Documento de especificación de requisitos

Índice

[1.](#_heading=h.gjdgxs) Introducción 2

[1.1.](#_heading=h.30j0zll) Objeto 2

[1.2.](#_heading=h.1fob9te) Ámbito de la aplicación 2

[2.](#_heading=h.3znysh7) Aspectos generales del proyecto 3

[2.1.](#_heading=h.2et92p0) Objetivos del sistema 3

[2.2.](#_heading=h.tyjcwt) Descomposición en subsistemas 4

[3.](#_heading=h.3dy6vkm) Catálogo inicial de requisitos 5

[3.1.](#_heading=h.1t3h5sf) Requisitos funcionales 5

[3.1.1.](#_heading=h.4d34og8) Subsistema de visualización y edición de diagramas 5

[3.1.2.](#_heading=h.2s8eyo1) Subsistema de transformación 6

[3.1.3.](#_heading=h.17dp8vu) Subsistema de análisis 6

[3.2.](#_heading=h.3rdcrjn) Requisitos no funcionales 7

[3.2.1.](#_heading=h.26in1rg) Permisos 7

[3.2.2.](#_heading=h.lnxbz9) Usabilidad 7

[3.2.3.](#_heading=h.35nkun2) Rendimiento 7

# Introducción

## Objeto

Este documento tiene como objetivo describir los requisitos funcionales y no funcionales del proyecto “NOMBRE DEL PROYECTO”, los cuales tendrán que cumplir para su realización. En base a dichos requisitos, se desarrollará una aplicación dividida en subsistemas que incluirán toda funcionalidad propuesta.

El producto software consistirá en un plugin para Eclipse para la creación y análisis de sistemas con variabilidad utilizando diagramas Unified Model Language (UML) o Business Process Model and Notation (BPMN); para el análisis, será necesaria la transformación de estos diagramas a redes de Petri estocásticas temporizadas.

Para el desarrollo de este proyecto, se seguirá un modelo de ciclo de vida incremental e iterativo en el que se distinguirán tres incrementos base, desarrollando cada uno de los subsistemas propuestos. Estos incrementos contarán con las fases de análisis de requisitos, diseño, implementación, integración y pruebas. Cabe destacar que, como todo el proyecto será hecho por una sola persona, las tareas no estarán solapados y cada una de ellas empezará una vez termine la anterior.

Teniendo en cuenta esto, en la primera sección de este documento se propone una descripción de alto nivel relacionada con el ámbito y alcance que ocupará este Trabajo de Fin de Máster. Se propondrán una serie de funcionalidades que debe tener la aplicación, y una variedad de elementos que no la compondrán con el objetivo de marcar los límites de la aplicación.

En la segunda sección define el proyecto y da una descripción profunda de cada uno de los sistemas que lo componen y sus funcionalidades.

Por último, en el tercer apartado se expone un catalogo que acoge todos los requisitos funcionales y no funcionales que conformarán este proyecto.

## Ámbito de la aplicación

El proyecto será un plugin que será compatible con el entorno de programación Eclipse (versión de marzo de 2021) con las siguientes funcionalidades:

* Generación de diagramas BPMN con variabilidad.
* Generación de diagramas UML con variabilidad.
* Generación de redes de Petri.
* Transformación de diagrama BPMN a red de Petri.
* Transformación de diagramas UML a red de Petri.
* Análisis de redes de Petri.
* Almacenamiento de los análisis realizados.
* Análisis de diagramas BPMN en base a análisis de redes de Petri.
* Análisis de diagramas UML en base a análisis de redes de Petri.

El proyecto no incluirá las funcionalidades:

* Generación de diversos tipos de diagramas en el mismo esquema.
* Transformación de red de Petri a diagramas UML.
* Transformación de red de Petri a diagrama BPMN.
* Análisis de forma directa sobre diagramas UML.
* Análisis de forma directa sobre diagramas BPMN.

# Aspectos generales del proyecto

En esta sección se analizarán los objetivos principales del proyecto, para lo cual se describen las necesidades que han dado lugar a la realización del mismo. A continuación, se describe su relación con otros sistemas y, por último, se analizarán los subsistemas en que se divide la aplicación mediante una descripción detallada.

## 2.1. Objetivos del sistema

Este Trabajo de Fin de Máster propone la generación de una serie de módulos y *plugins* para el análisis de variabilidad de UML y BMPN basado en Líneas de Producto de Redes de Petri. Ya que, hasta donde somos capaces de conocer a día de hoy, no existe una herramienta que, en base al diseño de diagramas de líneas de productos BPMN o diagramas de líneas de productos UML se haga un análisis extenso de su seguridad y variabilidad de los mismos.

Teniendo en cuenta esto, uno de los *plugins* que compondrá el proyecto, deberá ser una herramienta de diseño que permita la visualización, creación y la edición de diagramas basados en BPMN, UML. Adicionalmente, también se debería permitir la adicción a cualquiera de los diseños de los diagramas de un modelo de variabilidad que sirva para generar líneas de productos en base a los diagramas de tal forma que, cada elemento pueda corresponderse a una *feature* de un *feature model.*

## 2.2. Descomposición en subsistemas

Explicar el sistema completo

Explicar los subsistemas por separado

# Catálogo inicial de requisitos

## Requisitos funcionales

### Subsistema de visualización y edición de diagramas BPMN con variabilidad (VB)

1. El usuario podrá crear y editar diagramas conformes con el formato BMPN con variabilidad.
   1. Los diagramas creados se almacenarán en formato .bpmn.
   2. Como se almacena la variabilidad???
   3. Qué elementos podemos añadir?
   4. La herramienta para diagramas BPMN permitirá crear objetos de flujo, objetos de conexión, carriles de nado y artefactos.
      1. Dentro de los objetos de flujo encontraremos eventos, actividades y compuertas.
      2. Dentro de los objetos de conexión encontraremos uniones denominadas flujos de secuencia, de mensaje y asociaciones.
      3. Dentro de los carriles de nado se podrán crear piscinas y carriles.
      4. Dentro de los artefactos se encuentran objetos de datos, grupos y anotaciones.
2. Todos los usuarios de la aplicación podrán visualizar los diagramas de forma gráfica.
   1. Existirá restricción de edición si el usuario que ha creado un diagrama considera que otros usuarios no pueden editarlo.

Tienes que separar lo que es BMPN de lo demás… de hecho el subsistema de BPMN es igual que el de UML (solo hay que cambiar una cosa por otra).

### Subsistema de visualización y edición de diagramas UML con variabilidad (VU)

* + - 1. El usuario podrá crear y editar diagramas conformes con el formato UML con variabilidad.
         1. Los diagramas creados se almacenarán en formato .uml
         2. La herramienta para diagramas UML permitirá crear diagramas de clases, de componentes, de estados, de secuencia, de casos de uso y de despliegue.
         3. Dentro de la creación de diagramas de clase, se podrán generar clases, enumeraciones, asociaciones, composiciones, agregaciones, atributos y métodos. Todos los elementos tendrán unas propiedades para su edición.
         4. Dentro de los diagramas de componentes se podrán crear…
         5. Los elementos de variabilidad se corresponden a condiciones de presencia, expresiones y uniones con los elementos de cada diagrama.

Los ficheros donde se guardarán estos elementos serán de extensión .vrb.

### Subsistema de visualización y edición de diagramas Redes de Petri (VPN)

* + - 1. El usuario podrá crear y editar Redes de Petri con variabilidad.
         1. La herramienta para redes de Petri permitirá crear lugares transiciones y arcos
         2. Los elementos de variabilidad se corresponden a condiciones de presencia, expresiones y uniones con los elementos de cada diagrama.

Los ficheros donde se guardarán estos elementos serán de extensión .vrb.

### Subsistema de transformación

1. Todos los usuarios podrán acceder a las transformaciones de diagramas a pesar de que no sean especialistas programadores.
2. Las transformaciones que se pueden hacer serán desde BPMN a redes de Petri y desde UML a redes de Petri.
   1. La transformación de BPMN consistirá en la transformación directa de un diagrama BPMN a una red de Petri.
   2. La transformación de UML constará de la generación de todos los tipos de diagramas de UML (clases, componentes, estados, secuencia, casos de uso y despliegue) y serán transformados en una sola red de Petri.
3. Los ficheros de entrada para poder ejecutar la transformación serán de los tipos correspondientes a BPMN y UML.
   1. Los diagramas BPMN de entrada serán del formato .bpmn.
   2. Los diagramas de UML de entrada serán del formato .uml.
   3. Los diagramas de salida, serán redes de Petri guardadas en el formato .petrinets.
4. Las transformaciones se ejecutarán mediante un *plugin.*
   1. Cuando se utilice, este mostrará un menú donde el usuario elegirá qué tipo de transformación desea hacer.
      1. Diagrama BPMN - redes de Petri
      2. Diagramas UML - redes de Petri
   2. Una vez escogida la transformación, se mostrará otro menú de elección con todos los archivos del formato de entrada a la transformación (si se escoge la transformación de diagramas BPMN a redes de Petri, los archivos que se podrán escoger serán de extensión .bpmn).
   3. El fichero resultante de la transformación se guardará con el mismo nombre del primer fichero de entrada a la transformación.

### Subsistema de análisis

1. Cualquier usuario podrá acceder al análisis de las redes de Petri debido a que, cualquier usuario que haya creado un diagrama debe poder saber si el análisis de la misma da un resultado razonable.
2. Este subsistema tendrá como entrada archivos que contendrán redes de Petri.
3. El resultado de todo el análisis hecho sobre la red de Petri que trate, será guardado en un archivo en formato .txt.
4. Este subsistema dentro de la plataforma de eclipse, será creado como un *plugin* donde se preguntará al usuario cual es el fichero que se desea analizar, mostrando una lista de ficheros en el espacio de trabajo de la extensión .petrinets.
   1. Una vez escogido el archivo, se hará un análisis detallado acerca de…

## 3.2. Requisitos no funcionales

### 3.2.1. Interfaz y Usabilidad

1. La aplicación tendrá soporte en el entorno de programación de Eclipse, en la versión de marzo de 2021. Se entiende que el plugin será compatible con versiones posteriores siempre.
2. El interfaz de usuario de la aplicación deberá ser amigable y fácil de usar, intentando ser distinguibles entre sí mediante agrupaciones de tipo.
3. El interfaz de usuario de los menús de transformación y análisis será lo más simple posible teniendo en cuenta los elementos que han de mostrar.

### Permisos

1. Todos los usuarios tendrán acceso a cada uno de los elementos que conformarán el software resultante de este proyecto.
   1. Todos los usuarios podrán crear, visualizar y editar elementos de cualquier tipo.
   2. Todos los usuarios podrán utilizar la herramienta de transformación de diagramas BPMN y UML en redes de Petri.
   3. Todos los usuarios podrán acceder al uso del análisis de redes de Petri.
2. Un usuario podrá restringir el acceso a la edición de los ficheros que ha creado mediante la herramienta.

### 3.2.3. Rendimiento

1. Las transformaciones de otros diagramas a redes de Petri no durarán más de 30 segundos.

### 3.2.4. Disponibilidad/Seguridad/Privacidad

No hace falta

Idiomas de la aplicación?

Es una aplicación de escritorio??? Hay que explicarlo al decir que es un plugin de eclipse.