



INSTITUTO TECNOLÓGICO DE IZTAPALAPA

PETRI NETS

NOMBRE DE LA CARRERA

ING. SISTEMAS COMPUTACIONALES

PRESENTA:

FRANCISCO FERNANDO MARIA DEL CARMEN - 171080147

ASESOR:

ABIEL TOMAS PARRA HERNANDEZ

CIUDAD DE MÉXICO

JUNIO / 2021





Índice

Resumen.....	4
Introducción	5
Objetivos	6
General	6
Específicos	6
Justificación	7
Marco teórico.....	8
Metodología	12
Desarrollo.....	14
Resultados	23
Conclusiones	26
Referencias.....	27





Índice de figuras

Figura 1	9
Figura 2	9
Figura 3	10
Figura 4	10
Figura 5	10
Figura 6	11
Figura 7	14
Figura 8	14
Figura 9	15
Figura 10	15
Figura 11	16
Figura 12	16
Figura 13	17
Figura 14	17
Figura 15	18
Figura 16	18
Figura 17	19
Figura 18	19
Figura 19	20
Figura 20	20
Figura 21	21
Figura 22	21
Figura 23	22
Figura 24	23
Figura 25	23
Figura 26	24
Figura 27	24
Figura 28	25
Figura 29	25





Resumen

Este proyecto trata de explicar brevemente lo que es Petri y sus reglas, así como demostrar un uso general y su gran utilidad en los proyectos en la vida cotidiana, y su importancia en los proyectos de desarrollo y su relevancia en la implementación de autómatas





Introducción

Petri es una herramienta muy útil en el desarrollo de nuevos proyectos ya que permite ver el paso a paso de cada proceso en el proyecto, es por eso que en capítulo uno se plantean los objetivos a realizar durante este proyecto.

En el capítulo dos se trata la justificación del proyecto con la cual se da la relevancia de este.

En el capítulo tres se encuentra el marco teórico donde se dan las reglas y normas que rigen a Petri, así como un poco de su historia y algunas características

En el capítulo cuatro se encuentra la metodología, la cual es la descripción de la metodología que se utilizó al realizar este proyecto.

En el capítulo cinco se ve el desarrollo de este proyecto el cual es básicamente la instalación de la interfaz, así como el desarrollo de un par de ejemplo de redes Petri

En el capítulo seis se encuentran los resultados obtenidos en unos ejemplos más, esto es para demostrar la utilidad de Petri

En el último capítulo se dan las conclusiones obtenidas en la realización de este proyecto





Objetivos

General

Comprensión y análisis de redes usando las herramientas graficas Petri en un modelo de sistema discreto.

Específicos

1. Enunciar conceptos básicos de redes Petri
2. Comprender y practicar las reglas de marcado
3. Identificar y defender las estructuras de Petri
4. Comprender y realizar validación con Petri





Justificación

Petri es un sistema desarrollado en 1970 utilizado para la comunicación con autómatas por la universidad de Bonn, es utilizada para describir informalmente cualquier red que pueda ser representada gráficamente como con algún diagrama de flujo y esta es muy importante en el área de desarrollo de software ya que es una herramienta de validación que puede ser aplicada en distintas áreas del desarrollo del software.

La importancia de Petri es tan grande que hoy en día es utilizado en evaluación de rendimiento, protocolos de comunicación, modelado y análisis de sistemas distribuidos (sistema de base de datos distribuidas, programas paralelos y concurrentes).

Con Petri el diseñador puede plasmar su creación documentando en forma simultánea todos los resultados conforme se van desarrollando lo que posibilita que pueda capturarse no únicamente el diseño del autómata si no también la razón “del cómo” y “por qué” del diseño en sí.

Por todo lo descrito anteriormente es de vital importancia el aprendizaje e implementación de redes Petri en el sistema educativo actual.



Marco teórico

El concepto de red de Petri apareció en 1962 con la tesis doctoral de Carl Adam Petri "Comunicación con Automata" en la Universidad de Bonn. A partir de entonces se difundió en Europa y Estados Unidos, y ya en 1970 aparecieron grupos de investigación que incorporaban las redes de Petri a sus trabajos y/o que se dedicaban a estudiar sus propiedades. Con todo el trabajo acumulado se crearon ciclos de conferencias, libros, actas y revistas en esta área. Aunque no hay memorias y actas de todas las conferencias, los artículos más importantes de éstas, se condensaron en varios libros y revistas.

Las redes de Petri se pueden incorporar informalmente en cualquier área o sistema que pueda describirse gráficamente como diagrama de flujo y que necesitan algunos medios de representar actividades paralelas o concurrentes. Particularmente, en el área de desarrollo de software, las redes de Petri son una herramienta de validación que puede aplicarse en distintas etapas en el desarrollo de sistemas.

Sin embargo, el punto débil de las redes de Petri radica en la complejidad del problema, esto es, los modelos basados en redes de Petri, siempre tienden a ser muy grandes para su análisis. Con el fin de solucionar este problema se han desarrollado técnicas de reducción y extensiones a las redes de Petri. Por lo general, para aplicar las redes de Petri a un problema, se le realizan modificaciones o restricciones.

Una red de Petri es un grafo orientado con dos tipos de nodos: lugares (representados mediante circunferencias) y transiciones (representadas por segmentos rectos verticales).

Los lugares y las transiciones se unen mediante arcos o flechas.

Un arco une siempre lugares con transiciones y nunca dos lugares o dos transiciones.

Una transición puede ser destino de varios lugares y un lugar puede ser el destino de varias transiciones.

Una transición puede ser origen de varios lugares y un lugar puede ser origen de varias transiciones.

Los lugares pueden presentar marcas (una marca se representa mediante un punto en el interior del círculo).

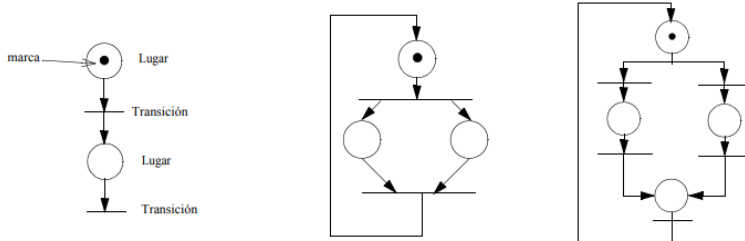


Figura 1

Cada lugar tiene asociada una acción o salida.

Los lugares que contiene marcas se consideran lugares activos.

Cuando un lugar está activo sus salidas están a uno. A las transiciones se les asocia eventos (funciones lógicas de las variables de entrada).

Una transición se dice que está sensibilizada cuando todos sus lugares origen están marcados.

Cuando ocurre un evento asociado a una transición (la función lógica se hace uno), se dice que la transición está validada.

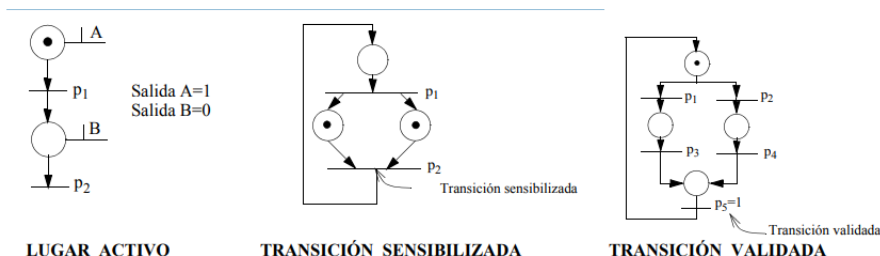


Figura 2

REGLAS DE EVOLUCIÓN DE MARCADO.

El marcado cambia al franquear las transiciones.

Para franquear una transición ha de estar validada y sensibilizada

Cuando una transición se franquea desaparecen las marcas de los lugares origen y se añade una marca a cada uno de los lugares destino

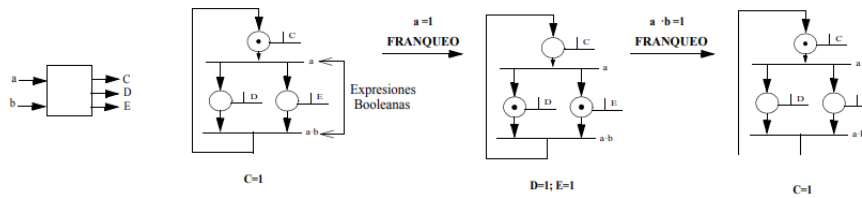


Figura 3

Un lugar puede tener más de una marca, pero cuando dos transiciones que están sensibilizadas a la vez, pueden entrar en conflicto puesto que para que la red sea válida las condiciones de validación t_1 y t_2 no pueden darse a la vez.

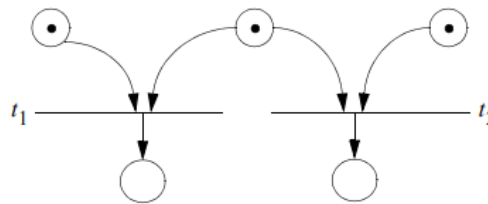


Figura 4

Las estructuras básicas son:

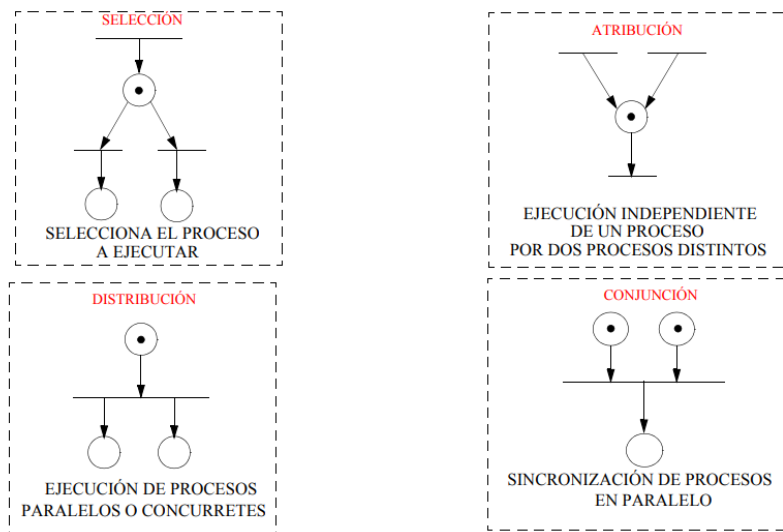


Figura 5

Lugares transiciones, arcos y tokens, en un diagrama que tiene una sintaxis.

- Los lugares son los elementos pasivos de la red de Petri y, junto con los tokens, se utilizan para modelar los estados del sistema.

- Las transiciones son los elementos activos de la red de Petri, y representan las acciones de un sistema. Estas acciones originan cambios en el estado de la red.
- El conjunto de lugares, transiciones y arcos son finitos y estáticos. Lo que indica que el sistema no puede tener más causas y eventos que los que originalmente tiene representados en el modelo.
- El conjunto de tokens y marcas pueden cambiar durante la ejecución de la red, describiendo las características dinámicas del sistema modelado.

Una estrategia para facilitar el análisis, es reducir el sistema a un modelo más simple el cual pueda conservar sus propiedades de análisis. Las técnicas para transformar un modelo abstracto a un modelo más refinado, en una forma jerárquica, se pueden utilizar como síntesis.

Existen muchas técnicas de transformación para redes de Petri. No es difícil ver que las siguientes seis operaciones conservan las propiedades de acotamiento, seguridad y activación

A continuación, se muestran distintas transformaciones que conservan propiedades de análisis según la especificación en cada caso.

- Fusión de una serie de lugares.
- Fusión de una serie de transiciones.
- Fusión de lugares paralelos.
- Fusión de transiciones paralelas.
- Eliminación de auto-ciclos de lugares.
- Eliminación de auto-ciclos de transiciones

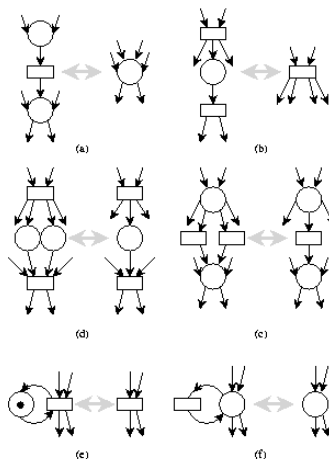


Figura 6

Metodología

Las metodologías ágiles son modelos de gestión de proyectos con raíces en TI y creados en base a la insatisfacción de varios desarrolladores con la falta de fluidez de los modelos de gestión tradicionales. Agile se asocia con una mentalidad productiva y se enfoca en segmentar el alcance de las tareas para anticipar las entregas, generando una percepción del valor del cliente más rápido.

La metodología Agile mantiene la dirección sin caer en la rigidez de los conocidos métodos en cascada o waterfall. Estos planean el trabajo desde el principio, sin lugar a imprevistos. De forma que cuando aparecen, resulta imposible reaccionar a tiempo. El agilismo, sin embargo, mantiene la capacidad de tomar la mejor opción en cada momento sin comprometer el proyecto. Los métodos Agile más populares del momento son Scrum y Kanban.

Aplicar Agile conlleva cumplir 12 principios:

1. La prioridad es que el cliente esté satisfecho y siempre informado del estado del proceso.
2. Los requisitos del proyecto pueden cambiar y no se verá como un problema, sino como una ventaja competitiva.
3. Las entregas se realizan periódicamente y en periodos cortos. La planificación se realizará desde las dos semanas, a los dos meses.
4. El equipo debe trabajar de forma conjunta y coordinada.
5. Es prioritario motivar al equipo, confiar en los miembros y proporcionarles los recursos o apoyos que necesiten.
6. Las reuniones Scrum son el método más efectivo para comunicarse.
7. El éxito depende de si el producto final funciona y es satisfactorio.
8. Los procesos deben ser sostenibles, tanto en recursos materiales, como en la gestión del tiempo y el ritmo de trabajo.
9. En todo proceso o etapa debe prevalecer la excelencia técnica.
10. Prevalece la ley de la simplicidad: menos, es más.
11. La organización de los equipos es esencial para dar con un buen diseño.
12. Los tiempos para la reflexión y buscar mejoras es necesario e igual que importante que el resto de fases.

Con lo que se obtiene como beneficios

- Reducir riesgos y adaptarse a sorpresas y cambios.
- Hacer tangibles los aspectos del resultado final de la primera semana.
- Transparencia y visibilidad del estado del proyecto.
- Flexibilidad para cambiar requerimientos.



- Entregas de valor constante.
- Visibilidad del compromiso y mejora de la comunicación durante el proceso.
- Anticipación de problemas y mayor agilidad para actuar.

Y es por esto que se ha seleccionado la metodología ágil, puesto que conforme se va realizando este proyecto se tiene que ir observando el resultado puesto que no solo así lo delimita este tipo de metodología si no que es a lo que nos orilla Petri gracias a que conforme se desarrolla más y más el modelado de red, se debe de poder apreciar el progreso pues así lo marca su propia elaboración



Desarrollo

Primero se procede a buscar una interfaz agradable para la realización de redes Petri, en este caso se usó PIPE¹ la cual se puede descargar desde sus colaboradores en este caso fue por GitHub.

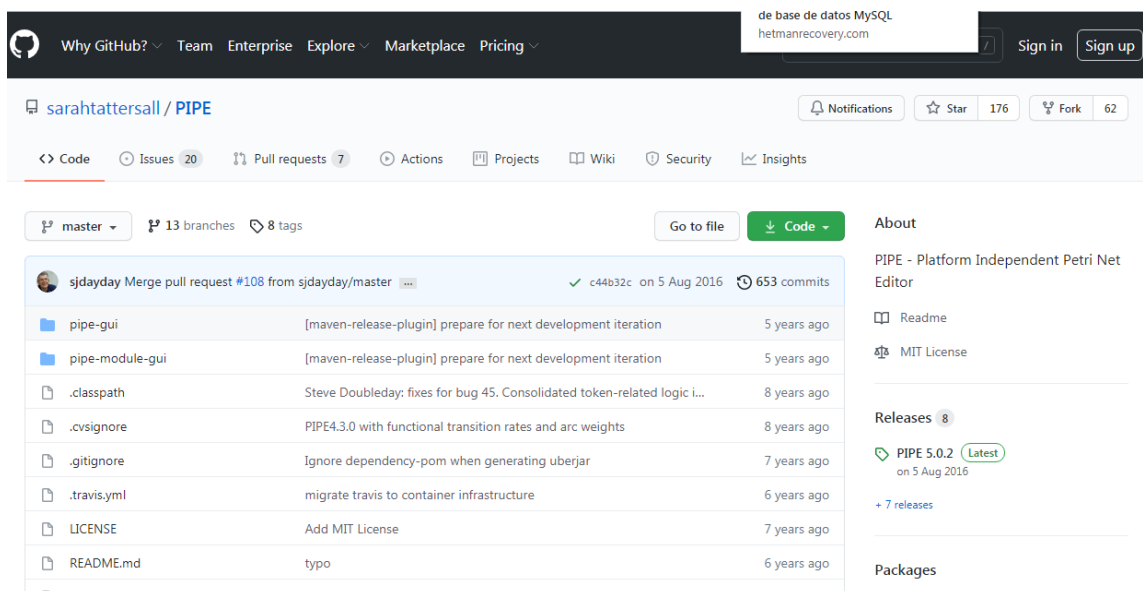


Figura 7

Al descargar este archivo se encontrará con la siguiente carpeta

Nombre	Fecha de modifica...	Tipo	Tamaño
Pipe	06/09/2013 12:45	Carpeta de archivos	
.DS_Store	06/09/2013 12:47	Archivo DS_STORE	7 KB
launch	28/04/2012 21:30	Archivo por lotes ...	1 KB
launch.sh	11/08/2013 20:44	Archivo SH	1 KB
README	06/09/2013 12:42	Documento de tex...	1 KB

Figura 8

Con lo que se tiene que ejecutar el archivo launch con lo que aparece

¹ Platform independent Petri-net editor

```
C:\Windows\system32\cmd.exe

C:\Users\MASTERY\Downloads\PIPEv4.3.0\PIPEv4.3.0>cd Pipe

C:\Users\MASTERY\Downloads\PIPEv4.3.0\PIPEv4.3.0>java -cp .;./lib/jpowergraph-0.2-common.jar;./lib/jpowergraph-0.2-swing.jar;./lib/powerswing-0.3.jar;./lib/drmaa.jar;./lib/hadoop-0.13.1-dev-core.jar;./lib/jcommon-1.0.10.jar;./lib/jfreechart-1.0.6.jar;./lib/jfreechart-1.0.6-swt.jar;./lib/tools.jar Pipe
```

Figura 9

Y solo queda ser paciente ya que unos segundos o minutos (dependiendo de la capacidad de la PC) y entonces aparecerá el editor con el que se trabajó

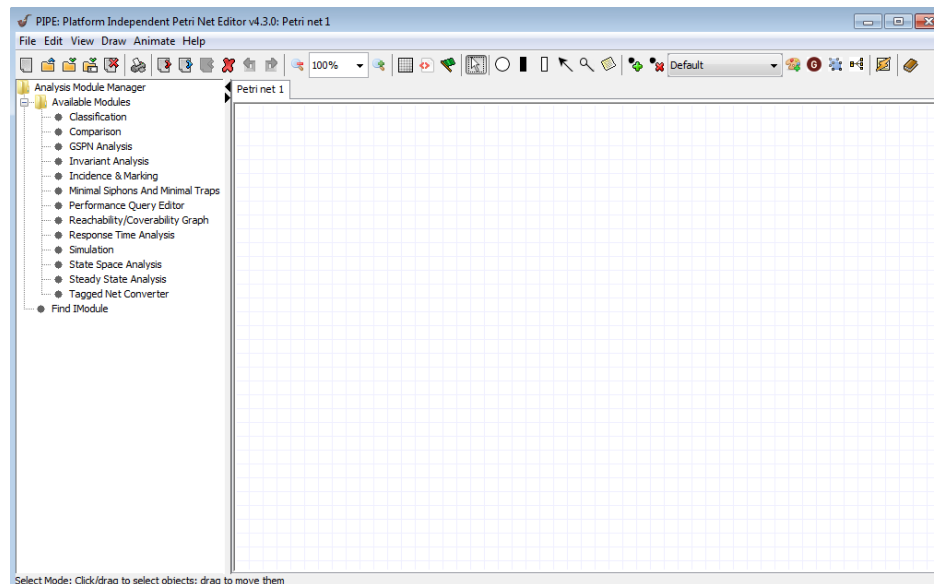


Figura 10

Con el que se pretende desarrollar el modelo Petri del siguiente problema

Tomando un semáforo de tránsito dibuje el diagrama de su red Petri

Paso uno: se colocan los estados del semáforo

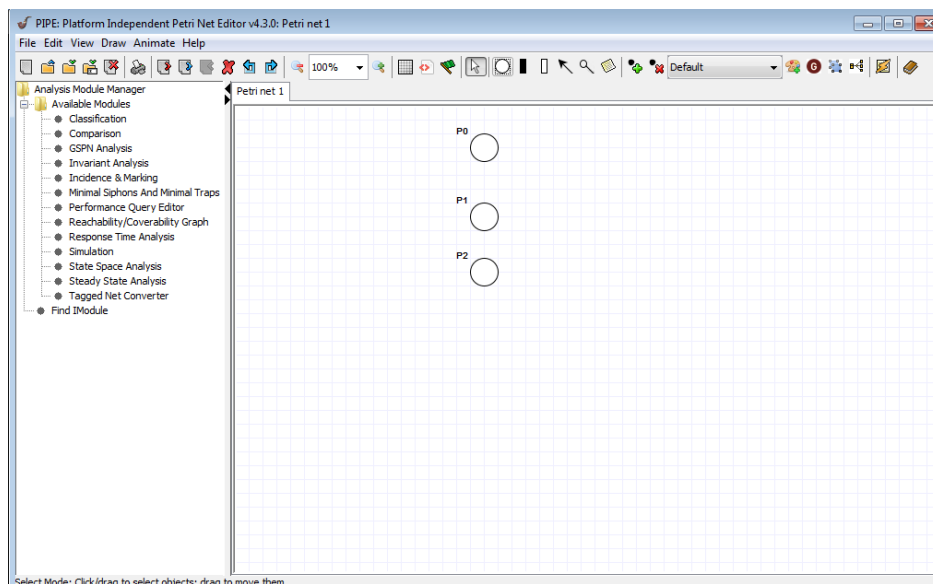


Figura 11

Posteriormente se editan los lugares dándole clic derecho y seleccionando editar lugar

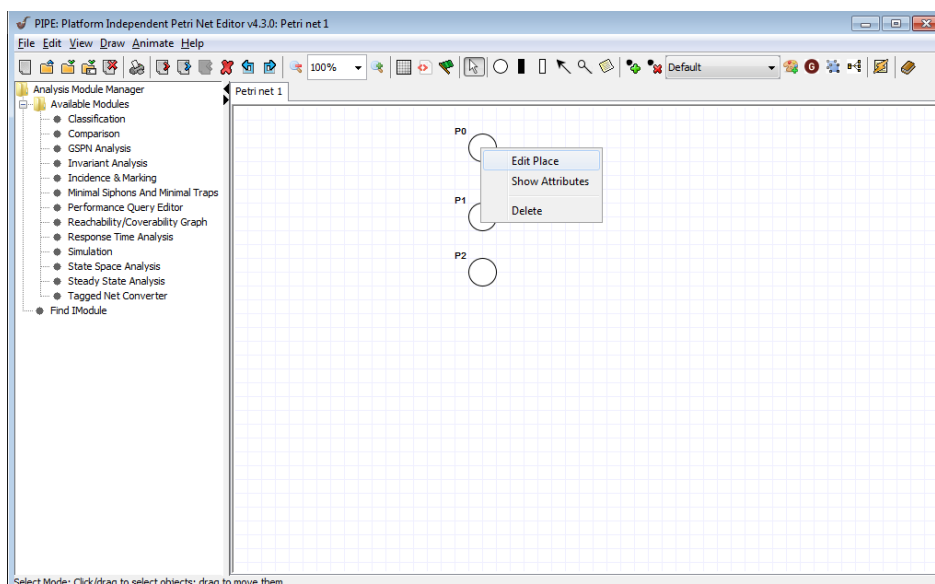


Figura 12

Estas ediciones son colocadas en la siguiente ventana y son personalizadas por cada lugar

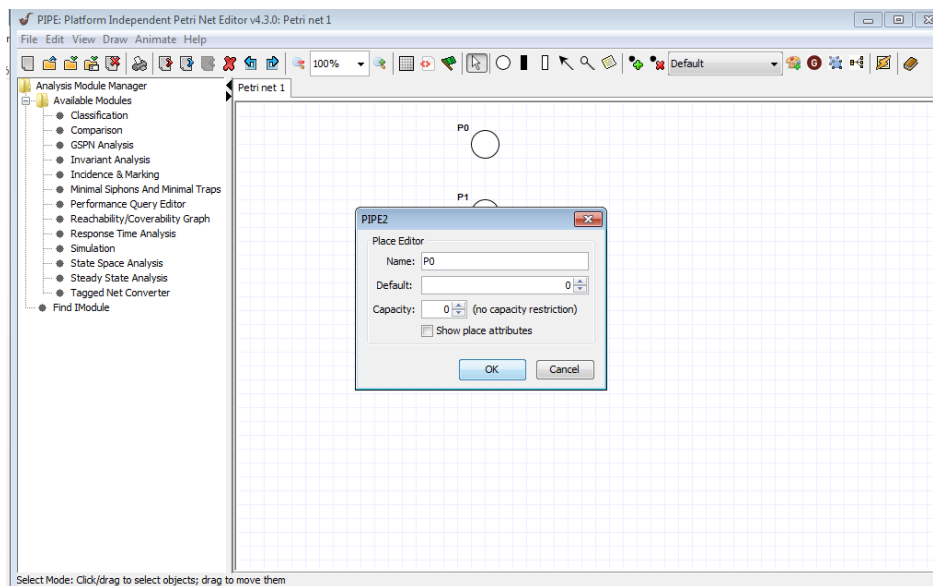


Figura 13

Una vez editados los lugares debe quedar como se muestra en la siguiente figura

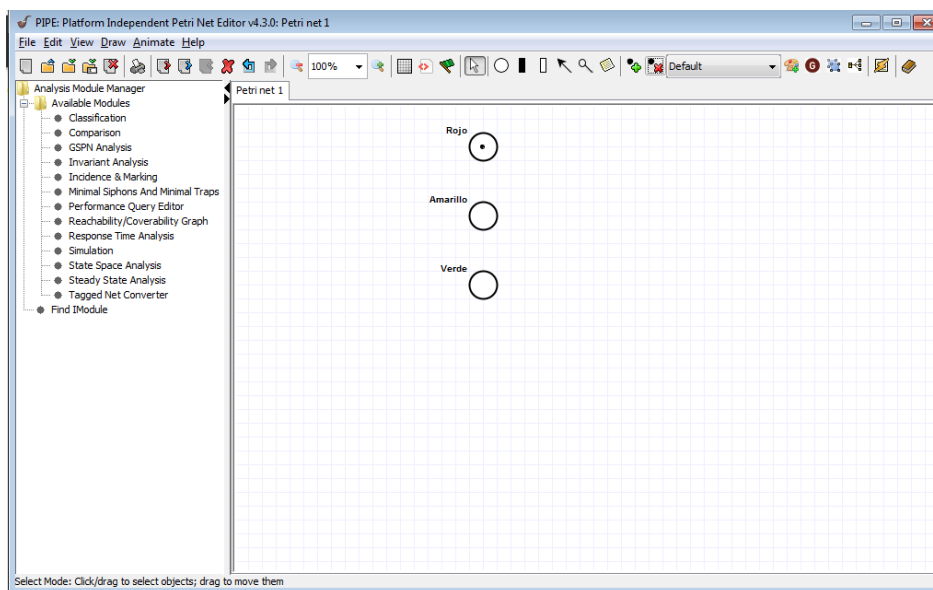


Figura 14

Con los lugares bien colocados se procede a ubicar las transiciones

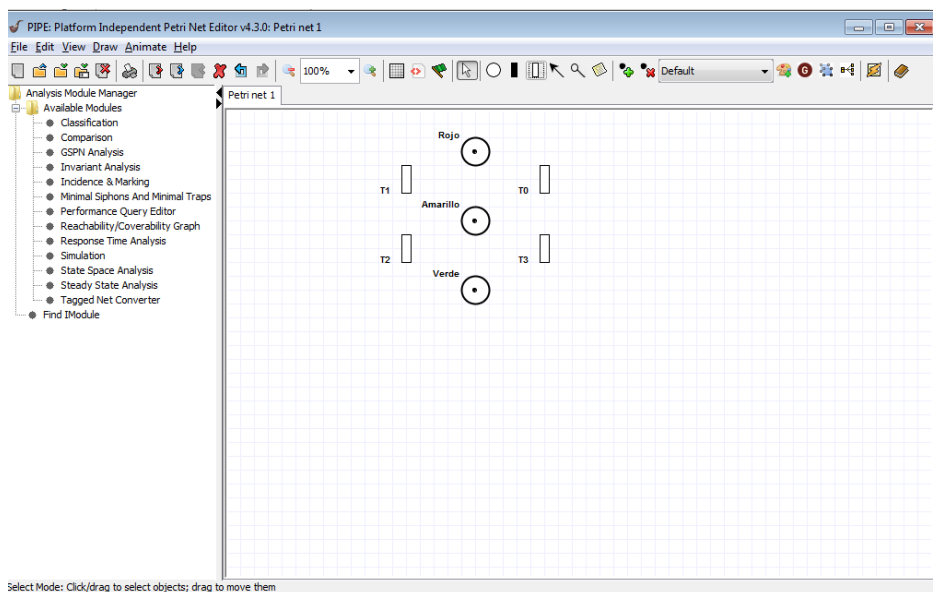


Figura 15

Y se procede a dar clic derecho sobre las transiciones para desplegar una lista de opciones y seleccionar editar transición

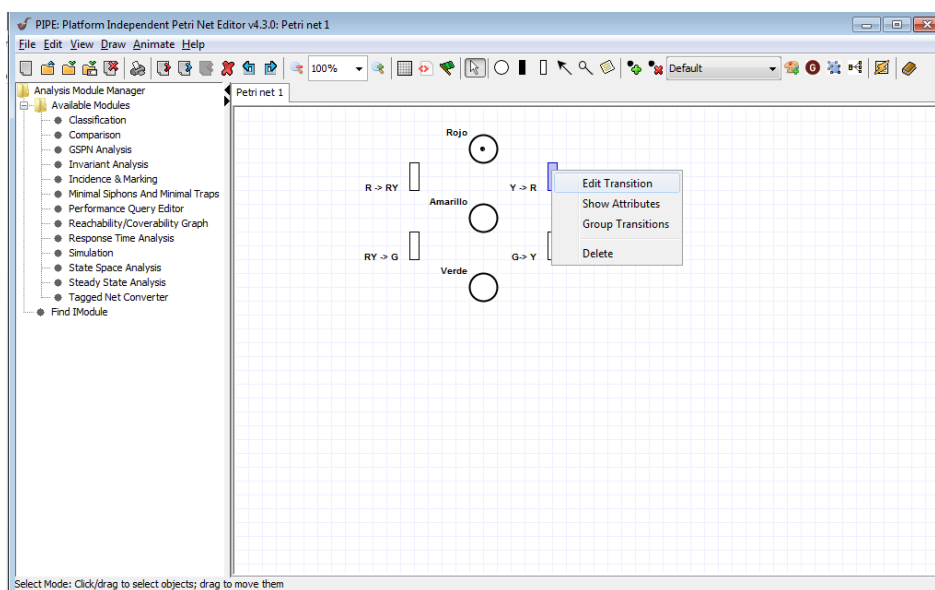


Figura 16

cuando se seleccione editar se mostrará la siguiente pestaña

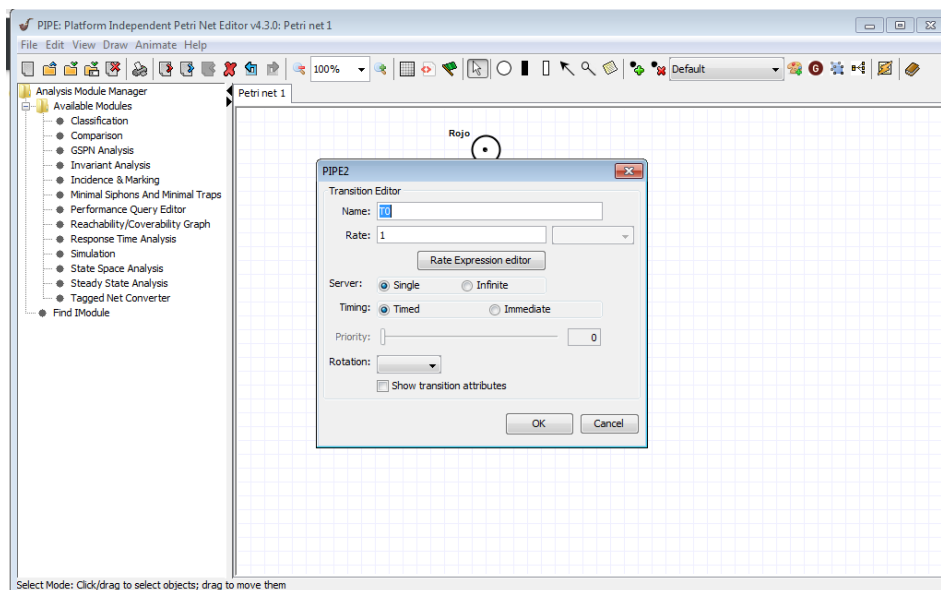


Figura 17

Al tener las transiciones editadas debe quedar algo así

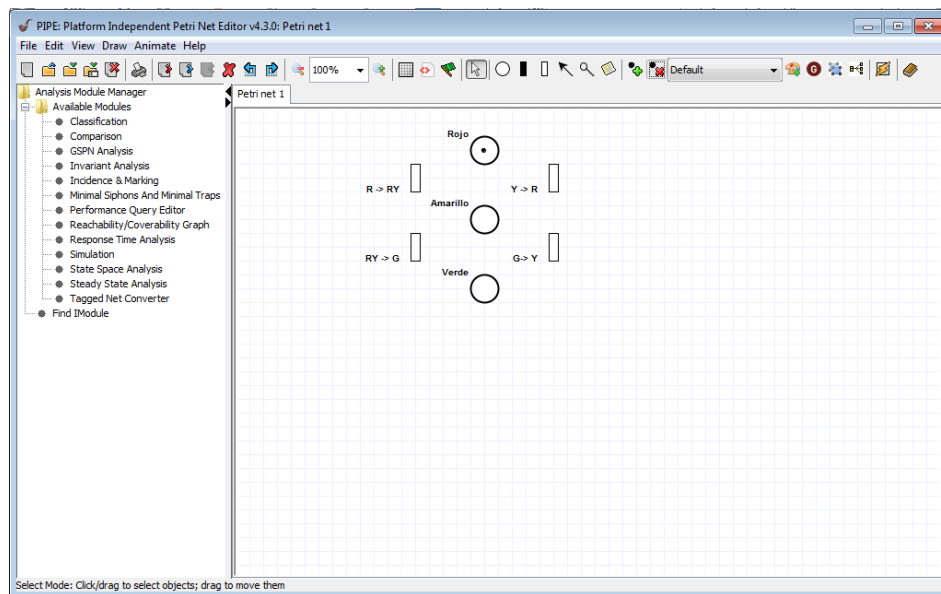


Figura 18

Se colocan las flechas que indican el oren de la transición (arcos)

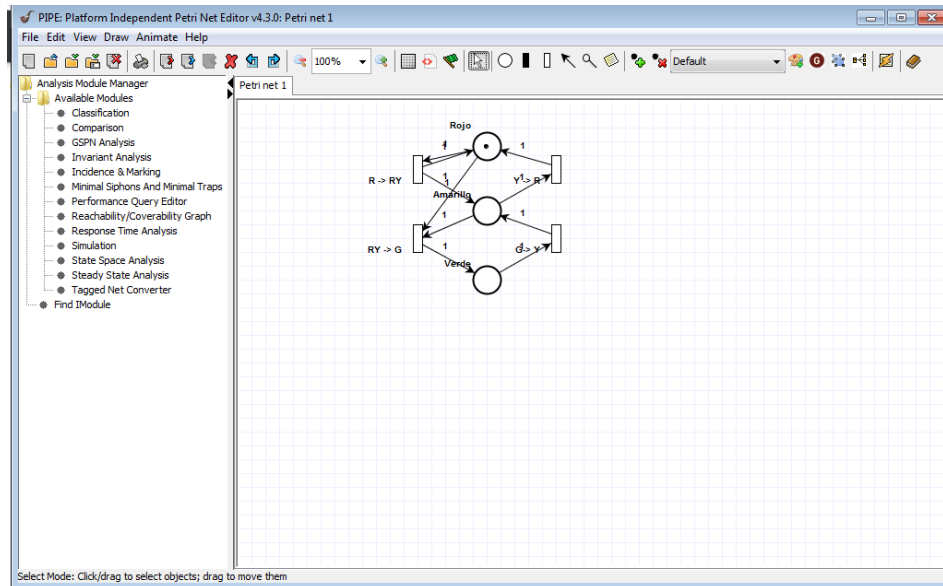


Figura 19

Con el diagrama armado se procede a reacomodar los elementos de manera que sea legible cada elemento en el diagrama

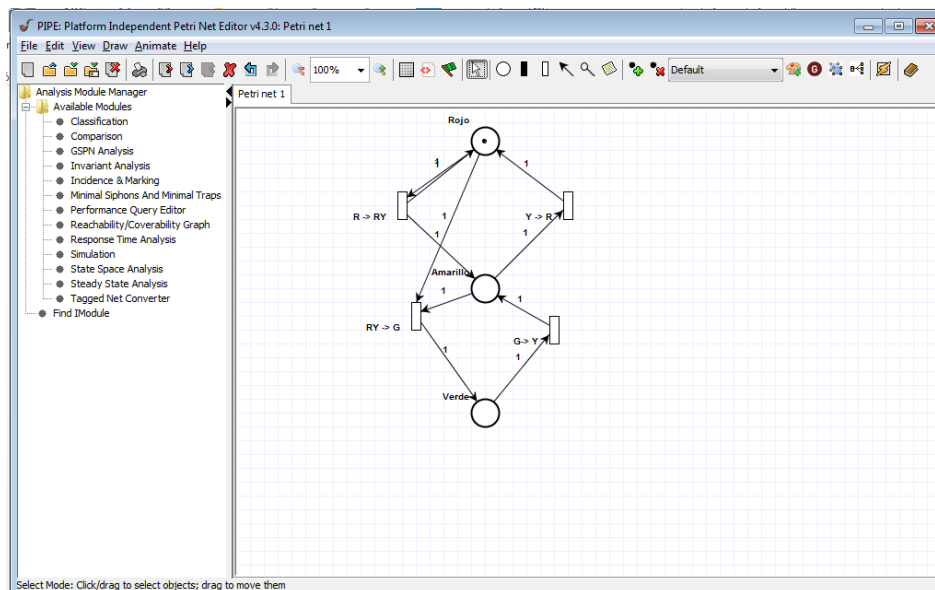


Figura 20

Una vez acomodado se verifica el correcto funcionamiento de esta red diseñada, esto se hace iniciando el modo animación

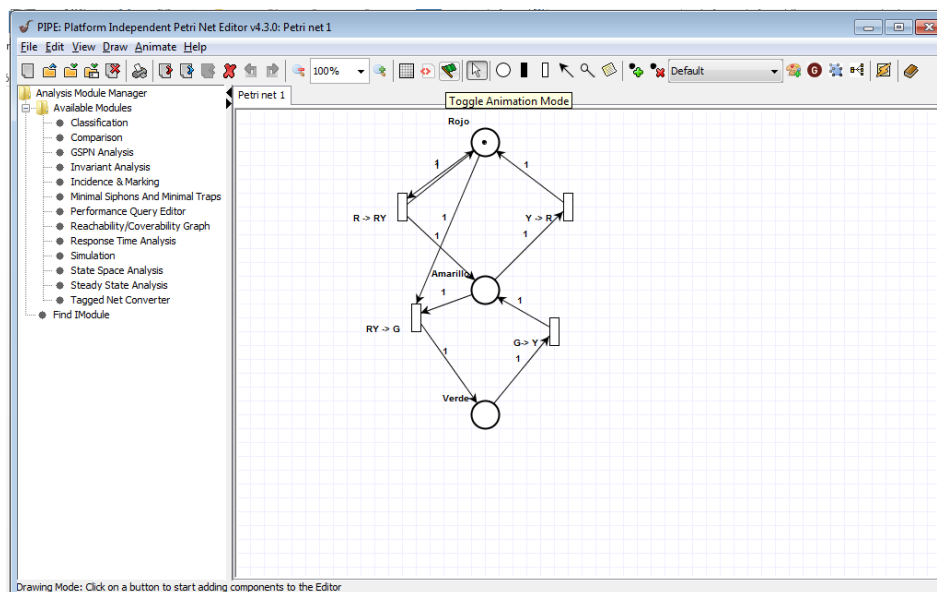


Figura 21

Para que la red Petri cumpla su función se debe de correr las transiciones

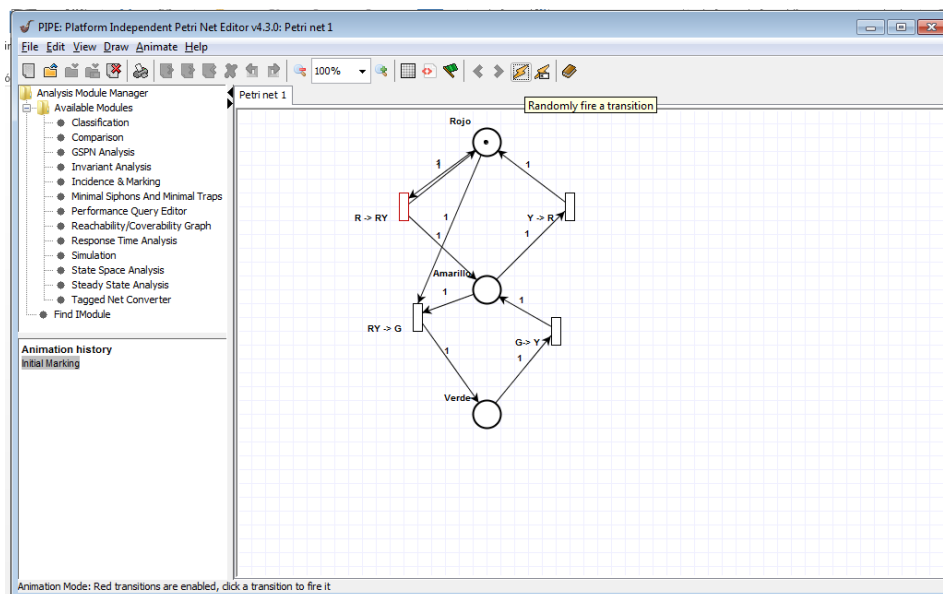


Figura 22

Del lado inferior izquierdo aparece el resultado de cada transición

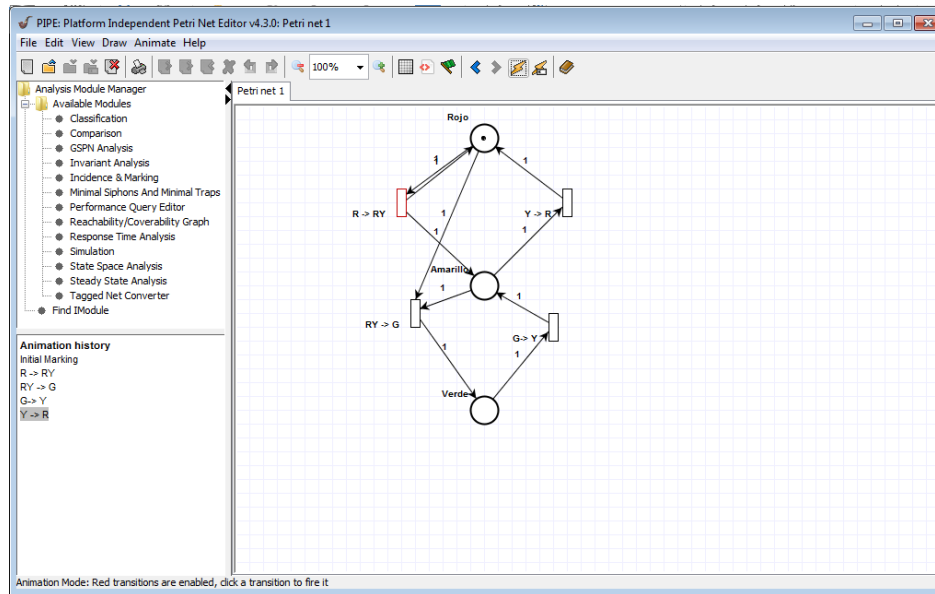


Figura 23

Resultados

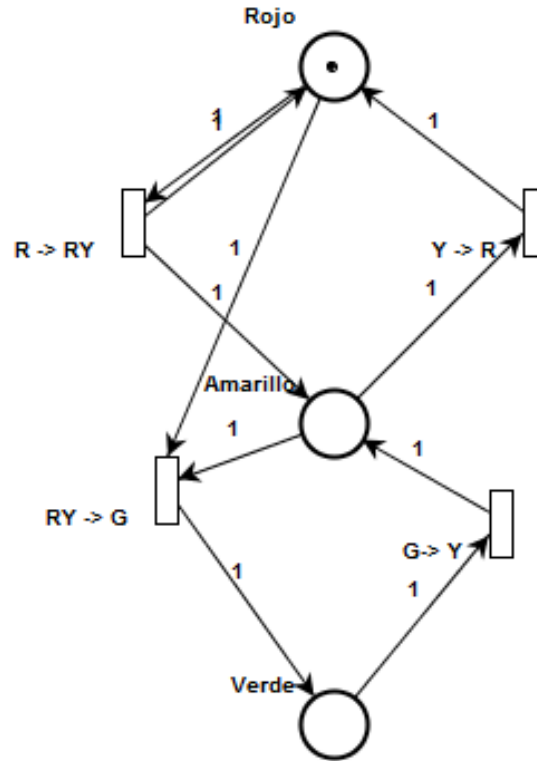


Figura 24

Animation history

Initial Marking

R -> RY

RY -> G

G -> Y

Y -> R

R -> RY

RY -> G

Figura 25

Cuatro filósofos sentados en una mesa redonda con un tenedor en medio de cada par desean comer espagueti de un plato que se encuentra en medio de la mesa

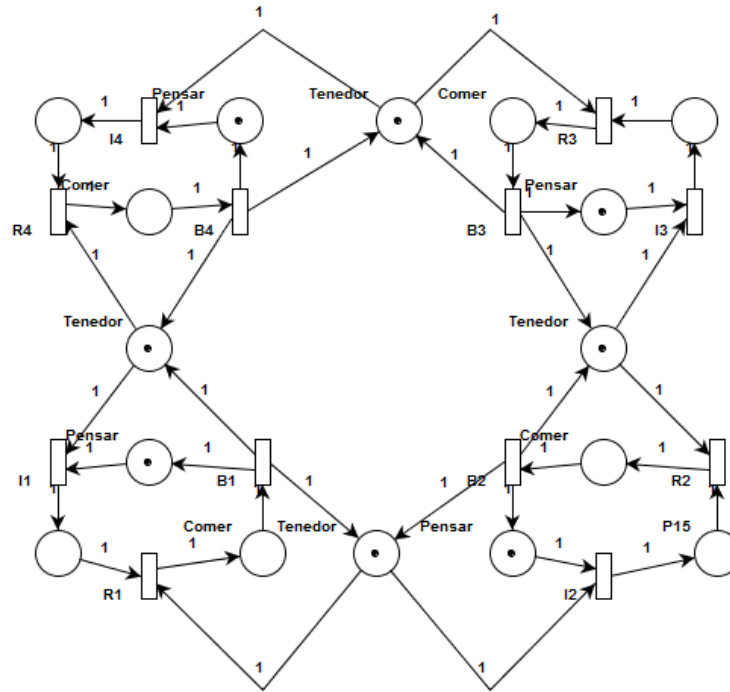


Figura 26

Animation history

Initial Marking

I1
I4
R1
B1
R4
I2
I3
B4
I4
R4
B4
I1
R3
B3
I3
I4

Figura 27

Un acceso concurrente a los datos mediante más procesos utilizando un semáforo. El semáforo está modelado por un lugar cuya marca representa el hecho de que los datos son accesibles. El primer uso de cualquier archivo elimina el token del semáforo que bloquea cualquier acceso a los datos de otros procesos. La eliminación del token del semáforo reserva todos los datos para el proceso exitoso que evita los puntos muertos.

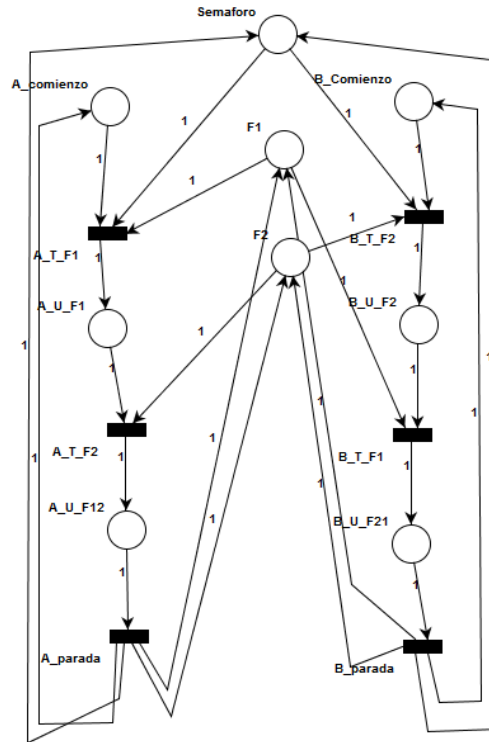


Figura 28

Animation history

Initial Marking

A_T_F1
A_T_F2
A_parada
B_T_F2
B_T_F1
B_parada
A_T_F1
A_T_F2
A_parada
B_T_F2
B_T_F1
B_parada
A_T_F1
A_T_F2
A_parada
B_T_F2
B_T_F1
B_parada

Figura 29



Conclusiones

Petri net es una herramienta muy útil al momento de contemplar los resultados de un proyecto normalmente de desarrollo de software esto es debido a que permite ver el progreso paso a paso del proyecto en sí y predecir el resultado, además de ser una herramienta fácil de utilizar aunque con las debidas precauciones ya que se deben de respetar las normas y reglas de Petri net, sin embargo su aplicación en sistemas grandes puede llegar a ser muy costosa esto debido a su complejidad ya que como se menciono es paso a paso del proceso



Referencias

Home. (s/f). Recuperado el 22 de junio de 2021, de Petrinet.org website: <http://petrinet.org>

Latorre-Biel, J.-I., & Jiménez-Macías, E. (2019). Petri Net Models Optimized for Simulation. En E. Abu-Taieh & A. A. E. S. Ahmed (Eds.), Simulation Modelling Practice and Theory. Londres, Inglaterra: IntechOpen.

López-Neri, E. (s/f). Modelado de Sistemas con Redes de Petri 3-LNS. Recuperado el 22 de junio de 2021, de Unam.mx website: <https://www.ingenieria.unam.mx/sistemas/PDF/Avisos/Seminarios/Seminario%20IV/5%20Modelado%20Redes%20de%20Petri.pdf>

Metodología Agile. (2019, octubre 11). Recuperado el 22 de junio de 2021, de Mjvinnovation.com website: <https://www.mjvinnovation.com/es/metodologia-agile/>

Petri.Net. (s/f).

Villán, V. R. (2019). Las metodologías ágiles más utilizadas y sus ventajas dentro de la empresa. Thinking for Innovation. Recuperado de <https://www.iebschool.com/blog/que-son-metodologias-agiles-agile-scrum/>

(S/f-a). Recuperado el 22 de junio de 2021, de Repositorioinstitucional.mx website: <https://ciateq.repositorioinstitucional.mx/jspui/bitstream/1020/61/1/BarrazaNavarroFranciscoJ%20CtAutSistDinamicos%202013.pdf>

(S/f-b). Recuperado el 22 de junio de 2021, de Edu.ec website: <http://repositorio.unemi.edu.ec/bitstream/123456789/3961/1/ESTUDIO%20DE%20LAS%20REDES%20DE%20PETRI%20EN%20EL%20MODELADO%20Y%20SIMULACIÓN%20DE%20SISTEMAS%20DE%20EVENTOS%20DISCRETOS.pdf>

(S/f-c). Recuperado el 22 de junio de 2021, de Cinvestav.mx website: <http://computacion.cs.cinvestav.mx/~ameneses/pub/tesis/mtesis/node5.html>

(S/f-d). Recuperado el 22 de junio de 2021, de Unam.mx website: http://fcaenlinea1.unam.mx/anexos/1566/1566_u6_act2.pdf

CPN Tools – A tool for editing, simulating, and analyzing Colored Petri nets. (s/f). Recuperado el 22 de junio de 2021, de Cpntools.org website: <http://cpntools.org>