***Objetivo:***

*Describir los elementos físicos del sistema y sus relaciones, determinando los que son ejecutables, las bibliotecas, tablas, archivos y documentos, así como los diagramas de despliegue y disposiciones físicas en donde se instalará el sistema, a fin de modelar la topología de hardware, componente en donde se ejecutarán las funciones lógicas.*

*Las siguientes dos líneas deben eliminarse una vez que se requisite el artefacto.*

DISEÑO DE COMPONENTES

**ID de Requerimiento: <****363>**

*Ejemplo:*

*ID de Requerimiento: 100100 – 105001*

*Ejemplo:*

*ID de Requerimiento: 100100*

*Condiciones*

* *Si el documento no existe, debe ser elaborado, en caso contrario debe ser actualizado conforme lo requiera el proyecto*

*Criterios de aceptación*

* *El documento es utilizado de acuerdo al formato establecido.*
* *El documento no contiene faltas de ortografía.*
* *Se presentan los diagramas de clases por todos los casos de uso identificados y acordados con el cliente.*
* *Se elaboran los diagramas bajo el estándar MTR-TRM: Modelado/ Estándares de Modelado*
* *Se documentan todos los elementos de los diagramas de clases (clases, relaciones, interfaces etc.)*
* *Se mantiene consistencia en el o los diagramas con respeto a los nombres de los atributos y tipos de datos en sus relaciones.*
* *Se mantiene el estándar establecido para los elementos que conforman los diagramas de clases.*
* *Se agrupan los componentes por funcionalidad (subsistemas o marcos de trabajo).*
* *Los componentes tienen una clara definición de su contexto y necesidad. En caso de no tenerlo, se tiene una anotación que explique debidamente al componente y a su contexto.*

**[Fase: Desarrollo] [Etapa: Aplicativa]**

**Tabla de Versiones y Modificaciones**

*La siguiente tabla debe listar las versiones y descripciones hechas a este documento desde el momento de su creación y cada vez que se actualiza. La descripción debe incluir una síntesis del cambio hecho.*

*Para la Versión, especificar un número entero consecutivo,* ***iniciando con 1*** *para la* ***creación del documento*** *e incrementar en decimales conforme se actualice (1.1, 1.2, 1.3, Etc.). Al terminarlo, especificar la versión final para firma, en caso necesario.*

*Para futuros proyectos donde este documento se reutilice, el versionamiento reiniciará en versión 2, incrementando las actualizaciones igualmente con decimales (2.1, 2.2, 2.3, Etc.). Posteriores reutilizaciones iniciarían entonces con versión 3, 4, etc.*

| Versión | Descripción del cambio  *Síntesis de la modificación hecha al contenido del documento* | Responsable de la Versión  *Especificar nombre completo del responsable(s) de la versión del documento* | Fecha  *Especificar la fecha de la versión.*  *Formato: dd/mm/aaaa* |
| --- | --- | --- | --- |
| 11 | Creación del documento | Isaac Díaz Vargas | 13/03/2019 |
| *1.1* | *Arquitectura de la solución* |  |  |
| 1.1 | Versión aprobada para firma | Jorge Abel Bolaños Nila | 13/03/2019 |
| *1.3* | *Refinamiento de ACDMA.* |  |  |

**[Fase: Desarrollo] [Etapa: Aplicativa]**

**Tabla de Contenido**

[dIAGRAMA DE CLASES DE NEGOCIO 3](#_Toc3393146)

[**Caso de Uso DE IDEWEb** 3](#_Toc3393147)

[**Caso de Uso MONITOR** 6](#_Toc3393148)

[**Caso de Uso TRADICIONAL** 7](#_Toc3393149)

[**Diagrama de secuencia** 9](#_Toc3393151)

[**Aplicaciones basadas en otras Arquitecturas** 10](#_Toc3393152)

[**Detalle del Diseño** 10](#_Toc3393153)

[**Desarrollo** 11](#_Toc3393154)

[**Componentes de la Aplicación** 11](#_Toc3393155)

[**Subsistema/Marco de Trabajo DYP** 11](#_Toc3393156)

[**Diagramas de Servicios** 12](#_Toc3393157)

[**Definición de los Servicios** 12](#_Toc3393158)

[**Vista de Componentes** 12](#_Toc3393159)

[**Sincronización y Concurrencia** 12](#_Toc3393160)

[**Arquitectura Física** 13](#_Toc3393161)

[**Procesos de Servicio** 13](#_Toc3393162)

[**Especificación de Procesos** 14](#_Toc3393163)

[**Asignación de la Clase** 14](#_Toc3393164)

[**Colaboración de Clases** 14](#_Toc3393165)

[**Subprocesos de Servicio** 14](#_Toc3393166)

[**Diagrama de Despliegue** 14](#_Toc3393167)

[**Registros y Auditorias** 15](#_Toc3393168)

[**Manejo de Excepciones** 15](#_Toc3393169)

## dIAGRAMA DE CLASES DE NEGOCIO

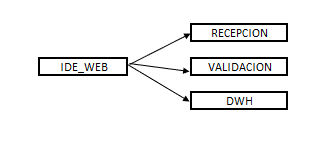
Las siguientes clases son arquitectónicamente las más significativas que representan el dominio del negocio para el aplicativo en descripción. El diagrama de clases ha sido dividido por los diferentes módulos del sistema.

### *Para evitar complejidad en el diseño, no se deben generar o mostrar los métodos getters y setters de las clases, se debe indicar claramente que dichos métodos se deben generar al iniciar la codificación. Las clases deben mostrar únicamente los métodos relevantes del diagrama, y especificar cuándo se considere importante alguna información adicional por medio de anotaciones (por ejemplo: “Las clases son instancias de la interfaz Y, por lo que cada una debe implementar los métodos A, B y C”). Se considerará como definición que un método relevante es aquél que sea definido como público***Caso de Uso DE IDEWEb**

* **28\_382\_ECU\_IDEWEB**
* **28\_382\_ECU\_IDEWEB\_Modulo**

Diagrama de clases

* Las siguientes clases son arquitectónicamente las más significativas que representan el dominio del Negocio para la creación de las estructuras de datos para las Declaraciones del IDEWEB.



*Incluir el diagrama de clases aquí*

Descripción de elementos del diagrama de clases

El módulo de IDE\_WEB\_MODULO consta de 3 clases: Recepcion, Validacion y DWH.

*Describir todas y cada una de las clases que forman parte del diagrama.*

##### Clase Recepcion

La clase Recepcion se encarga de capturar el RFC del usuario, su contraseña, certificado y firma y validar los permisos con las aplicaciones a usar.

*Describir la finalidad y responsabilidad de la clase.*

Atributos

* Contraseña
* RFC
* Certificado
* Firma

*Describir todos y cada uno de los atributos que conforman la clase.*

Métodos

* Mautentificacion
* Menvio

*Describir todos y cada uno de los métodos.*

GUÍA

|  |  |  |
| --- | --- | --- |
| RECEPCIÓN |  | VALIDACIÓN |
| Contraseña |  | Archivo\_Decla |
| RFC |  | Archivo\_Acuse |
| Certificado |  |  |
| Firma |  | Mcomparar |
| Archivo |  | Menvioacuse |
|  |  |  |
| Mautenticacion |  |  |
| Menvio |  |  |



##### Clase Validacion

La clase Validacion se encarga de validar la información capturada por el usuario.

*Describir la finalidad y responsabilidad de la clase.*

Atributos

* Archivo\_decla
* Archivo\_Acuse

*Describir todos y cada uno de los atributos que conforman la clase.*

Métodos

* Mautentificacion
* Menvio

*Describir todos y cada uno de los métodos.*

GUÍA

*Si se incluye un atributo de nombre ID Cliente de tipo String, este nombre y tipo de dato debe ser consistente en todas las clases que requieran guardar una referencia del atributo ID Cliente.*

|  |  |  |
| --- | --- | --- |
| VALIDACIÓN |  | DHW |
| Archivo\_Decla |  | Archivo\_Decla |
| Archivo\_Acuse |  |  |
|  |  | Malimentabd |
| Mcomparar |  |  |
| Menvioacuse |  |  |

##### Clase DWH

La clase DWH se encarga de almacenar en base de datos y alimentar a data where hause.

*Describir la finalidad y responsabilidad de la clase.*

Atributos

* Archivo\_decla

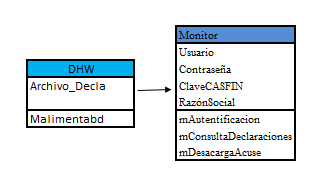
*Describir todos y cada uno de los atributos que conforman la clase.*

Métodos

* Malimentabd

*Describir todos y cada uno de los métodos.*

GUÍA



### **Caso de Uso MONITOR**

**28\_382\_ECU\_IDEWEB\_MONITOR\_BITACORA,**

**28\_382\_ECU\_IDEWEB\_MONITOR\_DECLARACIONES,**

**28\_382\_ECU\_IDEWEB\_MONITOR\_ESTADISTICA,**

Diagrama de clases

MONITOR

Descripción de elementos del diagrama de clases

Se representa monitor para consulta de las declaraciones presentadas en el IDE WEB

##### Clase Monitor

Muestra la información de la consulta realizada.

Atributos

* Estatus
* Folio
* Nombre de Archivo
* RFC
* Banco
* Tipo de Archivo
* Medio de Recepción
* Ultimo Estado
* Sector
* Formato
* Motivo de Rechazo
* Razón Social
* Periodo
* Ejercicio
* Operaciones
* Forma de la Declaración
* Periodo de la declaración
* Fecha presentación

Métodos

* mConsultaDeclaraciones
* mDescargaAcuse

GUÍA

|  |
| --- |
| Monitor |
| Estatus  Folio  Nombre de Archivo  RFC  Banco  Tipo de Archivo  Medio de Recepción  Ultimo Estado  Sector  Formato  Motivo de Rechazo  Razón Social  Periodo  Ejercicio  Operaciones  Forma de la Declaración  Periodo de la declaración  Fecha presentación |
| mConsultaDeclaraciones  mDescargaAcuse |

### **Caso de Uso TRADICIONAL**

**28\_382\_ECU\_IDETRADICIONAL\_DESCARGA**

Diagrama de clases

#### 

DescargaAcuses

MonitorTradicional

Descripción de elementos del diagrama de clases

Se representan 2 portales para consulta de las declaraciones presentadas en el IDE Tradicional.

##### Clase MonitorTradicional

Muestra la información de la consulta realizada.

Atributos

* Usuario
* Contraseña
* Folio
* Razón Social
* RFC
* Sector
* Tipo de Archivo
* Medio Recepción
* Fecha Recepción
* Último Estado
* Periodo
* Fecha Presentación
* Ejercicio
* Operaciones
* Procesada Manualmente

Métodos

* mAutentificacion
* mConsultaDeclaraciones

GUÍA

|  |
| --- |
| Monitor |
| Usuario  Contraseña  Folio  Razón Social  RFC  Sector  Tipo de Archivo  Medio Recepción  Fecha Recepción  Último Estado  Periodo  Fecha Presentación  Ejercicio  Operaciones  Procesada Manualmente |
| mAutentificacion  mConsultaDeclaraciones |

Clase DescargaAcuses

Muestra la información de la consulta realizada. Para descargar el acuse requerido.

Atributos

* Usuario
* Contraseña
* ClaveCASFIN
* RazónSocial

Métodos

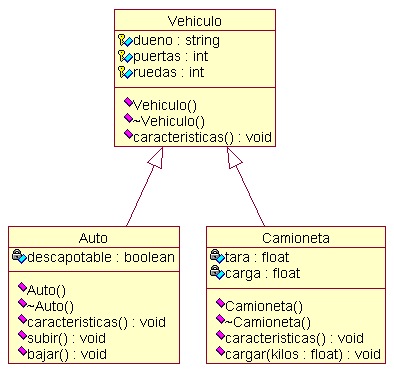
* mAutentificacion
* mConsultaDeclaraciones
* mDesacargaAcuse

GUÍA

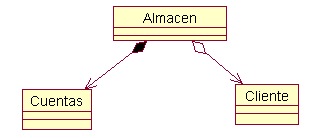
|  |
| --- |
| Monitor |
| Usuario  Contraseña  ClaveCASFIN  RazónSocial |
| mAutentificacion  mConsultaDeclaraciones  mDesacargaAcuse |

***Ejemplo de Tipos de relaciones en el diagrama de clases****.*

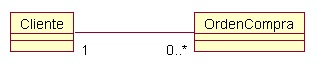
*Generalización/especialización*



*Agregación/Composición*



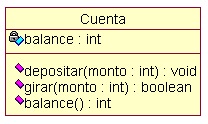
*Asociación/Cardinalidad*



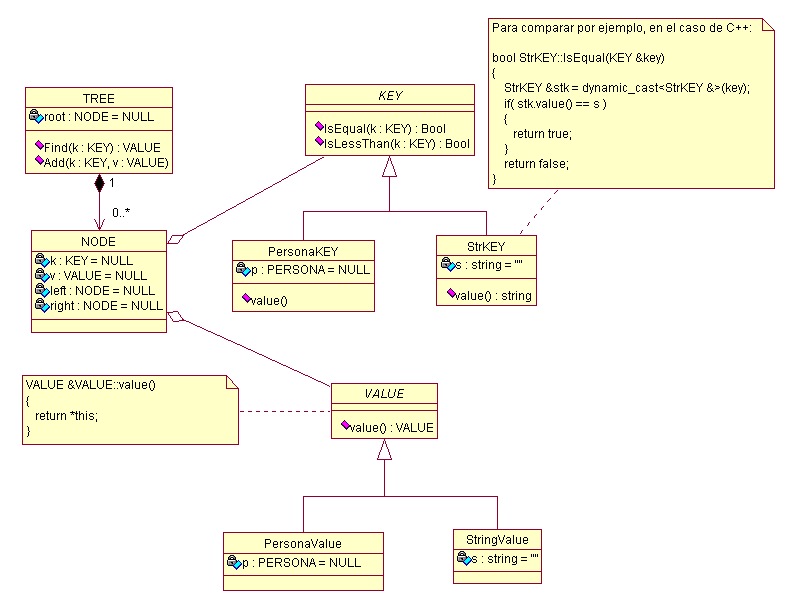
*Dependencia o instanciación (uso)*

http://www.dcc.uchile.cl/~psalinas/uml/img/modelo/edependencia.jpg

*Ejemplo de clase*



*Ejemplo integrado con todos los elementos expuestos*



### **Diagrama de secuencia**

Presentar declaración.



Monitor IDE WEB



IDE Tradicional

Monitor



### **Aplicaciones basadas en otras Arquitecturas *(Obligatorio para aplicaciones basadas en Cliente Servidor) (Obligatorio) [DESARROLLO CDS]***

Este aplicativo está basado en el Sistema Declaraciones Informativas Múltiples Electrónicas y Nuevo Esquema de Pagos Electrónicos, pero solo en forma, porque se estructuró una nueva estructura mejorada y adaptada a las propias necesidades.

### **Detalle del Diseño**

Especificar el Diseño por requerimiento de acuerdo a la tabla trazabilidad de requerimientos.

En esta sección se deberán plasmar cada uno de los diseños que se impactan derivado del requerimiento por lo que se tomará en cuenta cada uno de los siguientes tipos (Diseño Aplicativo, Diseño de Seguridad, Diseño de Información, Diseño Tecnológico)

Detallar el diseño de los objetos (Pantalla, Reporte, Layout, Tabla, Campo, etc.) a modificar en cada uno de los por cada uno de los requerimientos

Proporcionar los detalles de los objetos comprados, reutilizados o desarrollados para ésta solicitud de cambios

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objeto | Descripción del Objeto | Número de versión del Objeto | Hacer / Comprar/ Reutilizar | Fuente |
| Pantalla, Programa, Job Control Language (JCL), Página, Reporte, Clase, Otros | Describir brevemente la principal función del objeto. | Indicar el número de versión del componente a modificar. | Indicar si el objeto se va a desarrollar (hacer), comprar, o modificar (Reutilizar). | Indicar herramienta y ruta de donde se obtiene el objeto a modificar. |
| Modulo Recepción | Recibe la información del usuario |  | Hacer |  |
| Modulo Validación | Valida la información del usuario |  | Hacer |  |
| Modulo DWH | Almacena la información del usuario |  | Hacer |  |
| Modulo Monitor | Visualiza la información del proceso |  | Hacer |  |
| Modulo Tradicional | Utiliza el proceso tradicional |  | Hacer |  |

### **Desarrollo**

Desarrolle los componentes de la aplicación de acuerdo a las especificaciones de diseño

### **Componentes de la Aplicación**



|  |  |  |  |
| --- | --- | --- | --- |
| Requerimiento # | Nombre del Objeto | Tipo de Objeto | Agregar/Cambiar/Borrar |
| 1 | Modulo Recepción | Servicio WEB | Agregar |
| 2 | Modulo Validación | Proceso | Agregar |
| 3 | Modulo DWH | Data Base | Agregar |
| 4 | Modulo Monitor | Servicio WEB | Agregar |
| 5 | Modulo Tradicional | Proceso | Agregar |

*Este diagrama muestra las operaciones (procesos) de los servicios descritos por pasos que pueden manejarse por el servicio en la secuencia en la que es ejecutado.*

*Desarrolle los componentes de la aplicación de acuerdo a las especificaciones de diseño*

### **Subsistema/Marco de Trabajo DYP**

* Recepción de archivos de declaraciones.

Los contribuyentes pueden enviar sus declaraciones al SAT mediante la página WEB, también pueden hacerlo mediante la presentación del archivo en la ADSC a la que pertenecen.

### **Diagramas de Servicios**

Este aplicativo no se conecta con servicios externos.

### *De existir servicios que se consumen de otro sistema, se deberán detallar en las siguientes secciones de acuerdo aplique.*

### **Definición de los Servicios**

Ver la sección “Procesos de servicios”.

*Esta vista, describe el mapeo entre el servicio de aplicación y los correspondientes servicios de negocio. Utilice las guías principales para descomponer los servicios de negocio en uno o más servicios de aplicación. En esta sección, proporcionar un mapeo del servicio de aplicación a los servicios de negocio de los que fueron creados.*

### **Vista de Componentes**

**File Share**

Clúster de Discos donde se almacena físicamente la declaración del contribuyente

**Microsoft .NET 2018**

Herramienta para el desarrollo o modificación del proyecto.

**Microsoft .NET Framework 3.5**

Componente necesario para el funcionamiento de los servicios.

**Servicios de Envío**

Repositorio SAT. Corresponde al medio de envío Socket de Seguridad que utilizan las instituciones financieras para la transferencia de archivos.

**Internet information services 6.0**

Componente necesario para el alojar páginas de internet.

**Database**

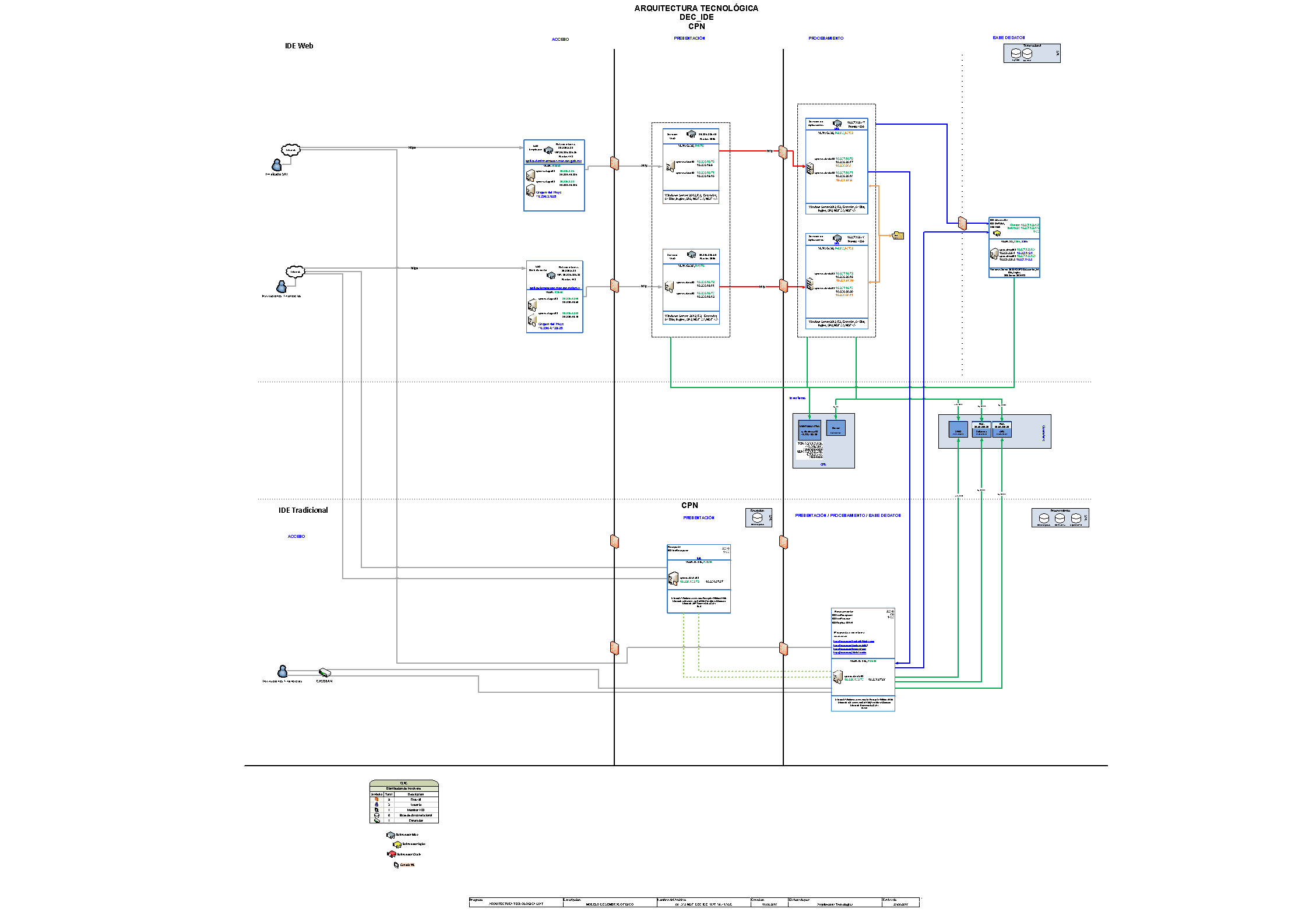
Clúster de Datos donde se encuentran hospedada la base de datos del aplicativo.

*Esta vista describe los componentes que necesarios para desarrollar y como se estructuran. Esto es esencial para el desarrollo de la solución.*

### **Sincronización y Concurrencia**

*Discutir los aspectos de concurrencia de la solución mediante la descripción de los diversos procesos del sistema operativo y los hilos, sus dependencias, los aspectos de administración, sus interacciones usando diagramas de tiempo. En esta sección se deben abordar cuestiones como la arquitectura de control de concurrencia y estrategias para el bloqueo de la base de datos (optimista, pesimista), etc.*

### **Arquitectura Física**



*Definir los principales componentes físicos que establecen la arquitectura lógica.*

### **Procesos de Servicio**

*Esta sección cubre los procesos ejecutados por el servicio web para realizar sus diversas funciones.*

* IDEWEB Recepción
* IDEWEB Envío Acuses
* Transferencia Recepción
* Transferencia Acuses
* IDEWEB Registro
* IDEWEB Validación
* IDEWEB Procesamiento DWH
* MONITOR

### **Especificación de Procesos**

*Esta sección contiene una descripción del proceso, en forma estructurada (pseudo código). Incluir un diagrama del flujo del procedimiento si eso le ayuda a describir el proceso.*

**Solicitud de Datos**

No aplica, no se solicitan datos a servicios externos.

*Solicitud de Datos para servicios externos:*

| **Nombre del Servicio** | **Operación** | **Esquema solicitado** | **Esquema de Respuesta** |
| --- | --- | --- | --- |
| *<Ejemplo: Recuperar preferencia del servicio>* | *<Ejemplo: getUserPreferences)>* | *<Ejemplo: OTA\_AirPriceRS>* | *<Ejemplo: OTA\_AirPriceRO>* |
| No aplica. | No aplica. | No aplica. | No aplica. |

### **Asignación de la Clase**

No aplica.

*Esta sección describe la forma en que el proceso será desplegado en la clase que le ha sido asignado. Incluye funciones prototipos de lenguaje independiente (nombre de función, argumentos y valor de retorno). También incluyen esquemas XML, en su uso para los parámetros.*

### **Colaboración de Clases**

No aplica, no se utilizan clases ni objetos externos.

*Esta sección identifica clases/objetos externos y sus componentes que serán usados por la función.*

### **Subprocesos de Servicio**

*Esta sección cubre los procesos ejecutados por el web service para realizar varias funciones. La aproximación define como el proceso se construye, 3GL, Web Service, procesos BPEL, rule checks o la orquestación de procesos.*

**Lista de Procesos:**

| **Procesos** | **Descripción** | **Acercamiento** |
| --- | --- | --- |
| <Ejemplo: Obtener el detalle del boleto> | <Ejemplo: Obtener el detalle del boleto de SHARES usando el número que fue pasado a este proceso. Regresará todos los boletos asociados con el boleto de interés así como todos los boletos asociados con la solicitud.> | <Ejemplo:  Procesos Workflow – invoca al SHARES Direct service.  Mapeando los procesos para extraer el número de boleto en la invocación del servicio > |
| No aplica | No aplica | No aplica |

*Se debe definir claramente las interfaces de entrada y de salida del componente (su conexión con el medio), e internamente se deben definir los elementos que lo conforman. Estos elementos pueden ser componentes o clases. Es importante que, de no ser lo bastante claro el diagrama, se debe facilitar su comprensión por medio de anotaciones.*

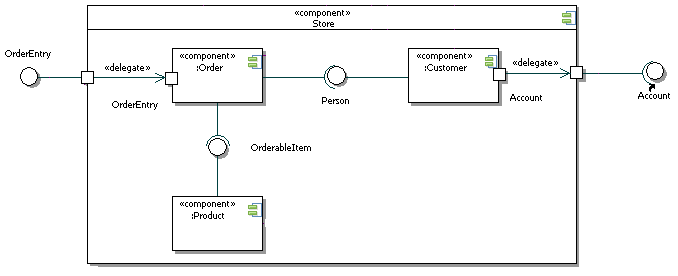
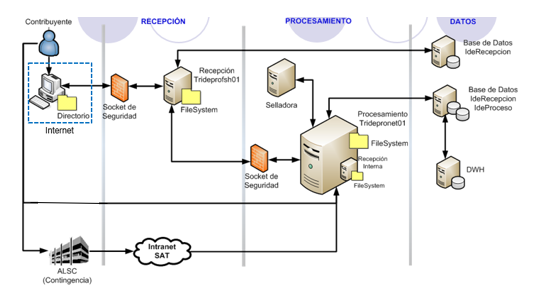


Figura 1: Diagrama de componentes XXX

### **Diagrama de Despliegue** (Obligatorio)



*Este diagrama contiene las dependencias entre los componentes que se subirán al servidor de aplicaciones. Es importante poner referencias y direccionalidad entre los componentes, para conocer las diferentes dependencias entre ellos. De existir una librería o servicio externo, deberá mencionarse a nivel interno, por medio de su WSDL (o API, en caso de ser componentes) correspondiente.*

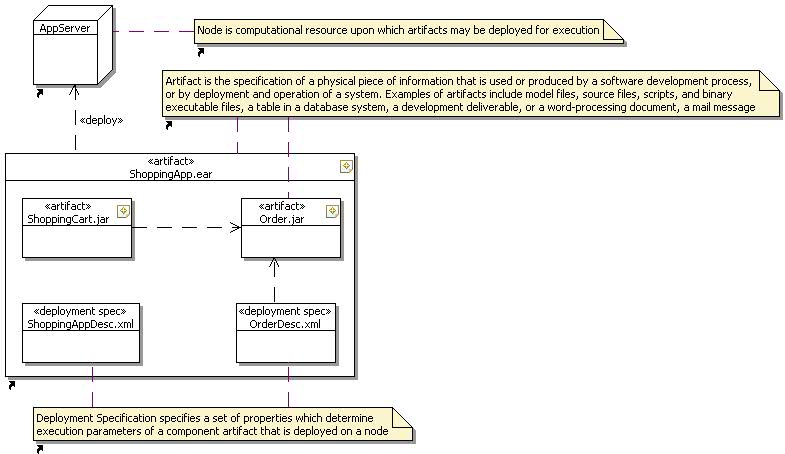


Figura 2: Diagrama de despliegue XXX

### **Registros y Auditorias** (Obligatorio)

No Aplica *La siguiente tabla identifica los elementos que son usados para capturar la información de registro y auditorias. Para cada servicio se requiere que se definan los eventos para los cuales se necesita realizar un registro u operación de auditoria. La información del encabezado es común a ambos Registro y Mensajes de excepción.*

**Tabla de Registro**

| **Elemento** | **Descripción** |
| --- | --- |
| No Aplica | No Aplica |
| Header | Registro y Segmento de Excepción del encabezado |
| TimeDuration | Longitud de tiempo |
| Category | Una de las categorías definidas para la aplicación |
| Messages | Consiste de nombre repetido y un valor par |
| Name | Arreglo opcional de nombres de registros generados por servicio |
| Value | Arreglo opcional de valores de registro generados por servicio |
| Status | Campo para distinguir los diferentes tipos de registro (es decir, En progreso, enviado) |
| TransactionBefore | Dato del inicio de la ejecución |
| TransactionAfter | Dato del fin de la ejecución |
| DataEncoding | STRING, BINARIO, HEX |
| LogLevel | DEBUG / INFO / ERROR |

**Tabla de encabezados de bitácoras y excepciones**

| **Elemento** | **Descripción** |
| --- | --- |
| No Aplica | No Aplica |
| ServiceID | Este id debe ser un número único asignado a cada nuevo servicio |
| ServiceName | Este campo debe ser la descripción del texto del id de la aplicación. (opcional) |
| ComponentName | El nombre del componente dentro de la aplicación con el cual se lanza la excepción |
| HostName | El host sobre el cual el servicio fue ejecutado |
| Timestamp | Día y Hora en los cuales la excepción ocurrió |
| TransactionDomain | Diferenciador de diferentes tipos de excepción. (opcional) |
| TransactionType | Tipo de transacción (INFO / WARNING / EXCEPTION / NOTIFICATION) |
| TransactionID | Campo que únicamente identifica este registro |
| Topic Name | Identificador de clasificación adicional (por que se suscribió) |

**Tabla de Registro de Eventos**

| **Id del Evento** | **Nombre del Servicio Origen** | **Descripción del disparador del evento** | **Nombre del Evento** | **Acciones de Notificación** | **Descripción** |
| --- | --- | --- | --- | --- | --- |
| ID único | servName1 | Excepción del SLA | Campo de Texto | Notificación del Sistema | Descripción del evento |
| No Aplica | No Aplica | No Aplica | No Aplica | No Aplica | No Aplica |

### **Manejo de Excepciones**

*Los detalles del manejo de excepciones deben colocarse en esta sección. Los siguientes formatos deben realizarse para ayudar a documentar las excepciones:*No Aplica

* *Flujo detallado de eventos – Manejo de Errores*
* *Flujo detallado de eventos – Entrantes*
* *Flujo detallado de eventos – Salientes*
* *Registros de auditoria*
* *Alertas y notificaciones*

*La tabla que se muestra a continuación identifica los elementos que son usados para capturar excepciones de la información. Cada servicio es requerido para definir los errores relacionados con su proceso de servicios.*

**Tabla de Excepciones**

| **Elementos** | **Descripción** |
| --- | --- |
| No Aplica | No Aplica |
| Header | Registro y segmento de Excepciones del encabezado |
| Category | Una de las categorías definidas para la aplicación |
| Severity | Una de las severidades definidas para la aplicación |
| Code | Algunos códigos específicos usados para los reportes y las búsquedas |
| Message | Mensajes específicos de la aplicación |
| ReplyDestination | Direcciones de Servicio (URL) las cuales manejan la excepción que pudieran resolverse con los mensajes |
| TransactionData | Datos de entrada para el proceso los cuales han sido lanzados por la excepción |
| DumpAnalysis | Datos extras que ayudan a resolver la excepción |
| DataEncoding | STRING, BINARIO, HEX |
| LogLevel | DEBUG / INFO / ERROR / WARNING |

*Las excepciones de los servicios pueden clasificarse en las siguientes categorías: Sistema, Aplicación y negocio. SOA provee un elemento especializado (SOAP Fault) para capturar la excepción. Este se encuentra mejor catalogado para reportar las excepciones a nivel de sistema. Las excepciones a nivel de sistema son generadas por un reporte de fallas del sistema de software o hardware. Las aplicaciones y las excepciones a nivel de negocio requieren significativamente más datos para definir apropiadamente la causa del error. Este tipo de excepciones requieren esquemas de información especiales para cada error específico. L atabla siguiente identifica los elementos que son usados para capturar la excepción de la información. Cada servicio requiere definir errores relacionados con sus procesos de servicio.*

| **Elementos** | **Descripción** |
| --- | --- |
| No Aplica | No Aplica |
| Header | Registro y segmento de excepción del encabezado |
| Category | Una de las categorías definidas por la aplicación |
| Severity | Una de las severidades definidas por la aplicación (WARNING / FATAL / INFORMATION) |
| Code | Algunos códigos específicos de la aplicación usados para reportar y hacer búsquedas |
| Message | Mensajes específicos de la aplicación |
| ReplyDestination | Direcciones de Servicio (URL) las cuales manejan la excepción que pudieran resolverse con los mensajes |
| TransactionData | Datos de entrada del proceso los cuales fueron lanzados por la excepción |
| DumpAnalysis | Datos extras que ayudan a resolver la excepción |
| DataEncoding | STRING, BINARIO, HEX |
| LogLevel | DEBUG / INFO / ERROR / WARNING / FATAL |