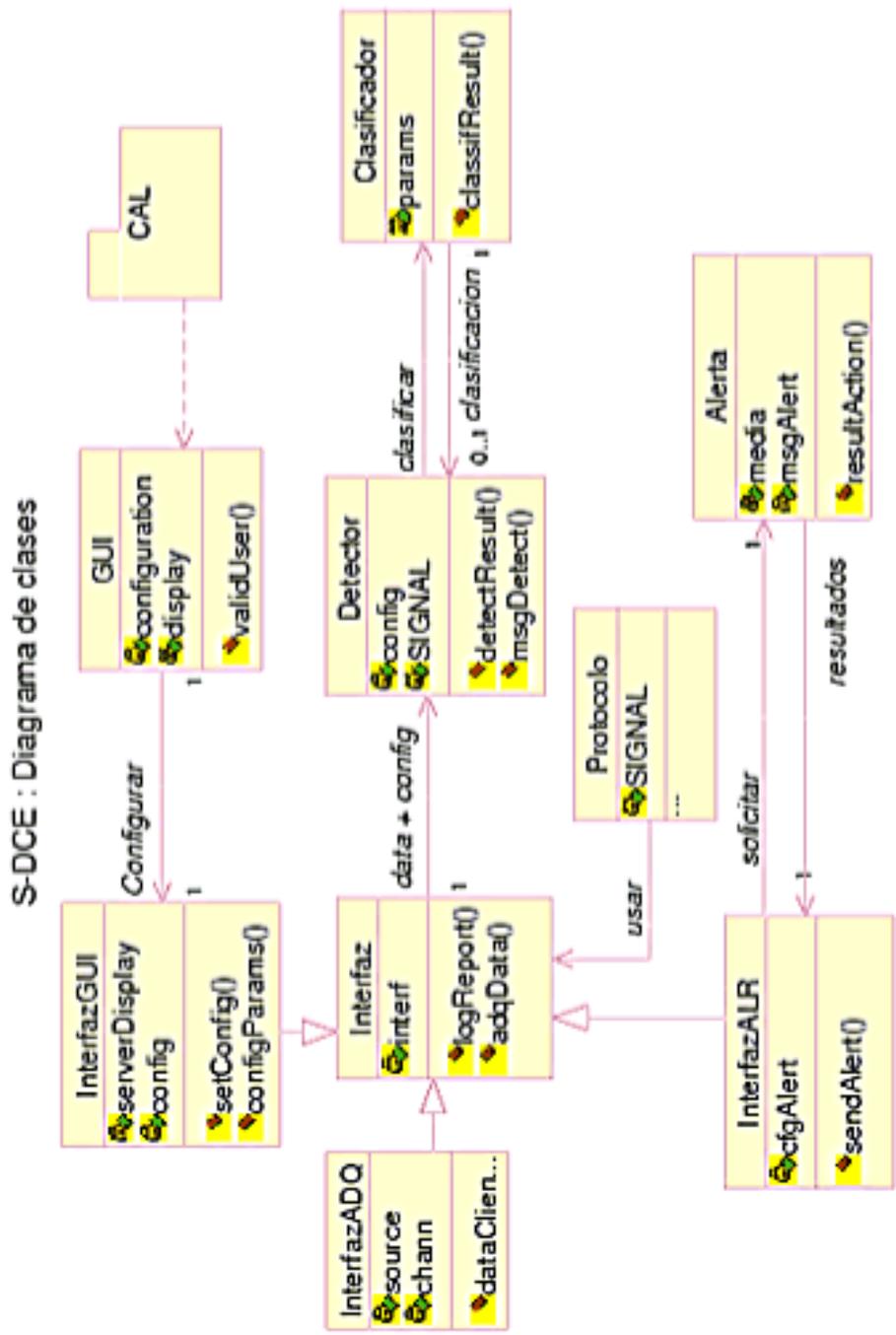


Figura 4.13: Diagrama de clases

terísticas, contrastando los resultados o reportes entregados por el sistema con la información conocida de las señales de entrada.