

Bizativiti

Grupo 4

Modelo de Diseño

Versión 1.3

Historia de revisiones

Fecha	Versión	Descripción	Autor
24/08/2013	1.0	Casos de Uso	Juan Pablo Lorier
15/09/2013	1.0	Revisión SQA	Valeria Rocha
27/09/2013	1.1	Diagramas de clases	Juan Pablo Lorier
19/09/2013	1.1	Revisión SQA	Valeria Rocha
13/10/2013	1.2	Adición de REST API y Servlet UI	Juan Pablo Lorier
13/10/2013	1.2	Revisión SQA	Valeria Rocha
24/10/2013	1.3	Actualización de UI	Juan Pablo Lorier
27/10/2013	1.3	Revisión SQA	Valeria Rocha

Índice

[Introducción](#)

[Propósito](#)

[Alcance](#)

[Definiciones, siglas y abreviaturas.](#)

[Referencias](#)

[Visión general](#)

[Diseño de Casos de Uso](#)

[Diseño del Caso de Uso Convertir archivo](#)

[Diagrama de paquetes](#)

[Diagrama de Interacción](#)

[Diseño de Flujo de eventos](#)

[Requerimientos especiales o de implementación](#)

[Diseño de Objetos](#)

[MetamodelElement](#)

[IPlugin](#)

[Translator](#)

[PluginManager](#)

[Diseño de Subsistemas](#)

[Subsistemas Específicos](#)

[Revisión de la Interfaz de Usuario](#)

[Formatos individuales de interfaz de pantalla](#)

[Controles y elementos de diseño de interfaz de pantalla](#)

[Forma de navegación de interfaz de pantalla](#)

[Formatos de impresión](#)

[Diseño de Datos](#)

[Modelo de Datos del Núcleo](#)

[Modelo de Datos General](#)

[Especificación de la Distribución de Datos](#)

Introducción

El documento describe el diseño del sistema Bizativiti a partir de distintos artefactos de representación.

Propósito

El documento busca ser una guía para la implementación del producto aportando los aspectos del diseño del mismo.

Alcance

Ser una herramienta de referencia para los implementadores.

Definiciones, siglas y abreviaturas.

Ver glosario

Referencias

- Glosario versión 1.1
- Especificación de Requerimientos versión 3.
- Modelo de Casos de Uso de la descripción de la arquitectura versión 1.2
- Pautas para la interfaz de usuario versión 3

Visión general

Busca hacer una descripción del caso de uso principal

Diseño de Casos de Uso

Diseño del Caso de Uso Convertir archivo

Diagrama de paquetes

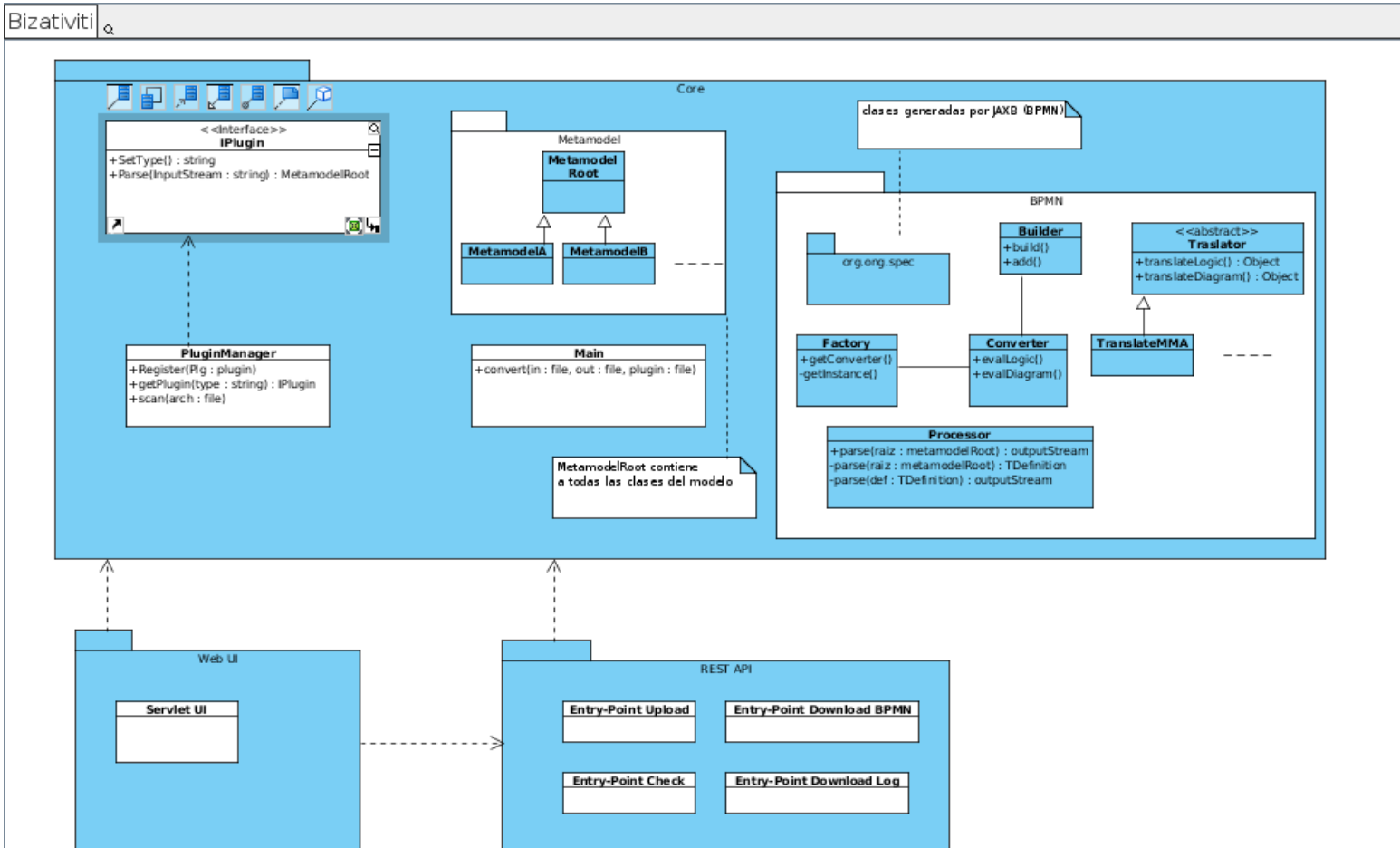
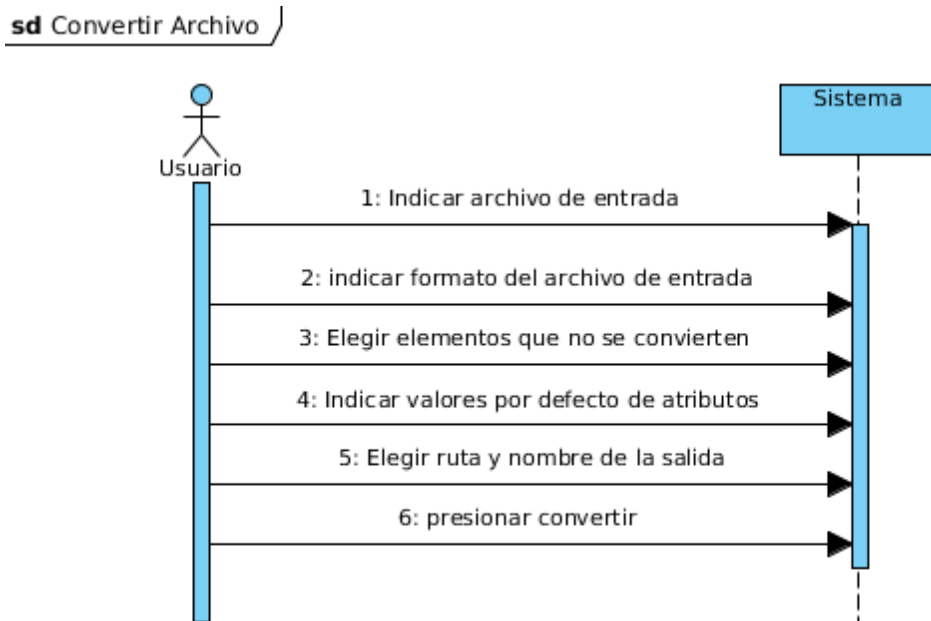


Diagrama de Interacción



Diseño de Flujo de eventos

El usuario elige el archivo a convertir, elige el tipo de archivo de origen, indica el destino del archivo convertido y selecciona convertir.

Opcionalmente puede elegir no convertir algún elemento deseleccionándolo de la lista de elementos a convertir. En ese caso, el elemento no será convertido aunque se encuentre en el modelo original. En caso de ser de los tipos actividad, evento o compuerta, tampoco aparecerán los elementos que tengan una referencia al mismo, ya sean líneas de flujo que le lleguen, que salgan, o sean eventos adjuntos. La otra opción de configuración que se brindará en la aplicación web es una lista de atributos extendidos por defecto que todas las tareas en el nuevo modelo deben tener. Si en el modelo original se cuenta con este atributo, se toma el que venía en el modelo original, pero si no cuenta con los atributos configurados por defecto, se agrega con el valor configurado. Estos atributos extendidos aplican para las tareas.

El sistema crea el archivo en la ruta de destino en formato XML según el estándar BPMN 2.0.

Requerimientos especiales o de implementación

Debe funcionar adecuadamente en todas las plataformas aceptadas (Windows, Linux, Mac OS) y los modelos generados deben representar el modelo de origen.

En caso de encontrar algún problema durante la conversión, el sistema debe devolver mensajes descriptivos de dicho problema.

Diseño de Objetos

MetamodelElement

Generalización que incluye todas las clases que definen los distintos objetos de procesos soportados por el sistema. Estas clases sirven de estructura intermedia en el pasaje desde el formato de entrada a BPMN 2.0

IPlugin

Clase que define las reglas de conversión de un tipo específico de archivo de entrada. Sirve para cumplir con la arquitectura extensible.

La función parse genera una estructura de árbol en memoria con los objetos encontrados en el archivo de origen siguiendo la jerarquía y devuelve la raíz del árbol generado.

Translator

Generalización de los conversores de cada objeto del metamodelo a BPMN. Son invocados específicamente para convertir un objeto determinado del origen.

Consta de dos funciones que permiten traducir la lógica y el diagrama por separado según lo especifica BPMN.

PluginManager

Clase encargada del manejo de los plugin. Cuenta con funciones para registrar un nuevo plugin y encontrarlo una vez instalado.

La función scan busca un plugin en disco y lo carga en el sistema.

Diseño de Subsistemas

El sistema no cuenta con subsistemas. Es una única unidad

Subsistemas Específicos

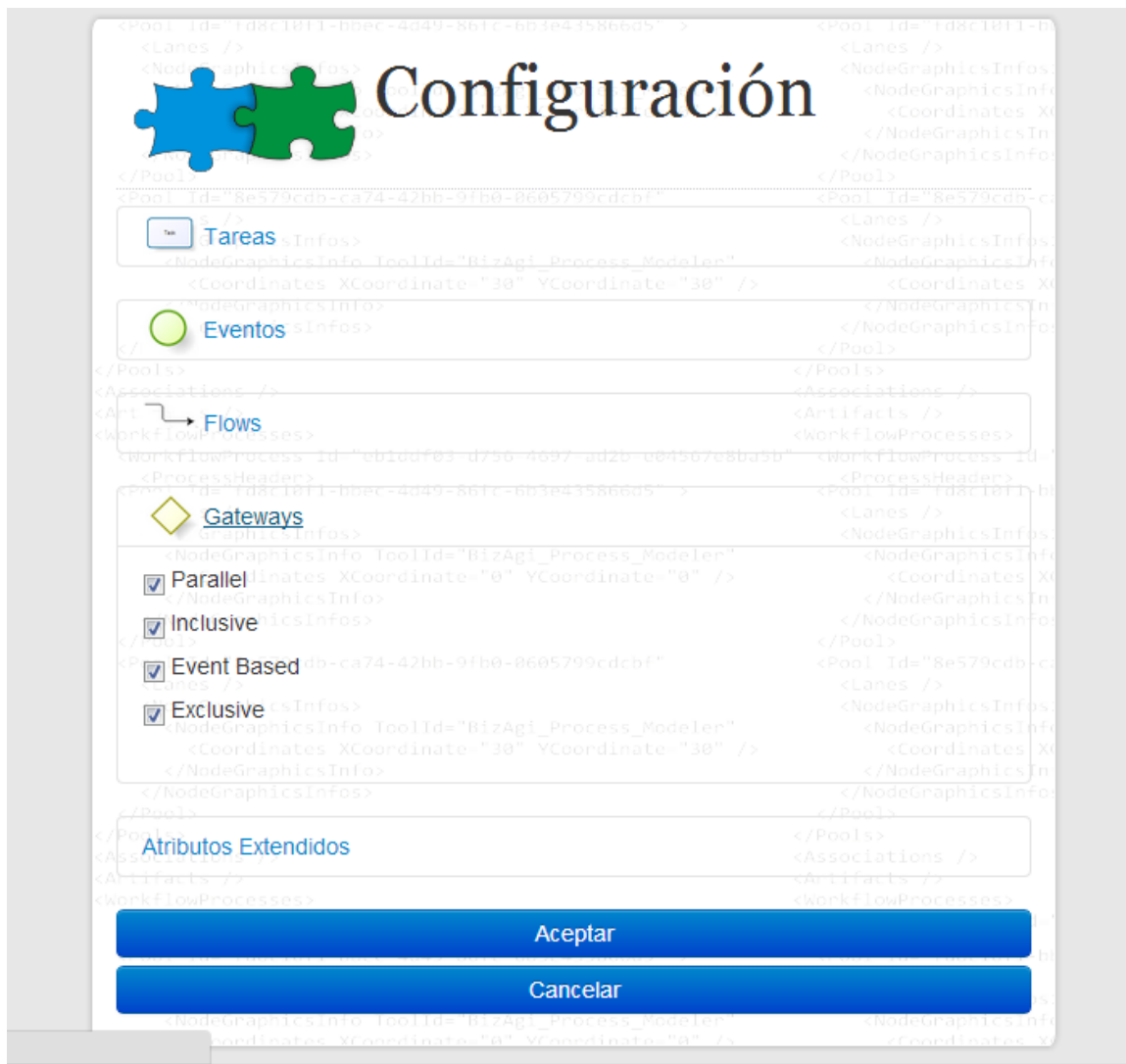
N/A

Revisión de la Interfaz de Usuario

Formatos individuales de interfaz de pantalla

Se cuenta con una única pantalla para la realización del caso de uso.





Controles y elementos de diseño de interfaz de pantalla

Se incorpora 1 campo para permitir seleccionar la ruta del archivo de entrada y dos botones; uno "Aceptar" que permite iniciar la conversión y otro "Configurar" que despliega una lista con scroll de categorías de elementos que se expanden a elección para mostrar los elementos de esa categoría y permitir seleccionarlos para ser convertidos o no. . El campo para la ruta del archivo de entrada cuenta con un botón que permite navegar por el equipo del usuario para especificar dichas rutas.

Forma de navegación de interfaz de pantalla

N/A

Formatos de impresión

N/A

Diseño de Datos

Modelo de Datos del Núcleo

N/A

Modelo de Datos General

N/A

Especificación de la Distribución de Datos

N/A