



Benemérita Universidad Autónoma de Puebla

Facultad de Ciencias de la
Computación

“Aplicación web para el diseño de procesos de
negocio”

Tesina
Para optar el grado de :

**Licenciado en Ingeniería en Tecnologías de la
Información**

Presenta
Jesus Garcia Licona



Facultad de Ciencias
de la Computación

Asesor de Tesina: Juan Manuel González Calleros

Agosto 2018

1	Introducción	3
2	Estado del arte	5
3	Análisis.....	13
4	Diseño.....	14
4.1	Casos de uso.....	14
4.2	Especificación de casos de uso.....	15
4.2.1	Importar modelo XML	15
4.2.2	Editar modelo de procesos	16
4.2.3	Exportar modelo	17
4.2.4	Interpretar XML	18
4.2.5	Mostrar modelo	19
4.3	Diagrama de clases.....	20
4.4	Diagramas de secuencia.....	21
4.4.1	Diagrama de secuencia Importar XML	21
4.4.2	Diagrama de secuencia Editar modelo	22
4.4.3	Diagrama de secuencia Exportar modelo.....	23
4.4.4	Diagrama de secuencia Interpretar XML.....	24
5	Desarrollo	25
5.1	Selección de herramienta BPMN	25
5.2	Estructura XML.....	26
5.3	Implementación caso de uso Importar XML.....	27
6	Pruebas	28
6.1	Toma de requerimientos.....	28
6.2	Elaboración del XML	28
6.3	Intérprete aplicación web.....	31
6.4	Editar modelo BPMN	32
6.5	Generar Sistema.....	33
7	Conclusiones.....	34
8	Bibliografía.....	35

1 Introducción

En décadas recientes las empresas han buscado cómo mejorar sus procesos para ahorrar recursos y maximizar ganancias, es por ende que se ha vuelto cada vez más impredecible tener una organización y una estructura en la realización de cada tarea, viendo como interactúa con las demás áreas de la empresa.

Al ser hoy en día tareas cada vez más complejas e interacción con diferentes actores, se tiene la necesidad de crear modelos de procesos de negocio, para tener un mayor control de producción y reglas que se siguen para brindar un mejor producto o servicio. En el área de computación específicamente en la toma de requerimientos de un nuevo software aun falta herramientas que puedan plasmar modelos de procesos que apoyen a visualizar las tareas de un nuevo software.

Todas las herramientas de modelado de procesos de negocios no usan de forma automática conocimiento recolectado en etapas tempranas de ingeniería de requerimientos. Por lo que un analista de procesos, debe elaborar los diagramas a partir de información plasmada de manera informal o formal resultado de la ingeniería de requerimientos. En consecuencia, mucho trabajo previamente realizado solo sirve de forma teórica pero no podemos rápidamente incorporar sus resultados en el diseño de la solución informática.

Se ve un área de oportunidad en esta área, ya que se puede integrar varias herramientas de desarrollo ya existentes para desarrollar una nueva forma de creación de software, es decir si un nuevo cliente le pide al desarrollador un nuevo sistema, en la toma de requerimientos se realiza procesamiento de lenguaje natural se puede generar con palabras clave un archivo XML con las características principales del software, utilizando un interprete importara el archivo XML y lo plasmara en un modelo de procesos de negocio (BPMN), dando la facultad al desarrollador de detallar cada tarea, consiguiente se exportara el archivo XML para que en su etapa final se desarrolle un sistema a base del XML generado del modelo de procesos de negocio.

El objetivo principal de este proyecto plantea desarrollar es una aplicación web enfocado al Modelado de Procesos de Negocio (BPM), en donde a partir de un XML se pueda procesar y por siguiente graficar el modelo de proceso de negocio, con la opción de editarlo y exportarlo a un archivo XML o guardarlo en imagen. El origen del XML son datos recolectados de la ingeniería de requerimientos a través de técnicas de procesamiento del lenguaje natural se representa conocimiento en formato XML, esto último queda fuera del alcance del proyecto.

Los objetivos específicos son:

- Seleccionar la herramienta de software abierto que permita la importación de modelos a partir de un archivo XML y que genere diagramas BPMN.
- Modelar la estructura de un proceso de negocios en formato de meta-clase para poder generar un XML que sirva de entrada al editor arriba seleccionado.
- Generar las reglas de transformación que permitan transferir conocimiento de ingeniería de requerimientos al meta-modelo de procesos de negocio en el formato XML.
- Desarrollar un intérprete de los modelos de proceso de negocio en la aplicación web, usando el formato XML consistente con nuestro meta-modelo, para crear diagramas de procesos de negocios.

En este documento mostrará el desarrollo de este proyecto siguiendo la metodología RUP con notación UML, en donde se verán los retos y soluciones que se utilizaron para implementar la aplicación web.

2 Estado del arte

Relacionado con el modelado de procesos de negocio podemos encontrar en la literatura y la industria algunas notaciones que nos permiten modelar procesos. A continuación describimos algunas de las más usadas. En [4] encontramos una colección y análisis del contexto de uso de procesos de negocio, en la que se menciona se han propuesto varias notaciones de modelado, que puede ser: Petri Nets [5] como lenguaje de modelado, representa gráficamente la estructura de un sistema distribuido como un gráfico bipartito dirigido con anotaciones. Son una técnica para modelar y analizar procesos.

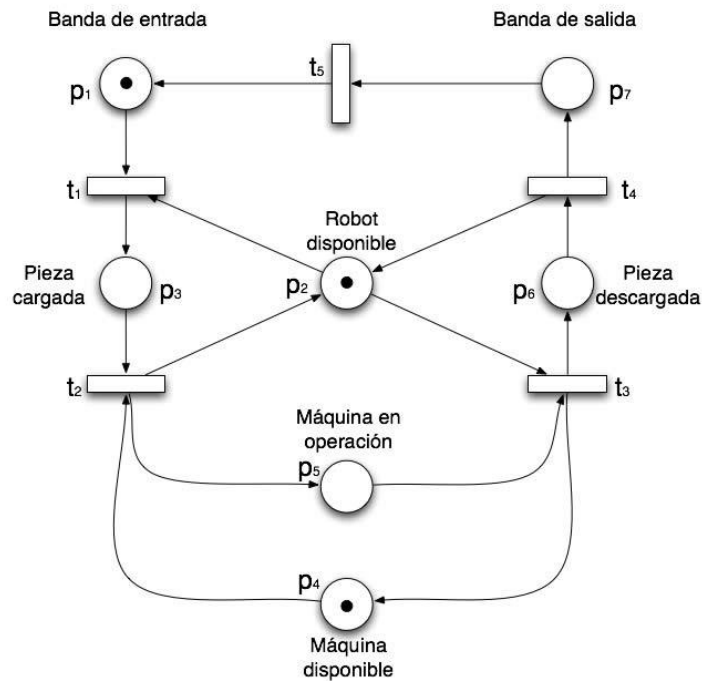


Figura 1: Aplicación de redes de petri en la modelación de eventos discretos

Una red de Petri está formada por lugares, transiciones, arcos dirigidos y marcas o fichas que ocupan posiciones dentro de los lugares. Las reglas son: Los arcos conectan un lugar a una transición así como una transición a un lugar. No puede haber arcos entre lugares ni entre transiciones. Los lugares contienen un número finito o infinito contable de marcas. Las transiciones se disparan, es decir consumen marcas de una posición de inicio y producen marcas en una posición de llegada. Una transición está habilitada si tiene marcas en todas sus posiciones de entrada.

Diagramas de estado [6][7] que son gráficos que representan máquinas de estados que describe la respuesta, de un objeto de una determinada clase, a la recepción de estímulos externos.

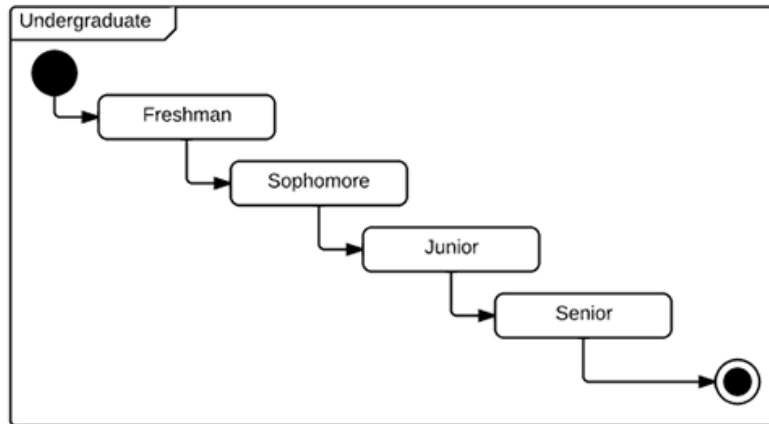


Figura 2: Ejemplo de diagrama de maquinas de estados en UML

Los diagramas de estado muestran el conjunto de estados por los cuales pasa un objeto durante su vida en una aplicación en respuesta a eventos (por ejemplo, mensajes recibidos, tiempo rebasado o errores), junto con sus respuestas y acciones. También ilustran qué eventos pueden cambiar el estado de los objetos de la clase. Normalmente contienen: estados y transiciones. Como los estados y las transiciones incluyen, a su vez, eventos, acciones y actividades, vamos a ver primero sus definiciones.

Al igual que otros diagramas, en los diagramas de estado pueden aparecer notas explicativas y restricciones.

Los principales elementos que representan los diagramas de estado son los estados y las transiciones. Los estados se captan por medio de rectángulos redondeados que se etiquetan con el nombre del estado. Las transiciones se marcan con flechas que fluyen de un estado a otro, mostrando cómo cambian los estados.

Cada diagrama de estado generalmente empieza con un círculo oscuro que indica el estado inicial y termina con un círculo con un contorno blanco que denota el estado final. Sin embargo, a pesar de tener puntos de inicio y finalización definidos, se debe recordar que los diagramas de estado no necesariamente son la mejor herramienta para plasmar un desarrollo general de eventos. En lugar de ello, se especializan en ilustrar tipos específicos de comportamiento —en particular, cambios de un estado a otro.

Notación de Modelado de Procesos Empresariales (BPMN) [8] es una notación gráfica estandarizada para dibujar procesos de negocios en un flujo de trabajo. Proporciona un medio simple de comunicar información de proceso a los otros usuarios comerciales, implementadores de procesos, clientes y proveedores.

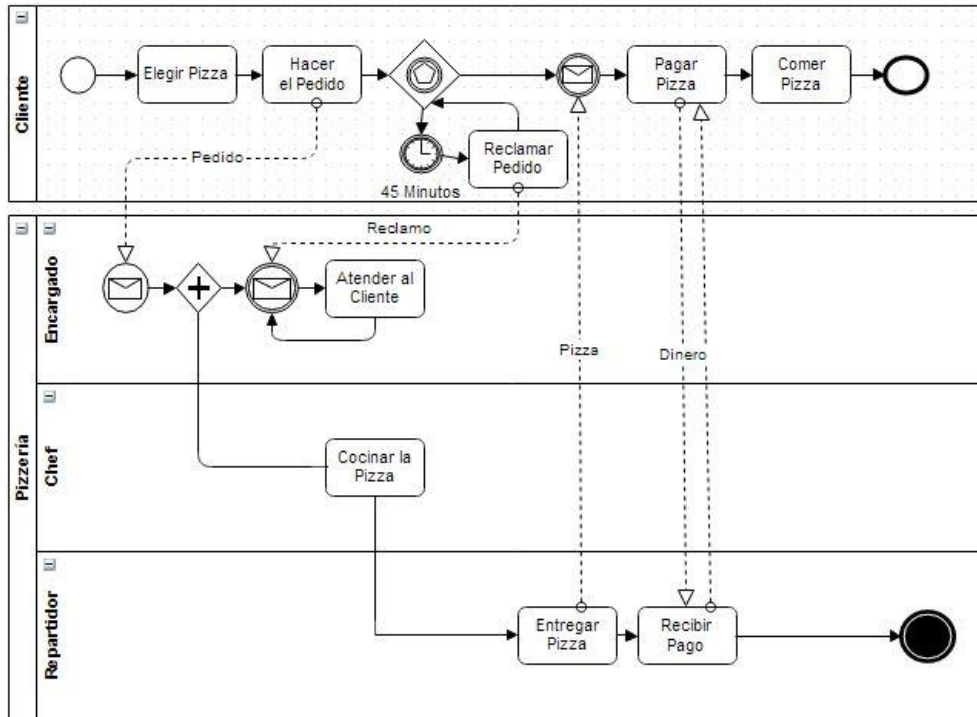


Figura 3: Modelo de procesos de negocio de pedido de una pizza

BPMN (Business Process Modeling Notation)

- Es el nuevo estándar para el modelado de procesos de negocio y servicios web.
- Es una notación a través de la cual se expresan los procesos de negocio en un diagrama de procesos de negocio (BPD)

Este estándar agrupa la planificación y gestión del flujo de trabajo, así como el modelado y la arquitectura.

Diagramas de actividad UML [9]. Un diagrama de actividad proporciona una vista del comportamiento de un sistema al describir la secuencia de acciones en un proceso. Los diagramas de actividad UML están destinados a modelar procesos computacionales y de organización.

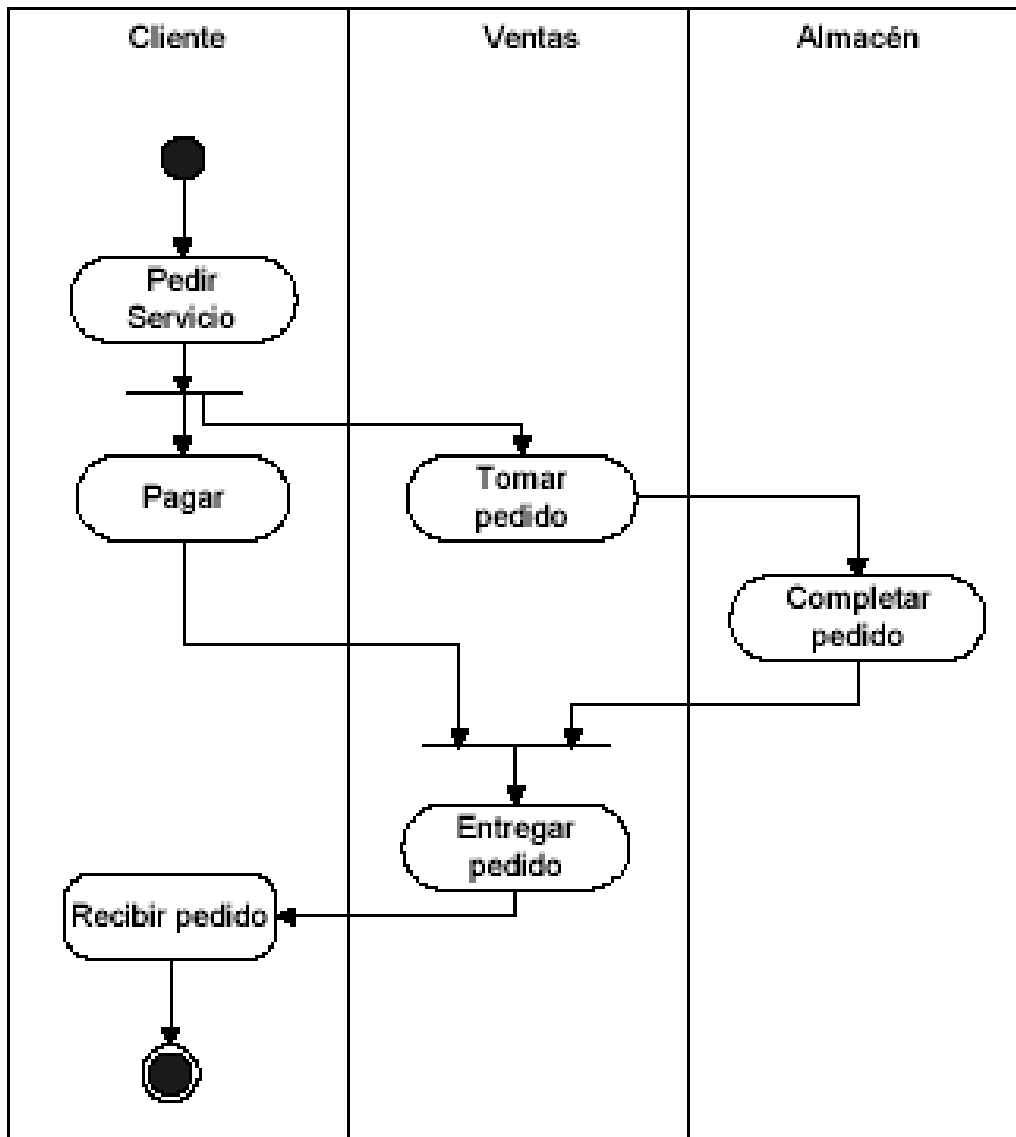


Figura 4: Diagrama de actividad de pedidos

En la actualidad, varios modelos y métodos de diseño respaldan el desarrollo de aplicaciones complejas basadas en flujo de trabajo que proporcionan notaciones para procesos comerciales e incluyen el soporte de herramientas. Algunos de ellos son:

El Modelo de progresión [10] ha incorporado algunos de los conceptos de gestión del flujo de trabajo para aumentar la flexibilidad en IS. Hace explícitamente los pasos y las transacciones según el usuario se compromete al usar un IS. A medida que el usuario avanza hacia el logro de una tarea u objetivo, la infraestructura del modelo de progresión registra cada paso y el estado de la transacción y el flujo de trabajo.

Microsoft Windows Workflow Foundation (WWF) [11] es un marco extensible para desarrollar soluciones de flujo de trabajo en la plataforma de Windows. Proporciona un modelo único y unificado para crear soluciones de extremo a extremo que abarcan categorías de aplicaciones, incluido el flujo de trabajo humano y el flujo de trabajo del sistema.

WebSphere MQ Workflow (IBM) [12] es compatible con el flujo de trabajo de procesos de negocios de larga duración a medida que interactúan con sistemas y personas. Automatiza y realiza un seguimiento del proceso empresarial de acuerdo con el diseño comercial. Proporciona procesos de integración con gran soporte para las interacciones humanas.

Flujo de trabajo en entorno de base de datos inteligente y distribuido (WIDE) [13] define un modelo conceptual avanzado para describir tanto el flujo de actividades como el entorno organizacional en el que se realizan estas actividades.

Business Process Visual ARCHITECT (BP-VA) [14] es una herramienta de modelado visual que proporciona el soporte más extenso para BPMN.

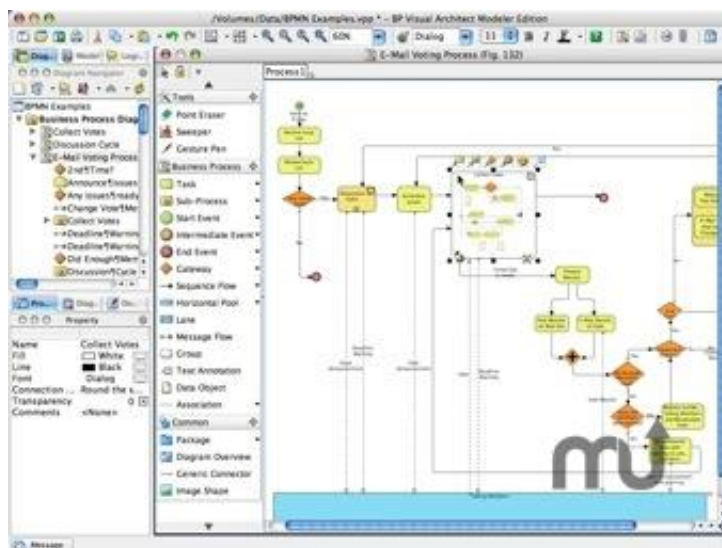


Figura 5: Entorno de desarrollo Business Process Visual ARCHITECT

Un segundo aspecto de este trabajo es la integración de los procesos de negocio a un marco de desarrollo de software más amplio donde la generación de modelos pueda ser resultado de un proceso de transformación que parte de la ingeniería de requerimientos y después pudiera generar de forma automática el sistema interactivo necesario para realizar el trabajo denotado en el modelo de proceso de negocio. Para esto nos apoyaremos del marco de referencia UsiXML [1], que es una metodología para generar UI en un enfoque basado en modelos. Describe la interfaz de usuario para múltiples contextos de uso, tales como Interfaces de usuario de caracteres (CUI), Interfaces gráficas de usuario (GUI), Interfaces de usuario auditivas e Interfaces de usuario multimodales. El marco conceptual de UsiXML se basa en el Cameleon Reference Framework [2].

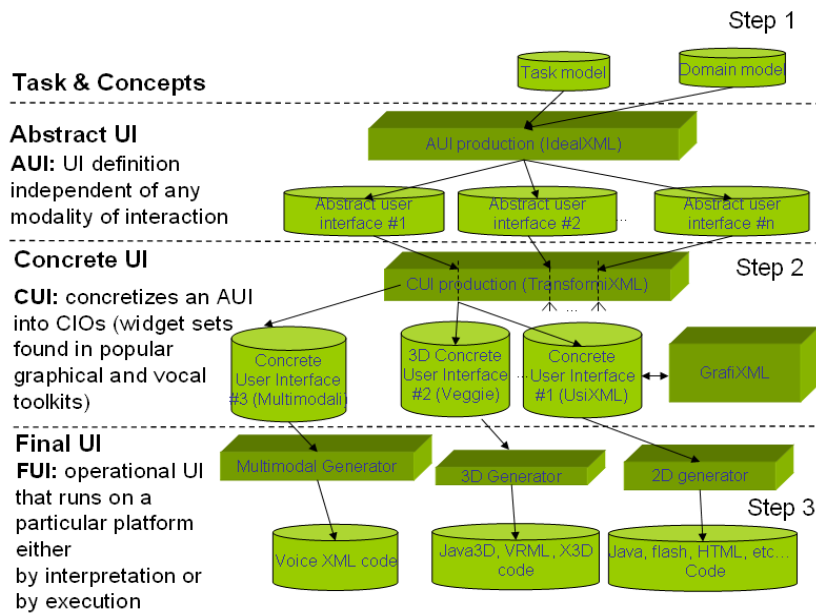


Figura 6: Descripción general del marco de generación de IU basado en UsiXML.

UsiXML propone cuatro pasos para definir la interfaz de usuario (Fig. 1). El nivel de Tareas y conceptos describe las especificaciones del sistema interactivo en términos de las tareas de usuario que se llevarán a cabo y los objetos de dominio de estas tareas. Una interfaz de usuario abstracta (AUI) abstrae una interfaz de usuario concreta (CUI) en una definición que es independiente de cualquier modalidad de interacción (como gráfica, vocal o táctil). Un CUI abstrae una Interfaz de usuario final (FUI) en una descripción independiente de cualquier lenguaje de programación o marcado en términos de Objetos de interacción concreta, diseño, navegación y comportamiento. Un FUI se refiere a una UI real procesada por interpretación (por ejemplo, HTML) o por compilación de código (por ejemplo, Java). Al re-utilizar este mecanismo, se puede generar la IU de un modelo de flujo de trabajo, que incluye modelos de tareas.

El enfoque basado en modelos tiene la intención de ayudar a diseñar interfaces de usuario con una metodología más formal respaldada por computadora en lugar del diseño de papel de información más común, como el guión gráfico. Describe el modelo de la aplicación como una especificación ejecutable, pero en una palanca de abstracción más alta que la del código. Intenta representar explícitamente el conocimiento que a menudo está oculto en el código de la aplicación. El problema de generar interfaces de usuario para mapas interactivos de gestos corporales. Nuestro escenario representa una secuencia particular de acciones que pueden realizarse con éxito para alcanzar un objetivo de tarea.

La primera tarea del enfoque es crear un modelo de tareas. Los modelos de tareas no imponen ninguna implementación en particular, por lo que las tareas de los usuarios se pueden analizar mejor sin restricciones de implementación. Este tipo de análisis es posible porque las tareas del usuario se consideran desde el punto de vista de la necesidad de los usuarios para la aplicación y no de cómo representar la actividad del usuario con un sistema en particular. Esta última capa, llamada modelo de tarea, está centrada en el usuario. Desde un punto de vista concreto obtenemos la siguiente representación en árbol, Fig. 2, en el editor CTTE [3].

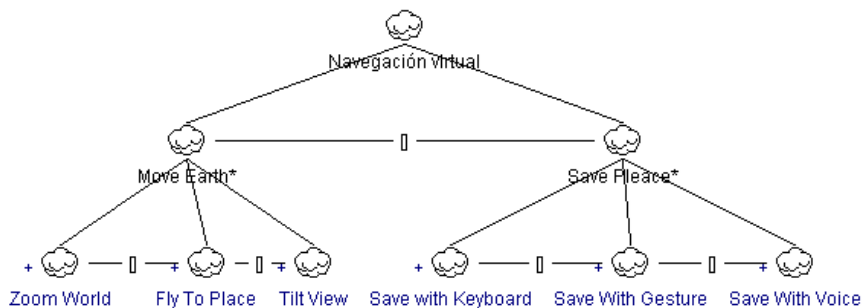


Figura 7: Mapas de navegación del árbol de tareas concurrentes. Elaborado por CTTE [3].

Es necesario para poder dar sentido a los modelos de tareas la creación de diagramas de procesos de negocios, que es un enfoque de nivel superior. Su origen es la información presente en la ingeniería de requerimientos, como podemos ver en la Fig. 3.

1. Elicitación de la organización. Suponemos que hay medios para recopilar información que servirá como entrada.

2. Modelado del flujo de trabajo del problema, que incluye la definición de: las unidades organizativas, los trabajos y los estereotipos del usuario, los modelos de procesos, los patrones de asignación del flujo de trabajo y los modelos de tareas.

3. Asignación de la especificación del flujo de trabajo en un sistema de información de flujo de trabajo.
4. Administre el flujo de trabajo a través del administrador de flujo de trabajo.

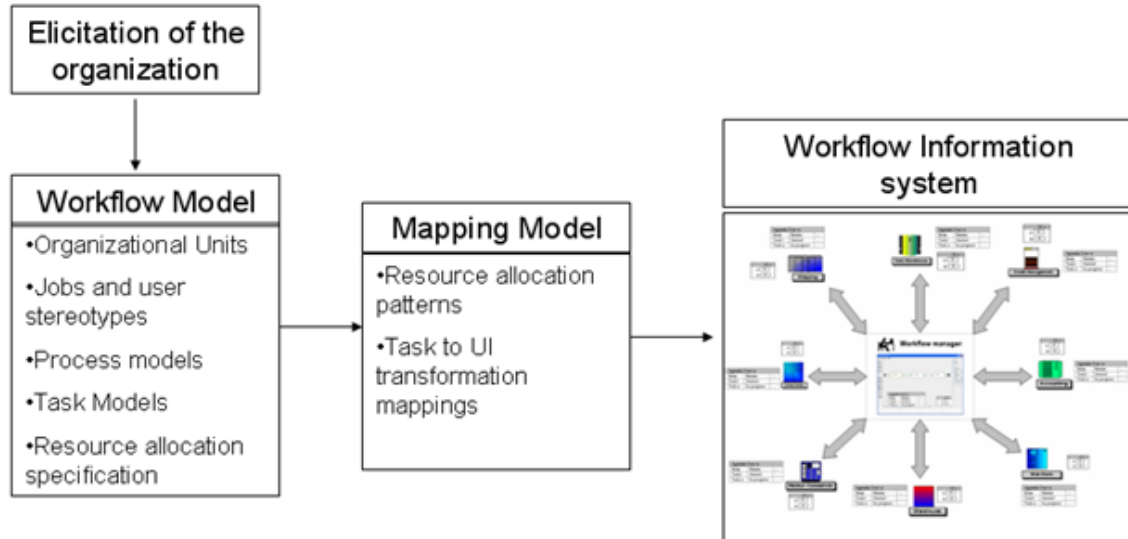


Figura 8: Vista general del método. Fuente [4]

Nuestro trabajo se centra en la etapa 1 y 2, que significa crear diagramas de proceso de negocio a partir de una elicitación de requerimientos. Y de esta forma reusar este resultado parcial en el marco del marco de referencia UsiXML.

Como se puede ver en los trabajos relacionados se quiere proponer una forma de lectura de tareas a través de un archivo XML y a partir de eso generar un modelo de proceso de negocio para facilitar el entendimiento de tareas. Se optó por usar BPMN como notación debido a que es un defacto estándar ampliamente usado en la industria actualmente.

3 Análisis

En la figura 3.1, la toma de requerimientos de un nuevo sistema parte desde el inicio, tomando los verbos importantes ya sea por medio del procesamiento de lenguaje natural y al final forme un archivo XML en donde el interprete generara un modelo de procesos de negocio. A continuación el usuario completara ese diagrama haciéndolo mas granular para especificar tareas especificas del sistema, al final se exportará a un archivo XML.



Figura 3.1: El cliente describe las funcionalidades del sistema, mientras la aplicación web hace un modelo de negocios.

El sistema para modelar procesos de negocio no contara con un registro para identificar usuarios ya que la finalidad de la plataforma esta enfocada a la lectura de archivos XML, donde cada archivo estará preestablecido con reglas de lectura para su correcta modelado en la aplicación web.

A partir del modelado generado de un archivo XML, el usuario podrá editar el modelo de negocio de acuerdo a sus necesidades.

Teniendo el modelo generado en la plataforma, se guardará en el equipo que se este utilizando en formato XML o en imagen del modelo.

La aplicación deberá estar realizada en un entorno web para que en un futuro se pueda implementar en un servidor y los usuarios de diversos lugares puedan acceder a la plataforma.

Definido el análisis acerca del sistema, pudimos sacar los requerimientos para tener en claro hasta donde se va delimitar el proyecto, esto nos dará una mejor idea de como hacer el diseño del sistema. En el diseño se especificara las funcionalidades del sistema con apoyo de UML de los diagramas de: Casos de uso, Clases, Secuencia.

4 Diseño

En esta sección se mostrará los diagramas de casos de uso, especificación de casos de uso, diagramas de secuencia y diagrama de clases en donde se detallara el funcionamiento de la aplicación web para el modelado de procesos de negocio.

4.1 Casos de uso

En el diagrama de casos de uso, se colocaran las tareas que hace cada usuario y su relación de ellas mismas. En la Figura 4.2.1 el diagrama casos de uso del sistema Aplicación web para el diseño de procesos de negocio.

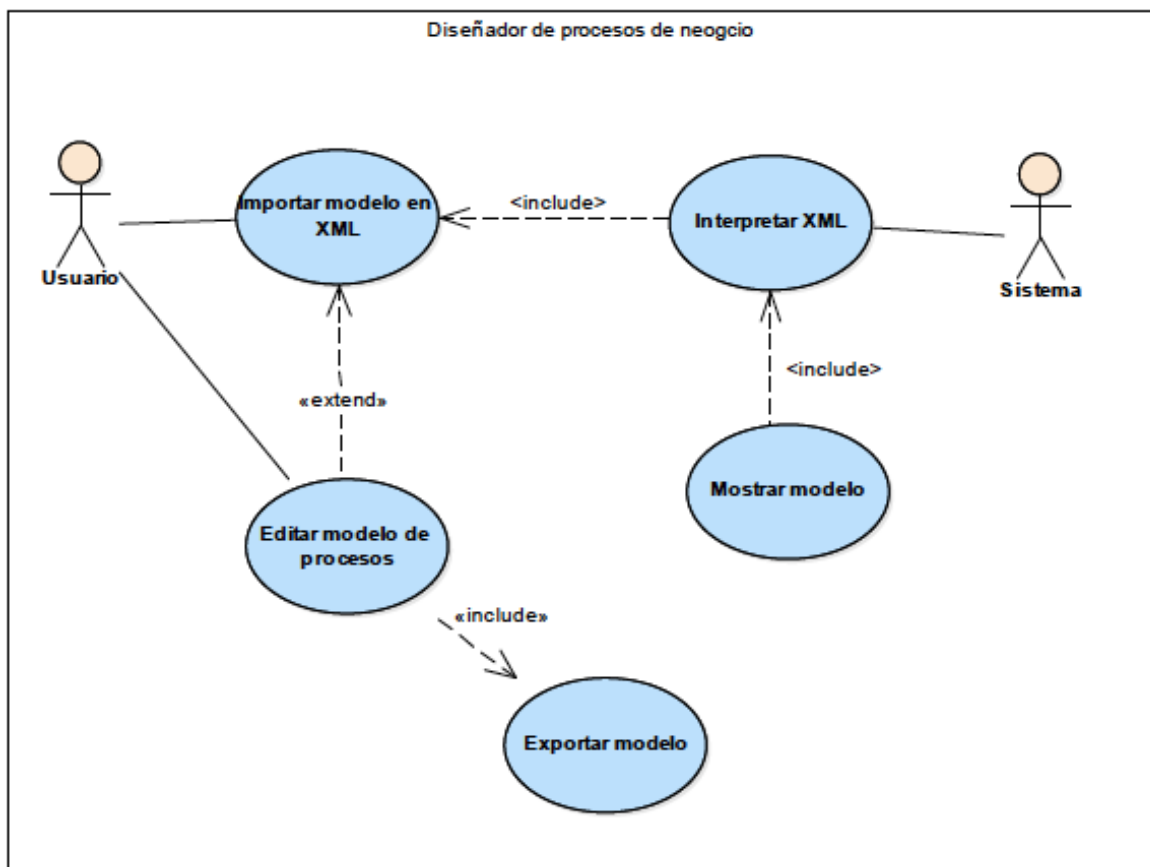


Figura 4.1: Casos de usos del modelador de procesos de negocio

4.2 Especificación de casos de uso

Se especificara cada caso de uso de la Figura 4.2.1 para entender su funcionamiento en el sistema, sus flujos alternativos de cómo realizar una tarea de diferente forma, las precondiciones que debe tener la tarea para poder realizarse , puntos de excepcion en donde si ocurre un error en donde se puede retomar la secuencia de tareas del caso de uso.

4.2.1 Importar modelo XML

Aplicación web para el diseño de procesos de negocio	Versión: 1.2
Especificación de Caso de uso: Importar modelo en XML	Fecha: 18/02/2017

Importar modelo XML

Descripción

En Importar modelo en XML, se debe partir de con un archivo XML ya con el estándar establecido para la correcta lectura de el en la aplicación web, por medio de un botón se seleccionará el archivo XML a importar.

Flujo de Eventos

Flujo Básico

1. *Seleccionar en el botón Cargar archivo*
2. *Escoger archivo en formato XML a cargar*
3. *Dar clic en aceptar PE 4.1, 4.2*
4. *Mostrar diagrama*

Flujo Alternativo

1. *Clic en crear nuevo modelo*
2. *Seguir Flujo Básico 2.1*

Precondiciones

El usuario ya debe contar con el archivo XML a cargar de acuerdo a las normas establecidas para la lectura correcta del archivo.

Puntos de Excepciones

- 1 El archivo XML no es compatible para su correcto modelado, regresar al FB2.1
- 2 El archivo no esta en el formato XML, regresar al FB 2.1

4.2.2 Editar modelo de procesos

Aplicación web para el diseño de procesos de negocio	Versión: 1.2
Especificación de Caso de uso: Editar modelo de procesos	Fecha: 18/02/2017

Editar modelo de procesos

Descripción

Editar modelo de procesos se encarga de editar el modelo del archivo XML o de un nuevo modelo, en donde se puede agregar las tareas, relaciones, interacción con base de datos etc, con la finalidad de que este mas detallado el comportamiento del sistema que se quiere modelar.

Flujo de Eventos

Flujo Básico

1. *Seleccionar componente*
2. *Arrastrar el componente al modelo*
3. *Colocar nombre del componente*
4. *Especificar componente*

Flujo Alternativo

1. *Seleccionar componente*
 2. *Eliminar componente Flujo Básico 1.1*
-
1. *Seleccionar tipo de relación*
 2. *Clic componente padre para la arista de relación*
 3. *Clic componente hijo para la arista de relación*
 4. *Colocar nombre de la relación*
 5. *Editar relación*

Precondiciones

El usuario ya debe haber realizado el caso de uso de Importar XML.

Puntos de Excepciones

- 1 Error al agregar componente, regresa al FB 1.1
- 2 Error al agregar relacion, regresar al FA 1.

4.2.3 Exportar modelo

Aplicación web para el diseño de procesos de negocio	Versión: 1.2
Especificación de Caso de uso: Exportar modelo	Fecha: 18/02/2018

Exportar modelo

Descripción

Después del caso de uso Importar modelo XML, el usuario puede exportar el modelo en cualquier momento, ya sea desde un archivo en donde todos los detalles del diagrama se pasara en un archivo XML cumpliendo con los estándares establecidos del XML Schema, o también en una imagen.

Flujo de Eventos

Flujo Básico

1. *Clic Guardar XML*
2. *Conversión de componentes y relaciones a XML*
3. *Validar XML*
4. *Descarga de archivo completado*

Flujo Alternativo

1. *Guardar modelo en imagen (JPG)*
2. *Utilizar función de transformación a imagen*
3. *Descargar de archivo completado*

Precondiciones

El usuario previamente debe haber creado un nuevo o importado un XML.

Puntos de Excepciones

- 1 Error en guardar el archivo XML. FB1.1
- 2 Error en guardar la imagen JPG. FA1.1

4.2.4 Interpretar XML

Aplicación web para el diseño de procesos de negocio	Versión: 1.2
Especificación de Caso de uso: Interpretar XML	Fecha: 18/02/2018

Interpretar XML

Descripción

Después de realizar el caso de uso Importar modelo XML y colocar la opción de cargar archivo XML, se ejecutara la función de Interpretar XML, en donde primero el archivo será validado mediante un XML Schema.

Flujo de Eventos

Flujo Básico

1. *Abrir archivo XML*
2. *Leer componentes*
3. *Comprobar estructura de componentes*
4. *Leer relaciones*
5. *Comprobar estructura de relaciones*
6. *Retornar resultado de la lectura*

Precondiciones

El usuario debio haber cargado un archivo XML previamente.

Puntos de Excepciones

- 1 Error al leer los componentes XML, regresar FB 1.1
- 2 Error al leer las relaciones, regresar FB 1.1

4.2.5 Mostrar modelo

Aplicación web para el diseño de procesos de negocio	Versión: 1.2
Especificación de Caso de uso: Mostrar modelo	Fecha: 18/02/2018

Mostrar modelo

Descripción

Después de haber interpretado el XML y verificado que tiene la correcta estructura se pasara a la graficación de los elementos del XML.

Flujo de Eventos

Flujo Básico

1. *Leer componente*
2. *Graficar componente*
3. *Leer relación*
4. *Graficar relación*
5. *Mostrar modelo*

Precondiciones

Haber realizado el caso de uso Interpretar XML.

Puntos de Excepciones

1 Error en mostrar modelo, regresar al FB 1.1

4.3 Diagrama de clases

En el diagrama de clases se vera la estructura de la aplicación web mostrando las clases del sistema, sus atributos, operaciones (o métodos), y las relaciones entre los objetos. Este diagrama será de gran ayuda para el desarrollo del sistema, ya que nos basaremos en el para su desarrollo.

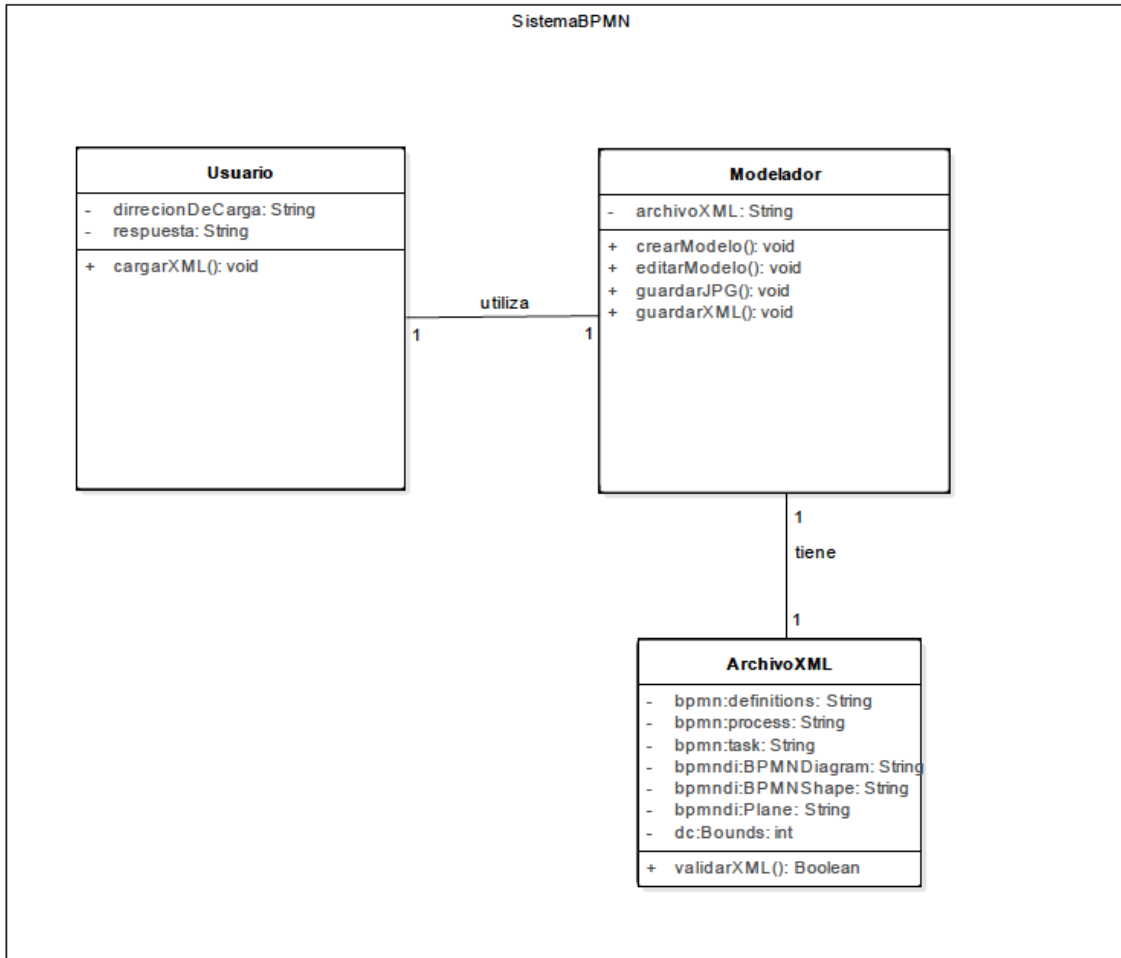


Figura 4.3: Diagrama de clases Aplicación Web BPMN

En la figura 4.3 se ven las siguientes clases principales:

Usuario: En esta clase se manejan los atributos y métodos para la correcta importación de un archivo xml.

Modelador: La clase modelador se encargara de manejar las acciones para el desarrollo del diagrama, como puede ser: crear, editar , etc.

Archivo XML: En esta clase se guardara los atributos del archivo XML y tendrá el método para validar si el XML esta correcto.

4.4 Diagramas de secuencia

En esta sección se mostrara el comportamiento del caso de uso con el actor y su ciclo de vida, con sus diferentes alternativas y casos de excepción.

4.4.1 Diagrama de secuencia Importar XML

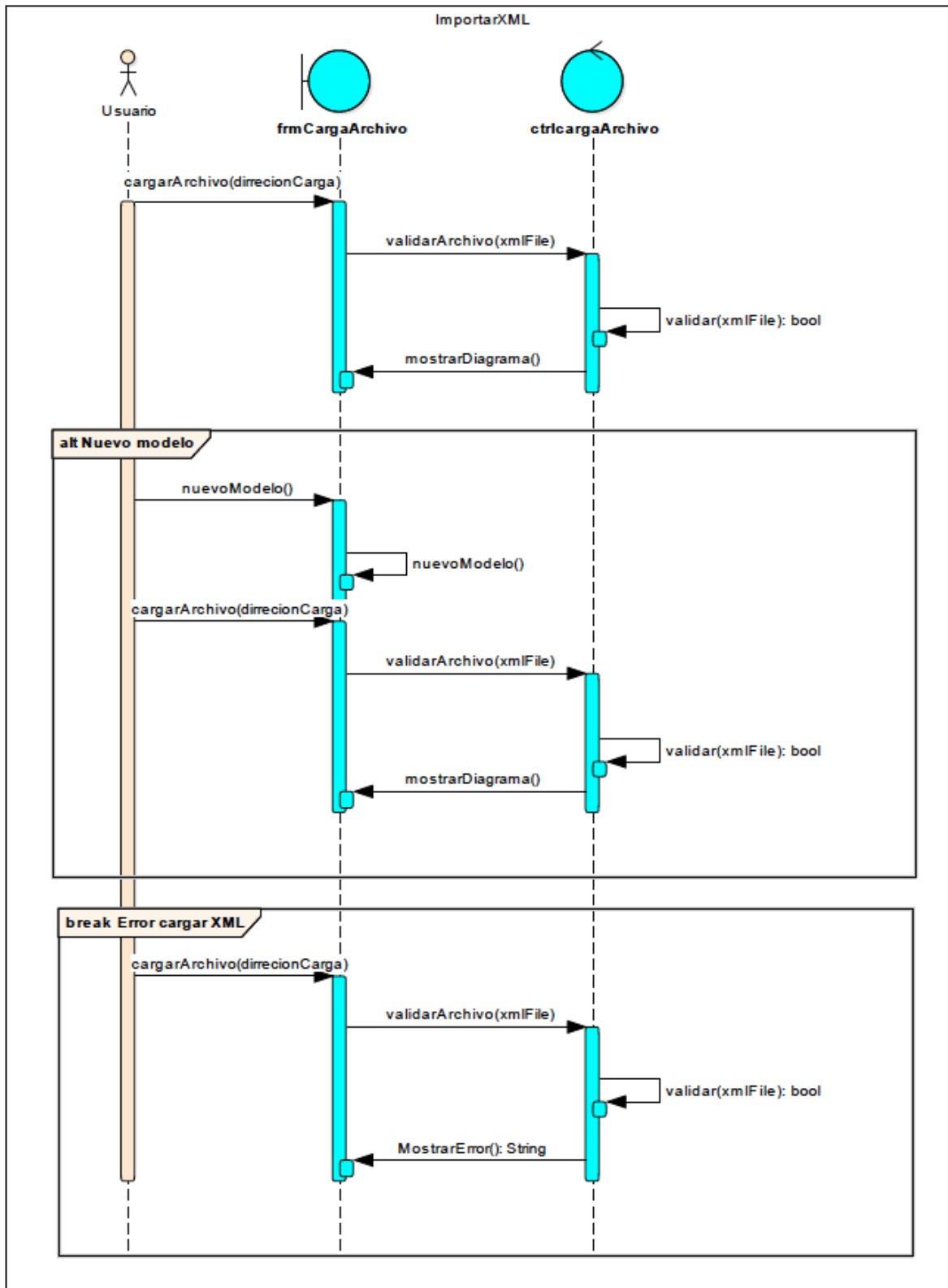


Figura 4.4.1: Importar XML

4.4.2 Diagrama de secuencia Editar modelo

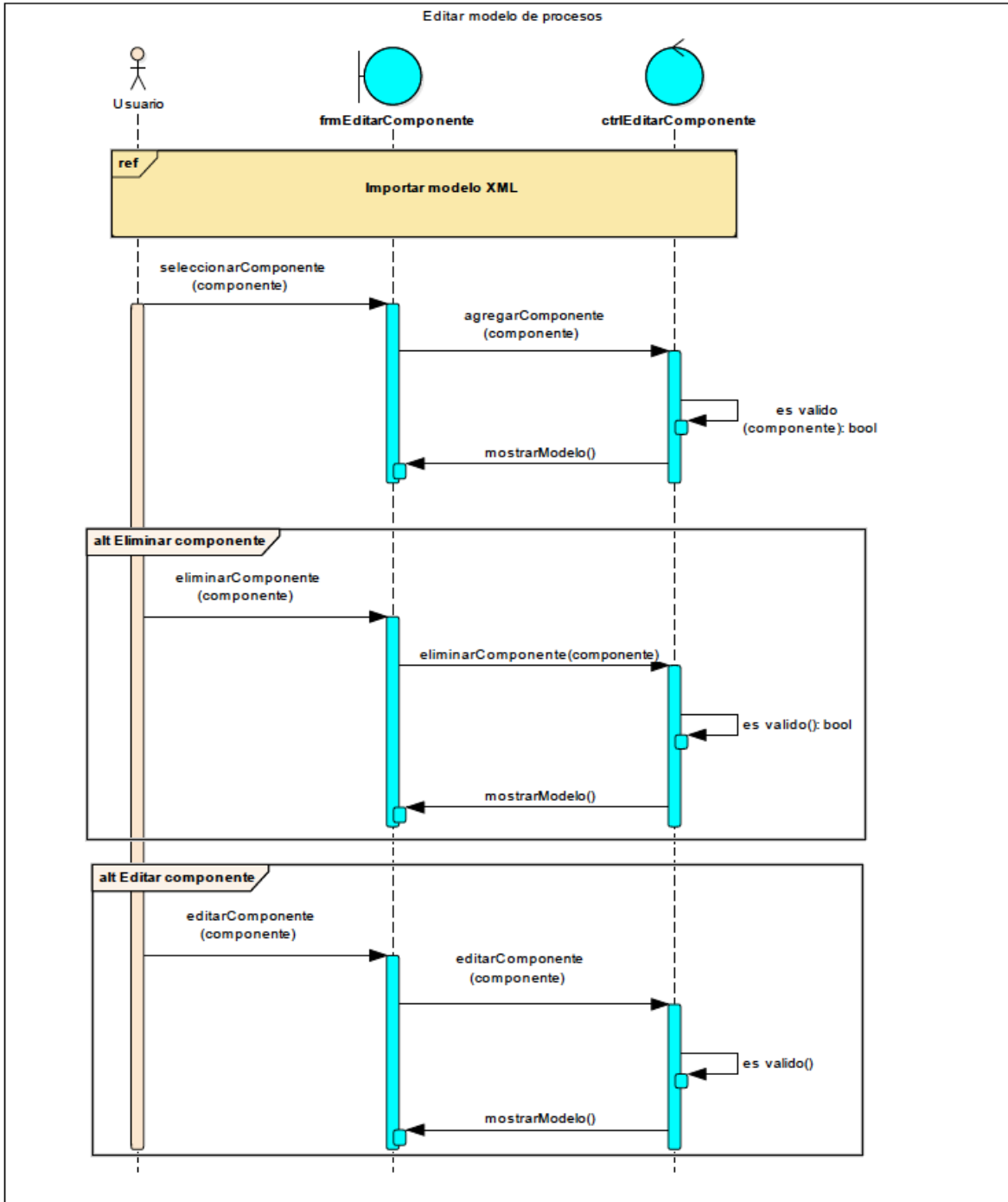


Figura 4.4.2: Editar modelo de procesos

4.4.3 Diagrama de secuencia Exportar modelo

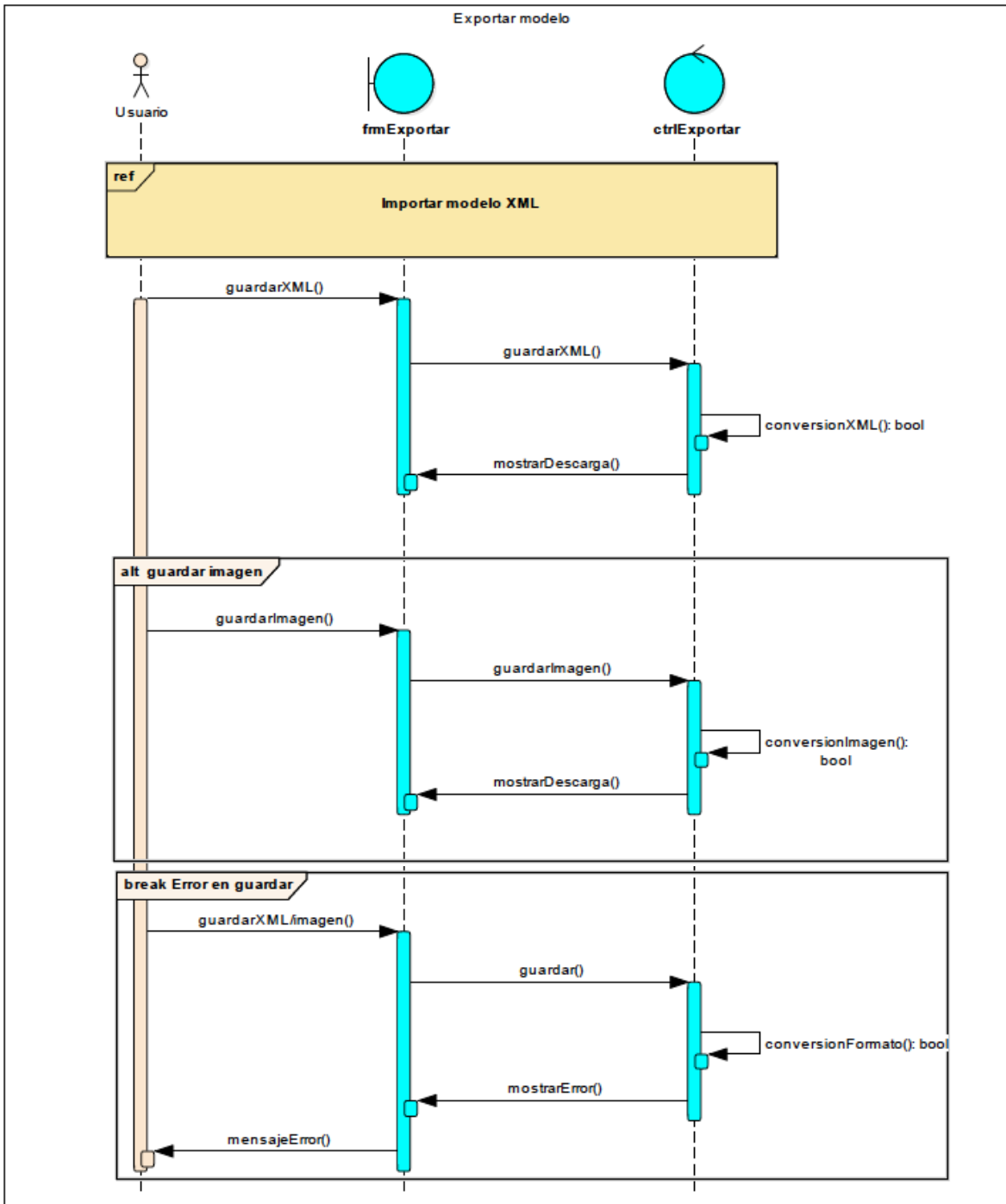


Figura 4.4.3: Exportar modelo

4.4.4 Diagrama de secuencia Interpretar XML

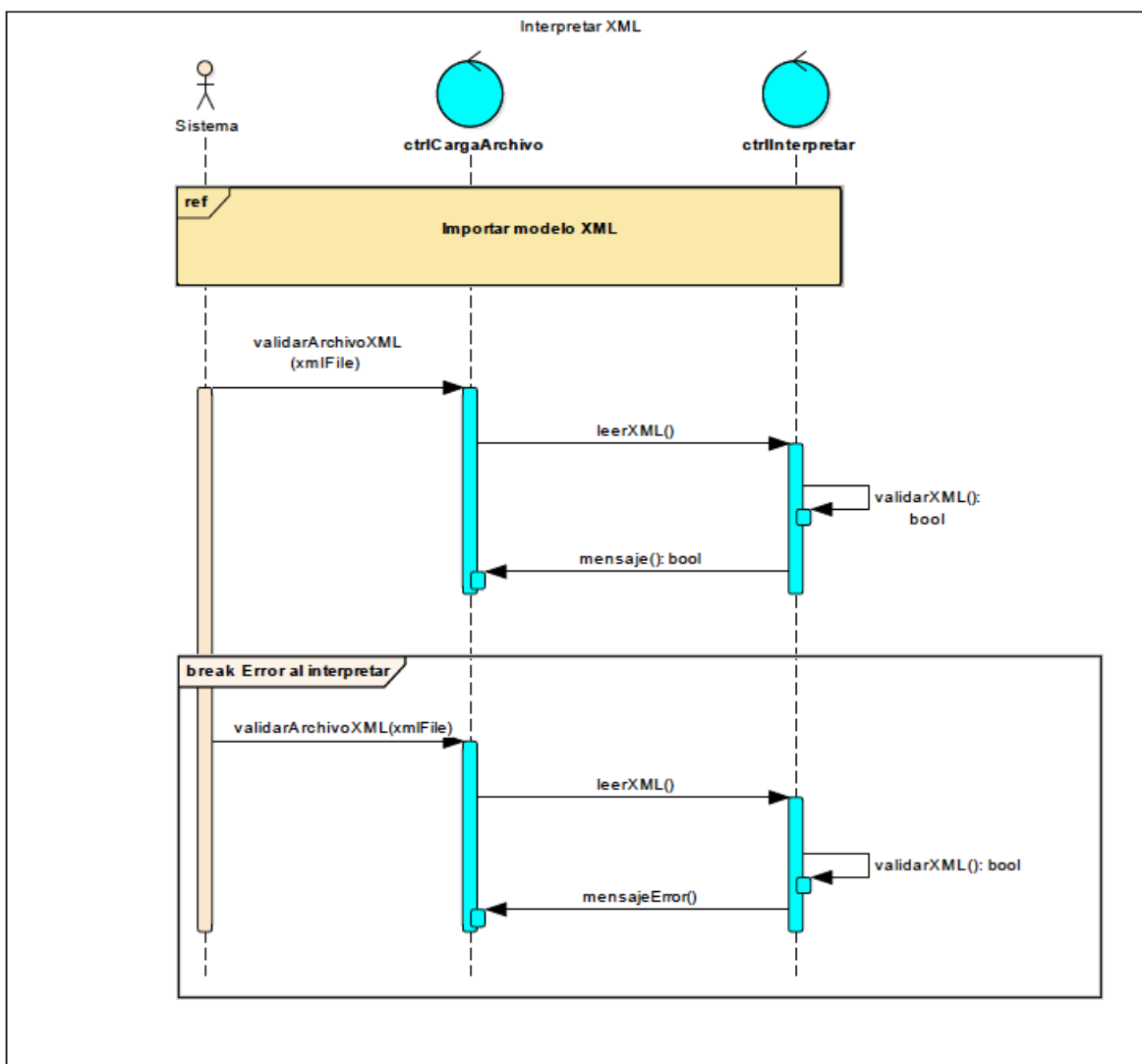


Figura 4.4.4: Interpretar XML

En los diagramas se especifico las funcionalidades y de como se interactuará con los objetos. Después de terminar los diagramas se procederá al desarrollo, se buscara la mejor herramienta para el apoyo del modelado en BPMN en una aplicación web.

5 Desarrollo

En el desarrollo se siguieron los diagramas realizados en la fase de diseño, acoplando la herramienta BPMN según nuestras necesidades y estandarizando el archivo XML para las lecturas.

5.1 Selección de herramienta BPMN

Para el desarrollo de este proyecto se investigo que herramientas de BPMN se podrían acoplar a la aplicación web, en donde se buscaban las siguientes características: confiabilidad, fácil de usar y un soporte constante a la herramienta. La herramienta que se selecciono fue BPMN-JS por su soporte que le da comunda y su facil lectura de elementos del archivo XML al modelo.

Las siguientes herramientas que se investigaron fueron las siguientes:

Nombre	Desarrollador	Lenjuage	¿Es web?	Desventajas
BPMN-Modeler	Eclipse	Java	No	Complejo, No es nativo para una aplicación web
BPMN-JS	Camunda	Node JS	Si	Uno de los inconvenientes es la falta de especificación el modelo o componentes
BPMN	Camunda	Java	No	Ya esta desactualizado y no le dan soporte.
BPMN	GoJS	JavaScript	Si	Nueva herramienta, falta de especificación el modelo o componentes

Figura 5: Investigación de herramientas BPMN para una aplicación web

5.2 Estructura XML

Para que la aplicación web funcione bien como interprete, el canal de comunicación se debe estandarizar es decir el archivo XML que se va a leer para generar las tareas y las muestres en el modelador. A continuación se mostrara las características que debe tener el archivo XML.

Se compone en dos partes el archivo XML:

Definición

- **Bpmn:definitions**
Esta etiqueta tiene como función validar el archivo xml a través de schemas que son dadas por la aplicación BPMN-JS.
- **Bpmn:process**
Se definira todos las tareas en esta etiqueta y contendra un id.
- **Bpmn:task**
Contiene un identificador unico donde se pueda identificar la tarea y el nombre de la tarea.

Diseño

- **Bpmndi:BPMNDiagram**
Contiene un identificador del diagrama de diseño
- **Bpmndi:BPMNPlane**
Tiene un identificador dado en el BPMNDiagram y relaciona con el identificador BPMN:process
- **Bpmndi:BPMNShape**
Identificador de la figura y elemento BPMN unico
- **Dc:Bounds**
Contiene los ejes X, Y , altura y ancho de la figura

En la sección de desarrollo, se pusieron en practica los conocimientos de HTML, NODEJS para el desarrollo de la aplicación. Donde se tuvieron mas contra tiempos en el desarrollo fue comprender el funcionamiento de la herramienta BPMN-JS con NODEJS, para probar la aplicación se planteo un caso hipotético en el cual se vera en la siguiente sección.

Es importante aclarar que se ocuparon las funciones de la API de BPM-JS para el desarrollo de la aplicación web.

5.3 Implementación caso de uso Importar XML

A continuación se explicara el código que se utiliza para importar un XML y transformarlo en un modelo de procesos de negocio.

```
function openDiagram(xml) {  
  
    modeler.importXML(xml, function(err) {  
  
        if (err) {  
            container  
                .removeClass('with-diagram')  
                .addClass('with-error');  
  
            container.find('.error pre').text(err.message);  
  
            console.error(err);  
        } else {  
            container  
                .removeClass('with-error')  
                .addClass('with-diagram');  
        }  
  
    });  
}
```

En la función se comprueba que el xml este correcto y bien formado, invocando la clase de la API de BPM-JS llamado “with error”, si esta bien formado se va remover la clase “with error” y se agregara la la clase “with-diagram”.

En el caso alternativo que se cree un nuevo diagrama se ejecuta esta función:

```
function createNewDiagram() {  
    openDiagram(newDiagramXML);  
}
```

En el siguiente codigo se invocara la funcion openDiagram pasando como parametro newDiagramXML, que contiene un modelo de ejemplo.

6 Pruebas

En la sección de pruebas se planteara un caso de estudio hipotético en donde la aplicación web entrara como intérprete de la toma de requerimientos y exportara en archivo XML, para que una aplicación final desarrolle un sistema a partir del archivo XML.

6.1 Toma de requerimientos

El caso es el siguiente: existe un cliente que cuenta con un servicio de venta telefónica y le pide al desarrollador un sistema en donde le especifica que funciones quiere que tenga el sistema, mediante el procesamiento de lenguaje natural, se extraen los siguientes verbos y actores:

- Cliente
 - Seleccionar articulo
 - Solicitar articulo
 - Reclamar articulo
 - Pagar articulo

- Encargado Pedidos
 - Recibir pedidos
 - Tranquilizar al cliente

- Almacén
 - Preparar entrega
 - Entrega a repartidor

- Contabilidad
 - Registrar envío
 - Registrar pago

- Repartidor
 - Recibir entrega
 - Entregar artículo
 - Recibir pago

A partir de la información obtenida del procesamiento de lenguaje natural se procederá a generar el archivo XML.

6.2 Elaboración del XML

En la siguiente Figura 6.2.1 se muestra el archivo XML ya estandarizado que se genero a partir de la toma de requerimientos.

```

<?xml version="1.0" encoding="UTF-8"?>
<bpmn:definitions xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:bpmn="http://www.omg.org/spec/BPMN/20100524/MODEL"
xmlns:bpmndi="http://www.omg.org/spec/BPMN/20100524/DI"
xmlns:dc="http://www.omg.org/spec/DD/20100524/DC" id="Definitions_00tiv7q"
targetNamespace="http://bpmn.io/schema/bpmn">
  <bpmn:collaboration id="Collaboration_10m9nxc">
    <bpmn:participant id="Participant_0xw6rsz" name="Servicio Venta Telefónica"
processRef="Process_1" />
  </bpmn:collaboration>
  <bpmn:process id="Process_1" isExecutable="false">
    <bpmn:laneSet id="LaneSet_0z4tuir">
      <bpmn:lane id="Lane_06lqxqp" name="Almacén">
        <bpmn:flowNodeRef>Task_1ocb0nx</bpmn:flowNodeRef>
        <bpmn:flowNodeRef>Task_1rpgs69</bpmn:flowNodeRef>
      </bpmn:lane>
      <bpmn:lane id="Lane_188theo" name="Contabilidad">
        <bpmn:flowNodeRef>Task_15je4ul</bpmn:flowNodeRef>
        <bpmn:flowNodeRef>Task_0onnpqv</bpmn:flowNodeRef>
        <bpmn:childLaneSet id="LaneSet_1hkaqh5" />
      </bpmn:lane>
      <bpmn:lane id="Lane_0mukuhx" name="Encargado Pedidos">
        <bpmn:flowNodeRef>Task_04ndw0f</bpmn:flowNodeRef>
        <bpmn:flowNodeRef>Task_0vpupth</bpmn:flowNodeRef>
      </bpmn:lane>
      <bpmn:lane id="Lane_1mgjm3g" name="Cleinte">
        <bpmn:flowNodeRef>Task_1n54kx8</bpmn:flowNodeRef>
        <bpmn:flowNodeRef>Task_17k27d9</bpmn:flowNodeRef>
        <bpmn:flowNodeRef>Task_1v8vkz1</bpmn:flowNodeRef>
        <bpmn:flowNodeRef>Task_1oezqh1</bpmn:flowNodeRef>
        <bpmn:childLaneSet id="LaneSet_0nsfr3v" />
      </bpmn:lane>
      <bpmn:lane id="Lane_0sq0u2m" name="Repartidor">
        <bpmn:flowNodeRef>Task_1n5ifql</bpmn:flowNodeRef>
        <bpmn:flowNodeRef>Task_09z07p5</bpmn:flowNodeRef>
        <bpmn:flowNodeRef>Task_1dl4rvq</bpmn:flowNodeRef>
      </bpmn:lane>
    </bpmn:laneSet>
    <bpmn:task id="Task_1n54kx8" name="Solicitar articulo" />
    <bpmn:task id="Task_17k27d9" name="Seleccionar articulo" />
    <bpmn:task id="Task_1v8vkz1" name="Reclamar artículo" />
    <bpmn:task id="Task_1oezqh1" name="Pagar articulo" />
    <bpmn:task id="Task_04ndw0f" name="Recibir pedido" />
    <bpmn:task id="Task_0vpupth" name="Tranquilizar cliente" />
    <bpmn:task id="Task_1ocb0nx" name="Preparar entrega" />
    <bpmn:task id="Task_1rpgs69" name="Entrega a repartidor" />
    <bpmn:task id="Task_15je4ul" name="Registrar Envío" />
    <bpmn:task id="Task_0onnpqv" name="Registrar pago" />
    <bpmn:task id="Task_1n5ifql" name="Recibir entrega" />
    <bpmn:task id="Task_09z07p5" name="Entregar artículo" />
    <bpmn:task id="Task_1dl4rvq" name="Recibir pago" />
  </bpmn:process>
  <bpmndi:BPMNDiagram id="BPMNDiagram_1">
    <bpmndi:BPMNPlane id="BPMNPlane_1" bpmnElement="Collaboration_10m9nxc">
      <bpmndi:BPMNShape id="Participant_0xw6rsz_di"
bpmnElement="Participant_0xw6rsz">
        <dc:Bounds x="146" y="-277" width="642.75" height="744" />
      </bpmndi:BPMNShape>
      <bpmndi:BPMNShape id="Lane_06lqxqp_di" bpmnElement="Lane_06lqxqp">
        <dc:Bounds x="176" y="30" width="612.75" height="147" />
      </bpmndi:BPMNShape>
      <bpmndi:BPMNShape id="Lane_188theo_di" bpmnElement="Lane_188theo">
        <dc:Bounds x="176" y="177" width="612.75" height="156" />
      </bpmndi:BPMNShape>
    </bpmndi:BPMNPlane>
  </bpmndi:BPMNDiagram>

```

```

</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Lane_0mukuhx_di" bpmnElement="Lane_0mukuhx">
  <dc:Bounds x="176" y="-117" width="612.75" height="147" />
</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Lane_1mgjm3g_di" bpmnElement="Lane_1mgjm3g">
  <dc:Bounds x="176" y="-277" width="612.75" height="160" />
</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Lane_0sq0u2m_di" bpmnElement="Lane_0sq0u2m">
  <dc:Bounds x="176" y="333" width="612.75" height="134" />
</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Task_17k27d9_di" bpmnElement="Task_17k27d9">
  <dc:Bounds x="232.75" y="-241.25" width="100" height="80" />
</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Task_1n54kx8_di" bpmnElement="Task_1n54kx8">
  <dc:Bounds x="366.75" y="-241" width="100" height="80" />
</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Task_1v8vkz1_di" bpmnElement="Task_1v8vkz1">
  <dc:Bounds x="484.75" y="-241" width="100" height="80" />
</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Task_loezqh1_di" bpmnElement="Task_loezqh1">
  <dc:Bounds x="615.75" y="-241" width="100" height="80" />
</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Task_04ndw0f_di" bpmnElement="Task_04ndw0f">
  <dc:Bounds x="301.75" y="-92.25" width="100" height="80" />
</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Task_0vpupth_di" bpmnElement="Task_0vpupth">
  <dc:Bounds x="467.75" y="-92" width="100" height="80" />
</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Task_1ocb0nx_di" bpmnElement="Task_1ocb0nx">
  <dc:Bounds x="302" y="55.75" width="100" height="80" />
</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Task_1rpgs69_di" bpmnElement="Task_1rpgs69">
  <dc:Bounds x="468" y="56" width="100" height="80" />
</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Task_15je4ul_di" bpmnElement="Task_15je4ul">
  <dc:Bounds x="294.75" y="203.75" width="100" height="80" />
</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Task_0onnpqv_di" bpmnElement="Task_0onnpqv">
  <dc:Bounds x="539.75" y="204" width="100" height="80" />
</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Task_1n5ifql_di" bpmnElement="Task_1n5ifql">
  <dc:Bounds x="302" y="354.75" width="100" height="80" />
</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Task_09z07p5_di" bpmnElement="Task_09z07p5">
  <dc:Bounds x="433.5" y="355" width="100" height="80" />
</bpmndi:BPMNShape>
<bpmndi:BPMNShape id="Task_1dl4rvrg_di" bpmnElement="Task_1dl4rvrg">
  <dc:Bounds x="560.5" y="355" width="100" height="80" />
</bpmndi:BPMNShape>
</bpmndi:BPMNPlane>
</bpmndi:BPMNDiagram>
</bpmn:definitions>
<?xml version="1.0" encoding="UTF-8"?>

```

Figura 6.2: Código generado de la toma de requerimientos

6.3 Intérprete aplicación web

Después de haber generado el archivo el XML, el desarrollador utilizara la herramienta web para crear el modelo de negocios y hacerlo mas granular.

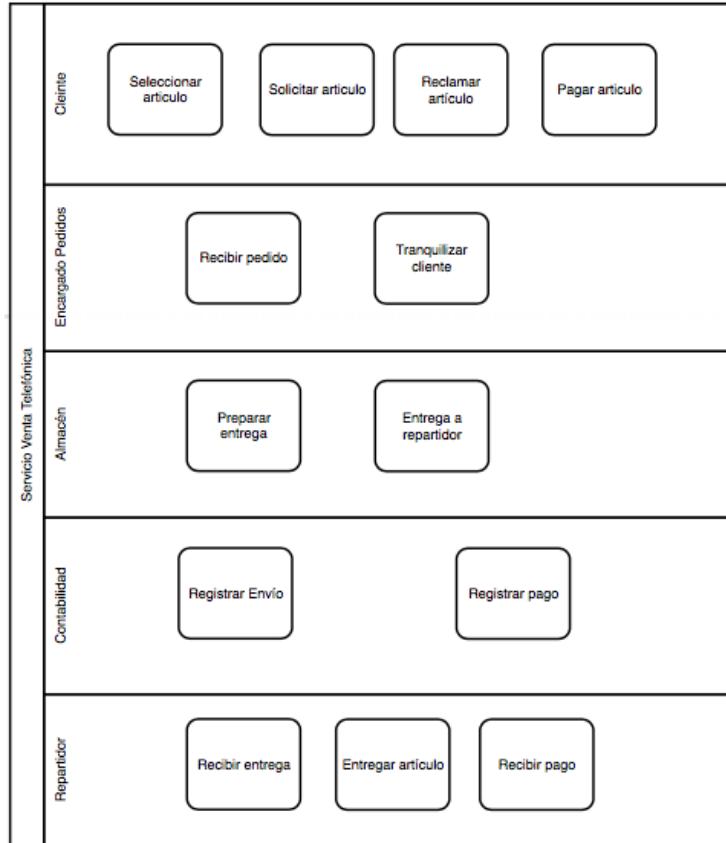


Figura 6.3: Modelo de procesos de negocio ya interpretado por la aplicación web.

6.4 Editar modelo BPMN

Después de importar el archivo XML y ya mostrado por la aplicación web se puede proceder a editar el diagrama y hacerlo mas detallado.

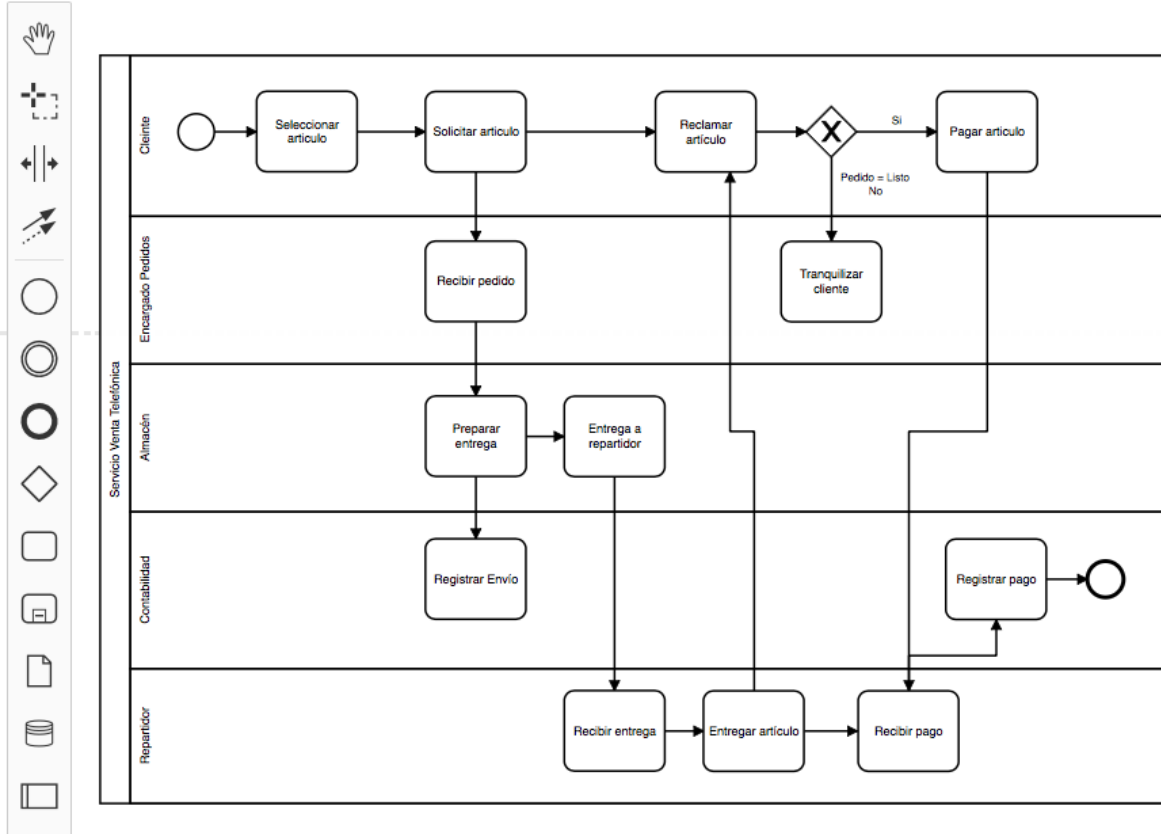
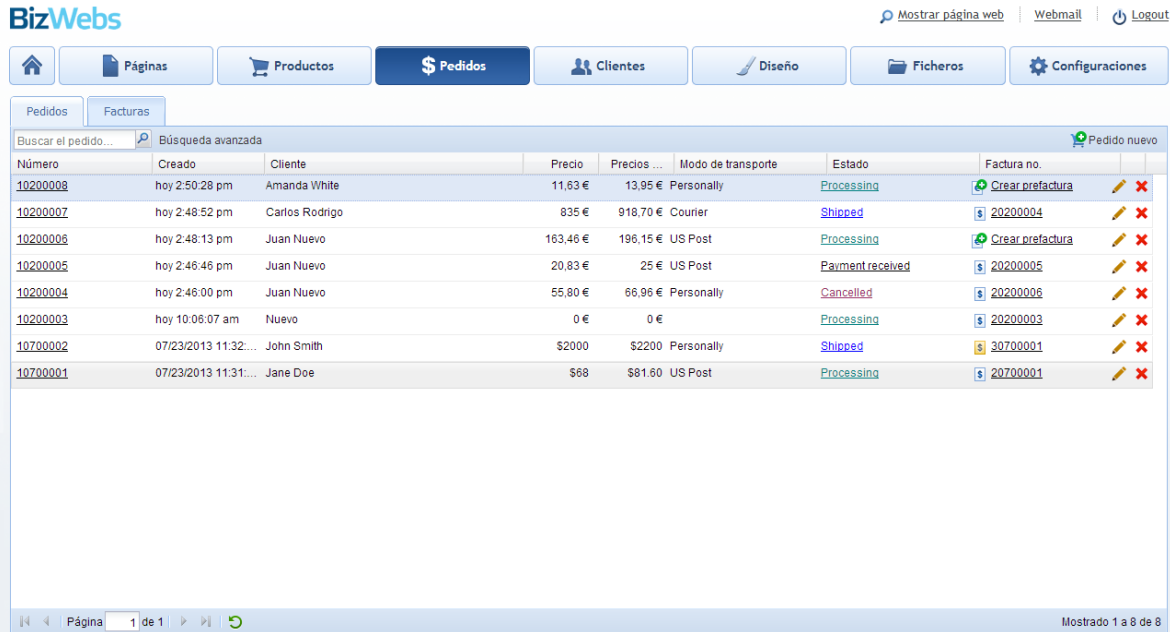


Figura 6.4: se muestra el modelo de proceso de negocio de Servicio Venta Telefónica, ya terminado y listo para exportarlo a XML o guardarlo en imagen.

6.5 Generar Sistema

Ya teniendo el archivo XML generado por la aplicación web, se procederá mediante otro sistema a leer el XML y generar un sistema prototipo con las funciones definidas en el modelo BPMN.



The screenshot shows the BizWebs application interface. At the top, there is a navigation bar with the BizWebs logo and several menu items: Páginas, Productos, Pedidos (highlighted), Clientes, Diseño, Ficheros, and Configuraciones. Below the navigation bar, there is a search bar and a table of orders. The table has columns for Número, Creado, Cliente, Precio, Precios..., Modo de transporte, Estado, and Factura no. The table contains 8 rows of order data. At the bottom of the page, there is a pagination bar showing 'Página 1 de 1' and 'Mostrado 1 a 8 de 8'.

Número	Creado	Cliente	Precio	Precios ...	Modo de transporte	Estado	Factura no.
10200008	hoy 2:50:28 pm	Amanda White	11,53 €	13,95 €	Personally	Processing	Crear prefactura
10200007	hoy 2:48:52 pm	Carlos Rodrigo	835 €	918,70 €	Courier	Shipped	20200004
10200006	hoy 2:48:13 pm	Juan Nuevo	163,46 €	196,15 €	US Post	Processing	Crear prefactura
10200005	hoy 2:46:46 pm	Juan Nuevo	20,83 €	25 €	US Post	Payment received	20200005
10200004	hoy 2:46:00 pm	Juan Nuevo	55,80 €	66,96 €	Personally	Cancelled	20200006
10200003	hoy 10:06:07 am	Nuevo	0 €	0 €		Processing	20200003
10700002	07/23/2013 11:32:...	John Smith	\$2000	\$2200	Personally	Shipped	30700001
10700001	07/23/2013 11:31:...	Jane Doe	\$68	\$81,60	US Post	Processing	20700001

Figura 6.5 se muestra un sistema prototipo como podría quedar después de la lectura del archivo XML generado por la aplicación web BPMN.

7 Conclusiones

En el desarrollo de este proyecto se estudio los beneficios del modelado de procesos de negocio enfocado hacia el desarrollo de software. En el enfoque transformacional con metodologías como UsiXML, es posible debido a que podemos generar sistemas interactivos a partir de modelos de tareas. El reto actual consiste en definir los modelos correspondientes y las reglas de transformación. Nuestro trabajo abona un poco en esta dirección con un interprete de tareas a modelado de proceso de negocio, modelos para las tareas y los procesos de negocio. Los beneficios de adoptar esta estrategia son muchos ya que podremos integrar soluciones existentes de este marco de referencia a nuestra solución lo cual la hace más robusta.

La adopción de una notación como BPMN fue necesaria como una actualización a la metodología ya que en el pasado usaban las redes de Petri para modelar procesos. Sin embargo, hoy en día BPMN es ampliamente usado en la industria y academia por lo que esta transición era necesaria.

En esta área de desarrollo de software con BPMN, casi no existen herramientas suficientes que a partir de un modelo se genere un sistema. Es una área de oportunidad para crear software a partir de la toma de requerimientos, del cual uno de los grandes beneficios sería en el tiempo de desarrollo de software sea cada vez menor y mas efectivo.

El proceso de desarrollo implico varios retos para integrar la aplicación web, desde que se requirió que la aplicación fuera hecho en un entorno web para su fácil acceso, se tuvo que investigar que herramientas existían en BPMN para el diseño de modelo de tareas, ya que hacer un modelador de procesos de negocio desde cero sería tardado y realizar una gran variedad de pruebas para ver su correcto funcionamiento.

Después de ver que la herramienta indicada era BPMN-JS se tuvo que hacer ingeniería inversa para ver como era su funcionamiento, desde como se asía la lectura del archivo XML hasta como exportarlo. Fue complicado ya que esta herramienta esta desarrollada en Node.js y se tuvo una curva de aprendizaje para aprender sobre este lenguaje, otro de los retos de la aplicación era saber como instalarlo en un servidor local.

En trabajo futuro se espera tener un sistema de procesamiento de lenguaje natural, en donde el cliente este explicando el sistema que necesita y el software tome las palabras clave del cliente, como podría ser verbos y sujetos, después se convertiría en un archivo XML y la aplicación web BPMN, se encargaría de hacer la lectura de este archivo y mostrarlo en un modelo.

8 Bibliografía

- [1] Vanderdonckt, J., 2005, June. A MDA-compliant environment for developing user interfaces of information systems. In International Conference on Advanced Information Systems Engineering (pp. 16-31). Springer Berlin Heidelberg.
- [2] Calvary, G., Coutaz, J., Thevenin, D., Limbourg, Q., Bouillon, L. and Vanderdonckt, J., 2003. A unifying reference framework for multi-target user interfaces. *Interacting with computers*, 15(3), pp.289-308.
- [3] Paternò, F., Mancini, C., & Meniconi, S. (1997). ConcurTaskTrees: A diagrammatic notation for specifying task models. In *Human-Computer Interaction INTERACT'97* (pp. 362-369). Springer US.
- [4] Guerrero Garcia, Josefina, Jean Vanderdonckt, and Juan Manuel Gonzalez Calleros. "FlowiXML: a step towards designing workflow management systems." *International Journal of Web Engineering and Technology* 4.2 (2008): 163-182.
- [5] van der Aalst, W.M.P. The Application of Petri Nets to Workflow Management. *The Journal of Circuits, Systems and Computers*, 8(1):21--66, 1998.
- [6] W3C (2005) Working Draft. World Wide Web Consortium. 5 July 2005. <http://www.w3.org>
- [7] Wodtke, D., Weikum, G. A Formal Foundation for Distributed Workflow Execution Based on State Charts. In *Proc. of the 6th International Conference on Database Theory (ICDT '97)*, Springer-Verlag, LNCS Series, p. 230-246, 1997.
- [8] Object Management Group: Business process modeling notation specification, final adopted specification dtc/06-02-02 (2006).
- [9] Dumas, M. and ter Hofstede, A. UML Activity Diagrams as a Workflow Specification Language. In M. Gogolla and C. Kobryn, editors, *Fourth International Conference on the Unified Modeling Language (UML 2001)*, pp. 76-90, Toronto, Canada, 2001
- [10] Stavness, N. & Schneider, K. Supporting Flexible Business Processes with a Progression Model. *Workshop: Making Model-based UI Design Practical: Usable and Open Methods and Tool*, 2004.
- [11] Esposito, D. Getting Started with Microsoft Windows Workflow Foundation: A Developer Walkthrough. September 2005. <http://msdn.microsoft.com/winfx/reference/workflow>
- [12] IBM WebSphere® MQ Workflow, 2006. <http://www-306.ibm.com/software/integration/wmqwf/>

[13] Casati, F. et al. WIDE Workflow model and architecture. Technical Report 96-19, Centre for Telematics and Information Technology (CTIT), University of Twente.

[14] Visual Paradigm International Ltd. Business Process Virtual ARCHITECT, User's guide. Retrieved January 2, 2007.