



Trabajo de final de grado

**Minería de procesos en el contexto médico: Análisis de algoritmos a través de
Healthcare System Specifications**

**Process mining in medical context: Analysis of algorithms through Healthcare
System Specifications**

Autor

David Ubide Alaiz

Directores

Cristian Mahulea

Jorge Albareda Albareda

Grado en Ingeniería Informática

Escuela de Ingeniería y Arquitectura

2019-2020

Agradecimientos

Me gustaría agradecer a mis directores, Cristian Mahulea y Jorge Albareda por su dedicación ,entusiasmo e implicación en el desarrollo a pesar de las circunstancias y de la época que pasamos.

A Raúl Lorenzo López, médico residente del hospital, por su implicación en la implementación del proyecto y su dedicación en la elaboración de la parte médica.

Por último, agradecer y dedicar este trabajo a Javier Navarro, Eduardo Cebollada, Alba Moreno , Paula Blasco y a la Casa de Juventud de Torrero . Los superhéroes de barrio que consiguen sacar el potencial oculto de jóvenes día a día sin pedir nada a cambio.

Resumen

El problema estudiado en este trabajo nace ante la necesidad de modelización, estudio y generación de vías clínicas en el área de *Cirugía Ortopédica y Traumatología* del Hospital Clínico Universitario “*Lozano Blesa*” de Zaragoza. Ante esta necesidad se decide desarrollar un sistema de información a través de una aplicación web que permita gestionar la información y las necesidades propuestas por el equipo médico basándonos en el diseño arquitectural del *Healthcare System Specification*.

El sistema de modelado de base de datos y el servidor para la captación de datos se desarrolló basándose en tecnologías *Python* a través de una implementación en un servidor *Django* y *SQLite*. Se desarrolló una interfaz amigable para el usuario a través de *Bootstrap Studio*, un software que permite el desarrollo de código *HTML* de forma intuitiva.

En la parte de recuperación y poblado de la información, se precisó de un proceso *ETL* (Extracción, transformación y carga) a través de orígenes de datos en formato físico de las historias clínicas de los pacientes con la colaboración del equipo médico. Todo ello, apoyado en el desarrollo de *scripts Python* para la carga y actualización.

Se desarrolló un modelo matemático de las vías clínicas a través de autómatas finitos probabilistas. Estos autómatas son un tipo de máquina de estados que permite trabajar con probabilidades y que se amolda a las diferentes necesidades, extrayendo los caminos más y menos probables que se pueden dar a la hora de tratar una patología. Esto ha permitido observar, entre otras cosas, el por qué hay pacientes que se quedan ingresados hasta 6 días más de lo que deberían y son informaciones fundamentales en el proceso de decisión para crear nuevas vías clínicas.

Posteriormente, este modelo se transformaría en un modelo de redes de Petri, que permite la simulación de diferentes situaciones hospitalarias a través del software *CPNTools*. Asimismo, este modelo, permite evaluar el comportamiento del sistema hospitalario ante unas determinadas condiciones (número de pacientes, recursos médicos, recursos humanos, etc.) que se establezcan y también se puede utilizar para estudiar propiedades del sistema sanitario utilizando técnicas estructurales.

Finalmente, el sistema de información que se ha desarrollado facilita una serie de indicadores médicos propuestos a través de un panel de mando en *Power BI*. Estos indicadores se obtienen utilizando los datos extraídos de las historias clínicas. Este software permite la interacción entre los diferentes indicadores a través de los filtros que se le apliquen. Todo ello, se ha encapsulado en un despliegue en la nube a través del uso de las máquinas de *Google Cloud* siendo el resultado observable en la siguiente dirección:

<http://www.hlozablesa.tk>¹

¹ Credenciales:

https://docs.google.com/document/d/1FteBF10tZcX44K_Ecw_8NHqf1HMT1KbFaeY_4r4sP-M/edit?usp=sharing

Índice

Resumen	4
1. Introducción	8
1.1 Dominio del problema	8
1.2 Objetivos del TFG	9
1.3 Técnicas utilizada para su resolución	10
1.4 Alcance	11
1.6 ¿Qué es HSS?	11
1.6.1 Análisis de de sistema hospitalario con HSS	13
1.7 Estado del arte de la modelización de procesos	14
1.7.1 Autómata finito probabilista	14
1.7.2 Redes de Petri	15
2. Implementación	16
2.1 Requisitos de la aplicación	16
2.2 Decisiones de implementación	17
2.3 Modelo de la base de datos	18
2.3.1 Modelado y extracción de datos	20
2.3.2 Transformación y carga de datos	22
2.4 Desarrollo del aspecto visual	25
3. Análisis de datos	29
3.1 Autómatas finitos probabilísticos	29
3.2 Redes de Petri Coloreadas	35
3.3 Panel de mando	37
4. Despliegue	41
5. Problemas encontrados	42
6. Resultados	43
7. Conclusiones	45
8. Bibliografía	47
Anexos	48
Anexo 1: Vías clínicas	48
Anexo 3: Apéndice médico	57
Anexo 3.1 ¿Qué es una prótesis de cadera? ¿Por qué hay que poner una prótesis de cadera?	57
Anexo 3.2 Proceso hospitalario de intervención quirúrgica	58

Anexo 3.3 Definiciones y siglas médicas	60
Anexo 4: Servidor Web Django	61
Anexo 5: Autómatas finitos probabilistas (PFA)	63
Anexo 6: Modelado de procesos y Análisis de datos	65
Anexo 6.1 YAWL	65
Anexo 6.2 Business Process Modeling Notation	66
Anexo 6.3 Event-Driven Process Chains	67
Anexo 6.3 Estado del arte en la minería de datos	68
Anexo 7: Búsqueda del camino mínimo en grafo: Algoritmo de Dijkstra	73
Anexo 8: Gestión del proyecto	75

1. Introducción

En los consiguientes apartados de este capítulo se va a desarrollar una introducción explicando el dominio del problema y la casuística que este engloba. Una vez determinado el mismo se procederá a un análisis de la técnica utilizada para su resolución y el estado del arte de la misma.

1.1 Dominio del problema

La razón del origen de este proyecto y por consiguiente el *TFG* surge de la necesidad de optimizar los procesos médicos que siguen los pacientes desde su ingreso hasta su posterior alta hospitalaria. Se entiende como procesos médicos a todas las pruebas y tratamientos (llamadas también *actividades*) que se hacen a un paciente en un tiempo determinado ordenados cronológicamente. Este proceso médico viene determinado por lo que se conoce como una vía clínica. Desde un punto de vista médico, una vía clínica recoge en un documento estas pruebas y tratamientos que se deben de hacer ante una determinada patología [Ver Anexo 1] y son diseñadas por los profesionales del ámbito sanitario. Sin embargo, en el diseño de las vías clínicas se han tenido en cuenta hasta ahora solo criterios sanitarios. En un proyecto de investigación entre la Universidad de Zaragoza y el Hospital Clínico Universitario "Lozano Blesa" de Zaragoza (HCU) se ha propuesto añadir a estas vías clínicas la información del uso de los recursos (materiales, humanos, etc.) y también introducir en el modelo la comunicación que puede haber entre las vías clínicas.

De esta forma, si se modelan la mayoría de las vías clínicas utilizadas en el hospital, se podría utilizar los modelos matemáticos para estudiar propiedades del sistema de salud utilizando técnicas matemáticas. Se ha propuesto un modelo basado en un perfil *UML* que se llama *Healthcare System Specification (HSS)* y técnicas de transformación para obtener modelos de tipo redes de Petri. El problema de la aplicación de esta metodología en los hospitales es que hay muy pocas vías clínicas desarrolladas. Cabe mencionar aquí que las vías clínicas no se pueden importar de un hospital a otro ya que los recursos, el tipo de pacientes, etc. son distintos. Además, el proceso de diseño de nuevas vías clínicas es muy laborioso y puede tardar años.

El objetivo de este *TFG* es ayudar al personal médico en el diseño de nuevas vías clínicas y para eso se ha desarrollado un sistema de gestión para que sea utilizado en los hospitales. Los médicos están muy interesados en el proceso de diseño de nuevas vías clínicas, primero el jefe de servicio está interesado en una gestión mejor de su servicio, y segundo, desde un punto de vista estrictamente médico, se quiere obtener tratamientos iguales para todos los pacientes que sufren la misma enfermedad. El objetivo específico es desarrollar un sistema de información para que (a) los médicos puedan introducir datos de pacientes; (b) aplicar algoritmos para obtener modelos matemáticos en forma de autómatas finitos probabilísticos; (c) obtener algunos indicadores para ayudar a los médicos a refinar

estos modelos y (d) obtener las vías clínicas. Asimismo, se ha implementado una función para transformar los autómatas finitos probabilísticos a modelos de tipo redes de Petri para poder utilizar luego los métodos de análisis formal. Se ha colaborado con el Hospital Clínico Universitario “Lozano Blesa”, en particular con el servicio de *Cirugía Ortopédica y Traumatología*.

Se han utilizado datos reales, en particular registros de pacientes proporcionados por los profesionales sanitarios, para la generación de una vía clínica para determinadas patologías que precisan de *prótesis total de cadera* [Ver Anexo 4]. Todo esto, planteado desde la generación de un sistema que ayude a los médicos en esta toma de decisiones y en la optimización de recursos. En la figura 1, se plantea un esquema de lo que se abordará en este TFG.

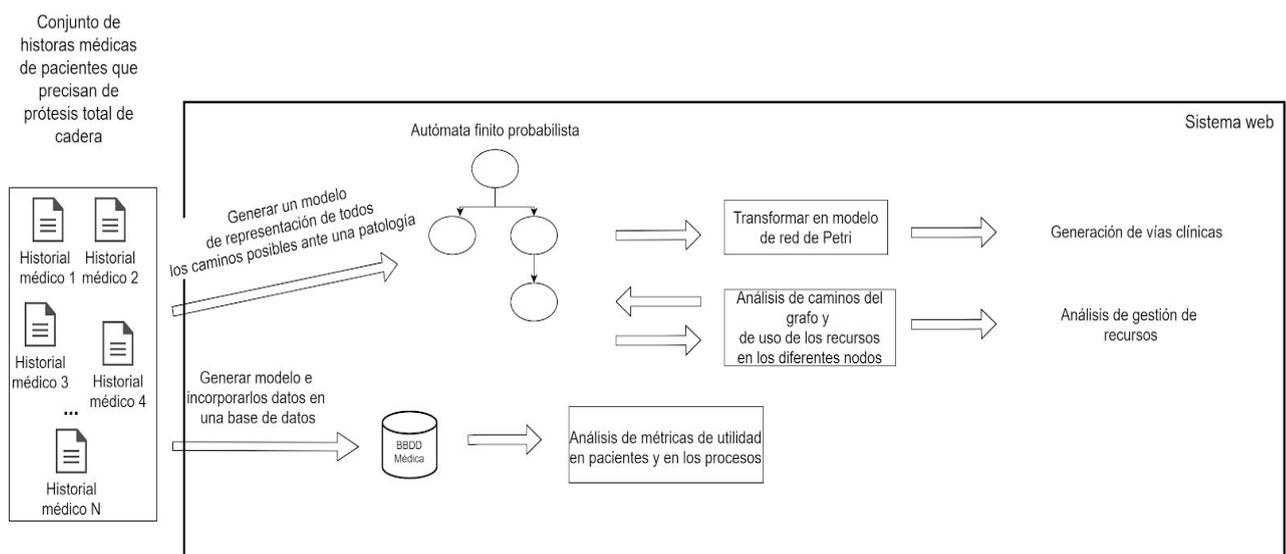


Figura 1: Caso de uso de minería de procesos

1.2 Objetivos del TFG

Los objetivos que se pretenden con el proyecto son los siguientes:

- **Procesar los datos médicos y ayudar al personal médico en el proceso de diseño de nuevas vías clínicas.** Como se ha dicho anteriormente, la generación de vías clínicas se realiza a través de varias reuniones de expertos y la redacción de un documento pertinente que se debe aprobar por la dirección médica del hospital. Con la aplicación desarrollada en este trabajo, se pretende ayudar al personal médico en el proceso de diseño de estos documentos procesando de forma automática los datos históricos de pacientes. Esta generación, se producirá a través del análisis de datos existentes en el servicio hospitalario.

- **Desarrollo de un modelo matemático para vías clínicas.** Otro aspecto importante, será la transformación de esos registros de procesos en un modelo matemático. Para este caso de uso, se utilizarán autómatas finitos probabilistas a los que se aplicarán una serie de algoritmos para la inferencia de conclusiones y ayudar al personal médico en el proceso de toma de decisiones en el proceso de diseño de vías clínicas.
- **Digitalización de los servicios de salud.** Todo este proceso, se realizará mediante un servidor web a través del cual los médicos interactúan tanto para introducir los datos como para visualizarlos y ver las conclusiones extraídas de los mismos.
- **Optimización de recursos.** Un aspecto que se estudiará en este trabajo será la optimización de los recursos hospitalarios. Se busca optimizar los procesos para así reducir los tiempos de estancia de los pacientes y todos los recursos que estos utilizan en su estancia.

1.3 Técnicas utilizada para su resolución

Para la resolución de este problema se ha utilizado un subconjunto de la minería de datos: la **minería de procesos**. En el presente trabajo únicamente se van a analizar registros de procesos que describen las trayectorias de diferentes pacientes a lo largo de su recuperación. Se describe a continuación cómo vamos a realizar este análisis:

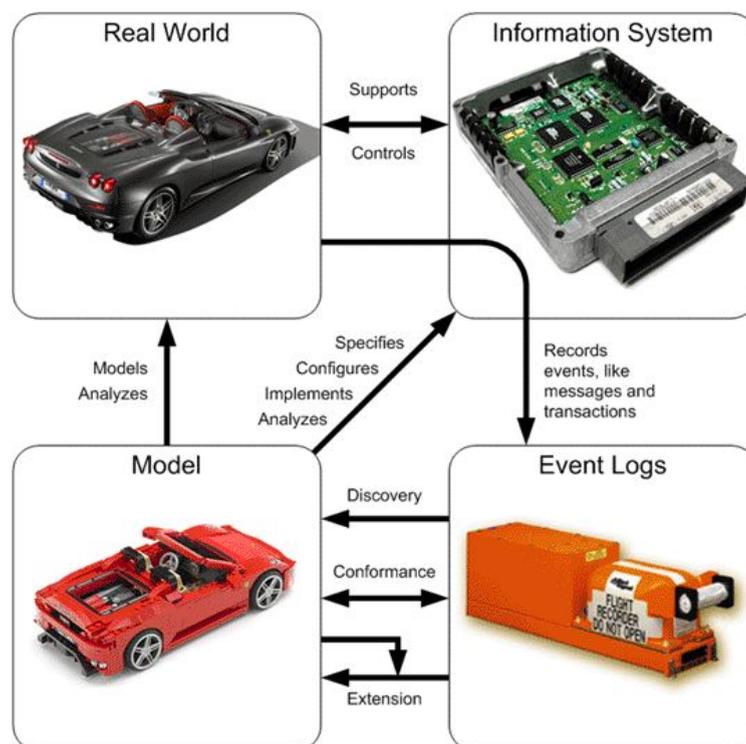


Figura 2: Qué es la minería de procesos (<http://www.processmining.org/research/start>)

Lo que se utiliza en este caso es una herramienta web que actúa como sistema de información en el cual a través de un servidor *Django* y una base de datos *SQLite* se registran los datos introducidos manualmente por los profesionales para un conjunto de pacientes determinado. Los algoritmos de análisis en este caso, no precisan exactamente de un análisis *Big Data* ya que se ha conseguido introducir solo la información de 30 registros (pacientes) Cabe destacar que en los hospitales la forma más habitual de trabajar es de utilizar registros en papel, por lo cual una parte muy importante de este trabajo ha sido el diseño e implementación del sistema de información e introducir datos buscando en los registros médicos. Sin embargo, dependiendo del algoritmo que se utilice se precisará de aumentar dichos registros.

Para poder extraer conclusiones y analizar estos datos se precisa de un modelo matemático que lo sostenga. En este caso, se ha considerado utilizar autómatas finitos probabilistas como una aproximación inicial al problema. Buscando en este autómata el camino de máxima/mínima probabilidad, ayudará a los médicos en refinar el modelo inicial para obtener al final la vía clínica. El autómata finito probabilístico final (correspondiente a la vía clínica) se transforma a un modelo formal de tipo *red de Petri* añadiendo la información de los recursos utilizados en cada actividad, con el objetivo de realizar posteriormente un análisis del sistema y la extracción de conclusiones.

1.4 Alcance

El alcance de esta aplicación se aplica tanto a nivel *micro* como *macro* en ámbitos de gestión. Se puede observar una aplicación directa de apoyo a los médicos ya que se pueden obtener diferentes métricas [Ver apartado 3.3]. Por otro lado, funciona como un sistema de ayuda a la toma de decisiones ya que se pueden generar diferentes grafos que muestren el recorrido de los tratamientos y las decisiones tomadas. Por otro lado, se puede utilizar a nivel de gerencia ya que a partir de la introducción de este sistema en el modelo de negocio repercutirá de forma positiva, ya que al optimizar la estancia de los pacientes se puede revertir en una nueva orientación de recursos por parte de dirigentes de las diferentes instituciones hospitalarias.

Otro aspecto a tener en cuenta sería una extensión del proyecto hacia una mayoría de enfermedades. Esto permitiría que con la gestión de recursos e información entre diferentes protocolos médicos se podría evaluar el modelo ante diferentes entradas y estudiar el soporte que este puede aportar.

1.6 ¿Qué es HSS?

El *Healthcare System Specification (HSS)* [2], definido como un perfil UML para que sea intuitivo para el personal médico se utiliza para el modelado de las vías clínicas. En particular HSS describe los procesos (secuencias de tratamientos, curas, análisis sanitaria, etc.) por los cuales pasan los pacientes, la utilización de recursos por parte del hospital y el

intercambio de información dentro de esta instalación debido a la interconexión entre departamentos y los procesos hospitalarios de un usuario.

Debido al uso en ingeniería del software que se le va a dar al sistema, *HSS* está definido bajo el estándar de *UML*. Este, es un lenguaje de modelado por excelencia en el ámbito de la ingeniería del software para sistemas del mundo real que además es entendible, siendo gráfico para los médicos ya que en su día a día estos realizan diagramas de decisión para llevar a cabo los diferentes tratamientos ante patologías. Este lenguaje además presenta la característica de que en algunos casos permiten la transformación directa a modelos formales del tipo *redes de Petri coloreadas* [Anexo 2], muy útiles para el análisis formal de los procesos.

Para representar el dominio hospitalario se ha utilizado un *diagrama de clases* basado en *UML*. En este diagrama se identifican las diferentes entidades que se utilizan en el hospital y se especifican las diferentes relaciones que existen entre ellas. En el caso de del Hospital Clínico Universitario “*Lozano Blesa*” se dedujo la infirió la siguiente implementación para el análisis de vías clínicas:

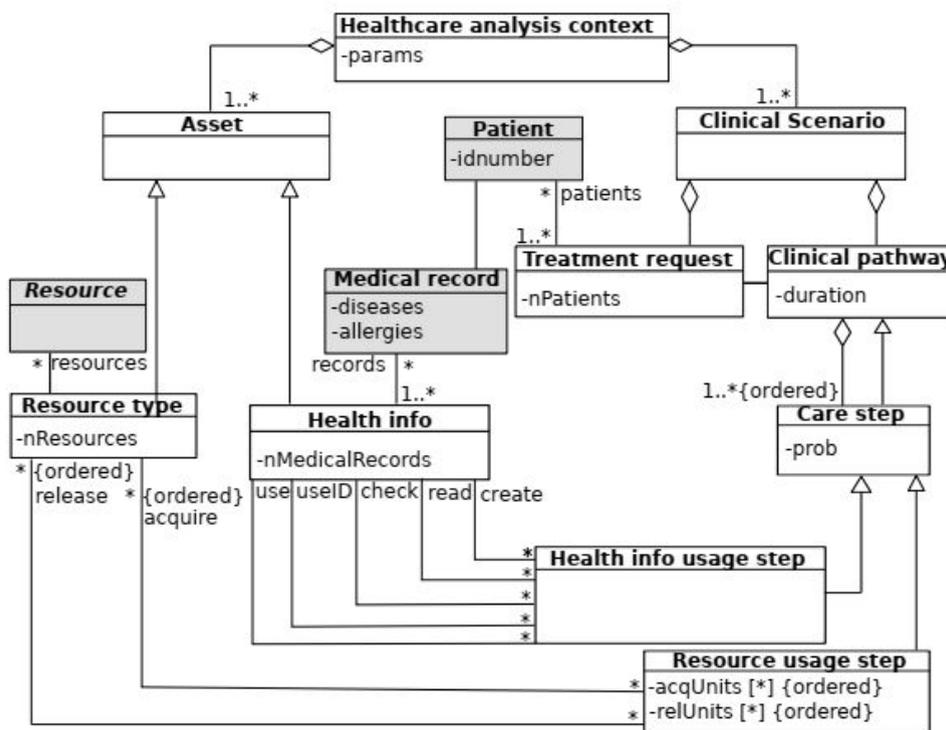


Figura 3: Diagrama de clases para el modelado y análisis de vías clínicas [2]

Aunque se exponen variedad de clases y relaciones entre ellas se ha decidido agruparlas en dos grandes grupos:

- **Vista estructural:** Se puede observar en el lado izquierdo del diagrama. En gris se especifica el recurso al que se está accediendo que puede ser *resource type* donde se englobaron camas, salas de operaciones, personal o paciente donde se recoge a

través de *Medical Report* donde se guarda el historial clínico (enfermedades padecidas, alergias...).

- **Vista de comportamiento:** A la derecha, se pueden observar los *Clinical Scenario* en estos están recogidas las vías clínicas y los pasos seguidos por un paciente ante una determinada patología. *Treatment request* recoge los pacientes que han seguido una determinada vía clínica para una patología. Los pasos de esta vía clínica se pueden observar en *care steps* donde se recoge además la fecha y la hora en la que se realizaron para un posterior análisis. Las dos últimas entidades recogen por un lado *Health info usage step* el tipo de operación que se realizará en un paso concreto de la vía clínica y por otro lado, *Resource usage step* el uso o liberalización de recursos del hospital.

Para un mayor detalle del diagrama se puede consultar la segunda referencia [2].

1.6.1 Análisis de de sistema hospitalario con HSS

En la figura de a continuación se puede observar la estructura que tiene el análisis de un flujo de trabajo en el hospital. Por un lado, el apartado del HSS se puede observar que el HSS engloba las dos partes, se tiene un análisis de trayecto que hace un paciente dentro del hospital y por otro lado los recursos que se usan asociados a este. Posteriormente, para ver los problemas que se tienen en estos flujos se procede a la formalización del modelo. Como UML es un lenguaje semiformal de modelado [3] es necesario aplicar un proceso de transformación que permita trabajar con un modelo formal de este. En esta línea de trabajo, en el hospital se ha estado trabajando con modelos de redes de Petri en trabajos de investigación de la Universidad de Zaragoza. Sin embargo, en este caso, como se verá en el apartado de implementación [Ver apartado 2] se ha decidido utilizar un modelo basado en autómatas finitos probabilistas. Finalmente, se ofrece un feedback a los profesionales en los que pueden observar los resultados en forma entendible a través de *UML-HSS*.

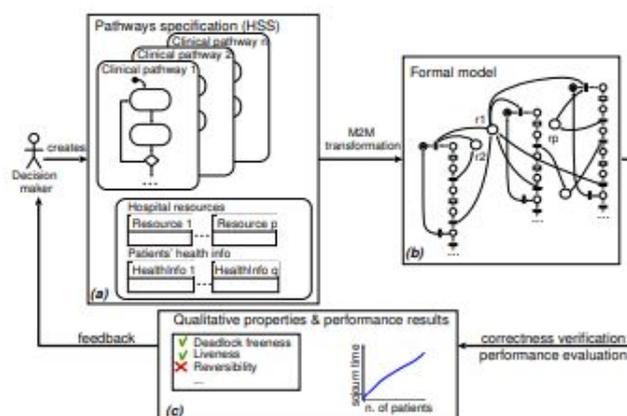


Figure 1: Modeling and analysis workflow.

Figura 4: Modelado de vías clínicas [2]

1.7 Estado del arte de la modelización de procesos

La primera parte con la que se trabajará será cómo se modelizan los diferentes tratamientos que recibe un paciente en una fecha y una hora determinadas durante la ejecución de la vía clínica. Además, se tiene que tener cuidado ya que en el dominio del problema se trabajará con procesos que tienen asociados recursos teniéndose en cuenta a la hora del modelado de estos.

En el campo de la modelación de procesos [1] se pueden encontrar diferentes formalismos, en las dos siguientes subsecciones se presentarán los dos tipos de modelos utilizados en este trabajo.

1.7.1 Autómata finito probabilista

Un autómata finito probabilista (FPA) [4] es un modelo que permite distribuciones probabilísticas a través de cadenas de eventos. Este modelo puede ser representado como un grafo etiquetado, las etiquetas siendo las probabilidades de salto de las transiciones, por lo que resulta muy útil y simple de entender. La definición de este formalismo se detalla en el Anexo 5. El objetivo dentro de este proyecto, es generar un modelo formal que nos permita registrar las diferentes posibilidades de tratamiento y utilizar estas probabilidades para refinar el modelo y obtener al final del proceso una vía clínica. Además, el uso de este modelo permite encontrar aquellos caminos de alta ante una determinada patología más o menos frecuentes ayudando de esta manera al equipo médico a realizar un análisis más exhaustivo de aquellos que interesen. En la Figura 5, se puede observar un ejemplo de un FPA. Este modelo se detallará más adelante [Ver Anexo 5 y apartado 3.1]

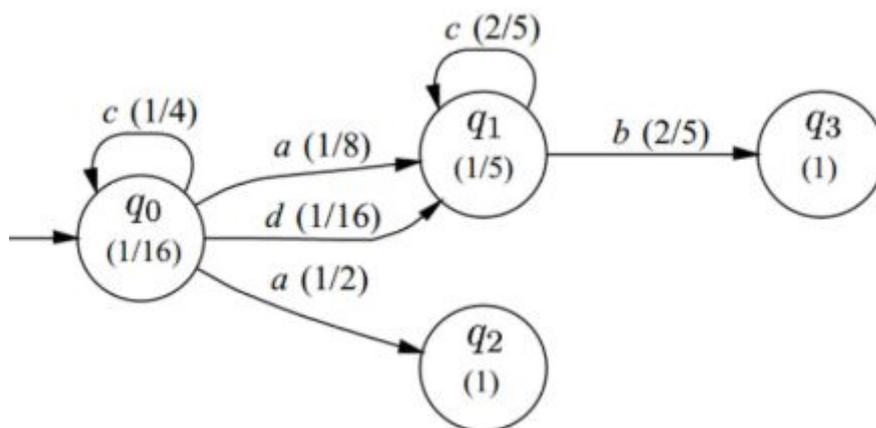


Figura 5: Ejemplo de FPA

2. Implementación

Este capítulo, describe el programa implementado que tiene como objetivo el muestreo de información de los pacientes y de un pequeño análisis de datos. Además, se van a debatir y decidir entre las diferentes tecnologías que se han barajado y finalmente su decisión e implementación. Se ha propuesto una herramienta que recoja y muestre los datos de los pacientes registrados en el hospital basándonos en tecnologías web para el desarrollo *frontend* y *backend*.

En los siguientes apartados, se podrán observar las necesidades de la aplicación (*Apartado 2.1*), las tecnologías que se han barajado para la implementación de la aplicación (*Apartado 2.2*), cómo se modelará la estructura de datos detrás de la aplicación (*Apartado 2.3*) y el desarrollo del apartado visual (*Apartado 2.4*).

2.1 Requisitos de la aplicación

A través de la colaboración con el Servicio de Cirugía Ortopédica y Traumatología del Hospital Clínico “Lozano Blesa” de Zaragoza en representación del *Dr. Albareda* a través de reuniones presenciales se determinaron una serie de requisitos funcionales con los que contará la aplicación que se presentan a continuación:

Requisitos funcionales	
RF - 1	El sistema contará con un sistema de gestión de usuarios.
RF - 2	El sistema contará con un sistema de administración de usuarios y entidades en un panel de administración.
RF - 3	El sistema contará con un sistema de recuperación de contraseñas de usuario.
RF - 4	El sistema mostrará una serie de métricas útiles para el personal sanitario.
RF - 5	El sistema mostrará la lista de pacientes asociados a un profesional.
RF - 6	El sistema mostrará las actividades que se tienen dentro del sistema hospitalario.
RF - 7	El sistema mostrará las actividades que se realizaron en una fecha a un paciente determinado con una duración en concreto.
RF - 8	El sistema mostrará la lista de recursos que se tienen dentro del sistema

	hospitalario.
RF - 9	El sistema mostrará un grafo con los diferentes caminos que se han escogido en las vías clínicas.

Requisitos no funcionales	
RNF - 1	La aplicación no tardará más de 10 segundos en mostrar las imágenes de grafos Correspondientes a los autómatas finitos probabilísticos (modelo formal).
RNF - 2	El sistema no tardará más de 10 segundos en iniciar sesión.

2.2 Decisiones de implementación

El proyecto, gira principalmente entre dos pilares. Por un lado, el desarrollo de un servidor web de muestreo de información y por el otro, la gestión y el trabajo con grafos y *redes de Petri*.

Se barajó la utilización de *Java* o *Javascript* para el *backend*. *Java* presenta una experiencia muy sólida, tiene diversas formas de implementación a través de servidores web *Tomcat* o *Spring*. Por otro lado, el uso de *Javascript* está siendo muy extendido en estos días ya que es muy utilizada en tecnologías web *full stack* a través de *NodeJs* (*backend*) y *React* (*frontend*) pudiéndose combinar también con tecnologías como *Angular*. Este tipo de tecnologías basadas en *Javascript* tienen un fácil acceso a paquetes y *plugins* hechos por la comunidad para su adición a los proyectos.

Sin embargo, el uso de *Java* puede ser tedioso en comparación a otros lenguajes más funcionales como *Python*. Por ejemplo, a la hora de trabajar con estructuras de datos necesarias para utilizar modelos de tipo *redes de Petri* o *autómatas finitos probabilistas* resulta más útil *Python* ya que permite instalar paquetes fácilmente con *pip* o desarrollar un módulo en muy poco tiempo. Asimismo, el uso de paquetes en *Java* no es tan fácil en comparación a las tecnologías basadas en *Javascript*. Por otro lado, *Javascript* a través de sus tecnologías resulta muy superior al resto en el desarrollo de aplicaciones web y con la gestión de paquetes, pero no existen muchas librerías para las estructuras de datos con que se pretende trabajar en este proyecto.

Por ello, se decidió utilizar tecnologías *Python* que aportan un enfoque sólido para el desarrollo de aplicaciones web y cuenta con una fácil instalación de paquetes a través de *pip*. Además, a la hora de tratar con grafos y recorridos se encuentran con facilidad librerías

y trabajos académicos que utilizan con solvencia este tipo de modelos. Se utilizó el *framework web Django* [Ver Anexo 4] que funciona como un servidor web basado en *Python* que implementa el patrón arquitectura *modelo vista controlador* frente a las tecnologías basadas en *servlets* de *Apache*. Este modelo aporta una mayor solvencia a la hora de atender peticiones web, además de otorgar un diseño basado en tres capas que permite separar los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

En este caso específico para la gestión de los datos *Django* integra una base de datos *SQLite* con la que trabajar, con lo que para casos de análisis como este con pocos registros no precisa de la instalación de otra base de datos. Además, *Django* cuenta con su propia *API* de alto nivel para la realización de consultas y gestión de datos sobre la base de datos. Se decide el uso de esta base de datos debido a que se encuentra ya integrada y no se va a trabajar con un alto nivel de datos ni un modelo muy complicado por lo que no se precisará de características propias de un gestor de bases de datos concreto.

Para poder controlar el versionado del proyecto se ha utilizado el software de gestión de proyectos y versionado *GitHub* con la creación de un repositorio público con la siguiente dirección:

https://github.com/daubide/SI_HLozanoBlesa

Todos los archivos reseñables que no sean propios del servidor *Django* de este proyecto se pueden encontrar en la carpeta *extra*. Los ficheros del proyecto *Django* se podrán encontrar en */processmining*.

2.3 Modelo de la base de datos

A su vez, acorde a las necesidades explicadas anteriormente y bajo la revisión del doctor Albareda se definió el siguiente esquema relacional para modelar el sistema hospitalario y de una determinada patología, la **artroplastia de cadera** en base a los estándares *HSS*.

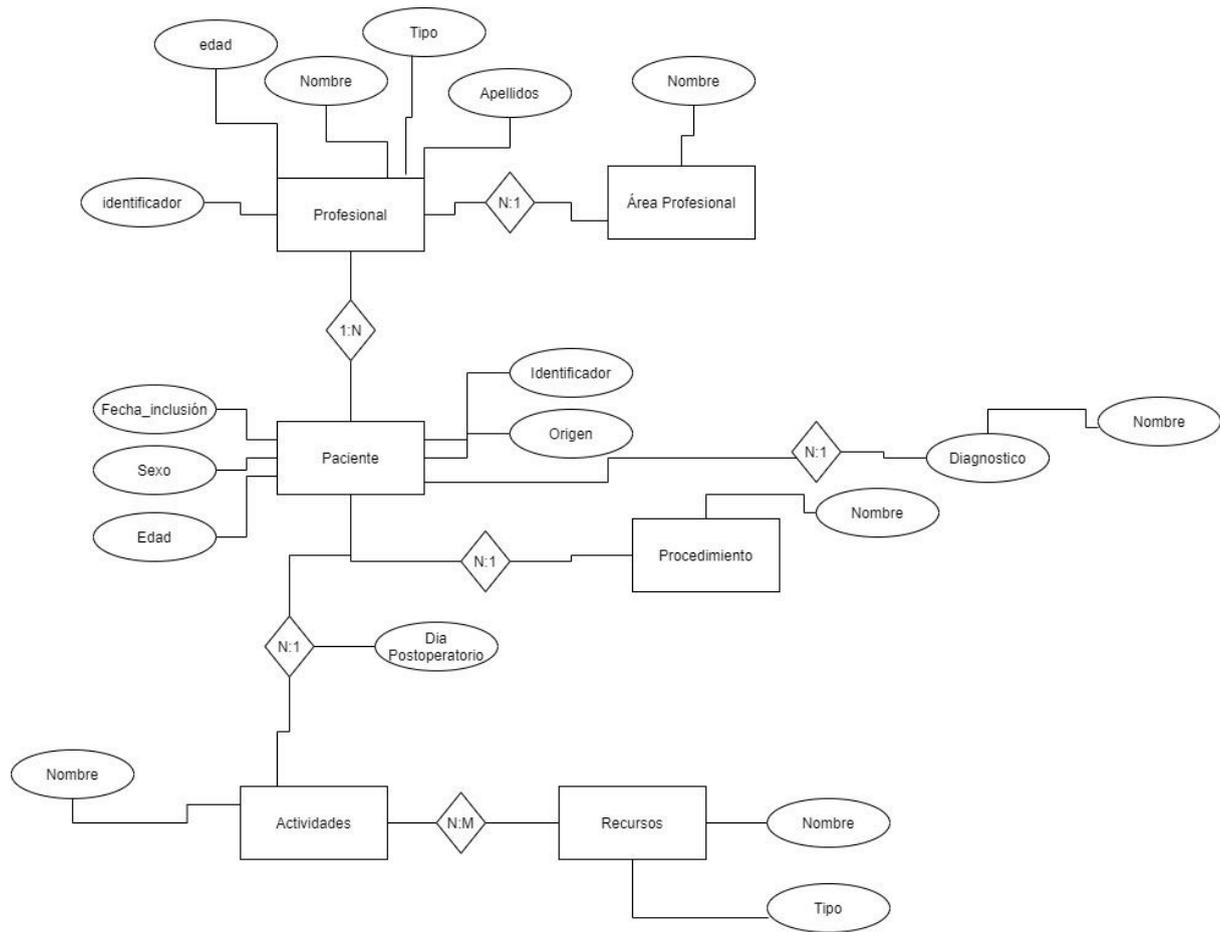


Figura 8: Esquema Entidad Relación de la base de datos

Cabe destacar los siguientes aspectos en las entidades:

- **Tipo:** En la entidad Profesional indica el tipo de profesional de que se trata. Es decir, si es un médico residente (*MIR*) o un médico adjunto al servicio. En el caso de recurso refleja el tipo de recurso del que se trata si es un recurso, humano o material.
- **Origen:** Refleja el origen de un paciente. En este sistema, sólo se ha tenido en cuenta si procede de área rural o de Zaragoza capital.
- **Diagnóstico y procedimientos:** Aunque únicamente se tiene un procedimiento y unos pocos diagnósticos se ha mantenido el diseño lógico separándolos como entidades.

A continuación se adjunta un esquema que refleja las tablas implementadas y sus tipos:

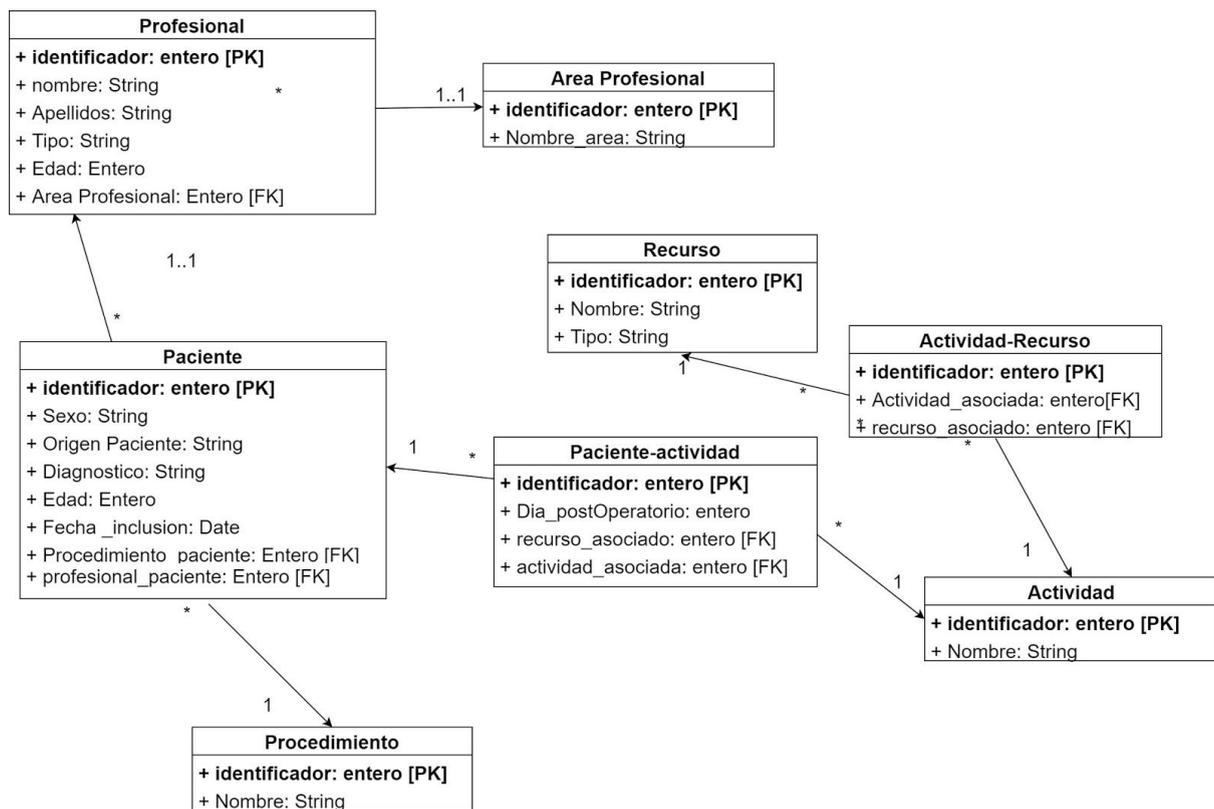


Figura 9: Esquema Entidad Relación de la base de datos

Se destacan los siguientes aspectos:

- Se ha utilizado el símbolo * para indicar una cardinalidad de N elementos y flechas para la direccionalidad de las relaciones.
- Para modelar la relación actividad-recurso con cardinalidad $M:N$ se ha creado una tabla que relacione los identificadores de ambos elementos. La misma solución se ha optado para relacionar pacientes y actividades e incluir el atributo de *dia_postOperatorio*.

Aunque se trata de un modelo sencillo, es el que mejor se amolda a los problemas descritos por los médicos y al *HSS*. Por ello, se descartaron modelos que incluyen diferentes especializaciones de tipos de personas (profesional, adjuntos, ...). Se había planteado un modelo más complicado que se descartó en la primera reunión con el equipo médico.

2.3.1 Modelado y extracción de datos

Este modelo, se trasladó directamente al servidor *Django* de la aplicación a través de la creación de tablas para la base de datos *SQLite*. Estas sentencias, se introdujeron dentro del archivo *Models.py* fijándose en el esquema anteriormente expuesto. A continuación, se expone un ejemplo de sentencia:

```

class PROCEDIMIENTOS(models.Model):
    nombre_procedimiento = models.CharField(
        null=False,
        max_length=100
    )

```

(<https://docs.djangoproject.com/en/3.1/ref/models/instances/>).

Además de eso, para la introducción de datos se tuvo una reunión con el equipo médico y se concretó que la manera más efectiva de introducción de datos sería la preparación de los registros en hojas *Excel*. Esto se debe, a que por un lado requería de la explicación de la introducción de datos en el sistema y ofrecía una alta resistencia al cambio.

Para la obtención de los datos se tuvo que hacer una traslación a los registros *Excel* con el equipo a través de registros como el que se observa a continuación.

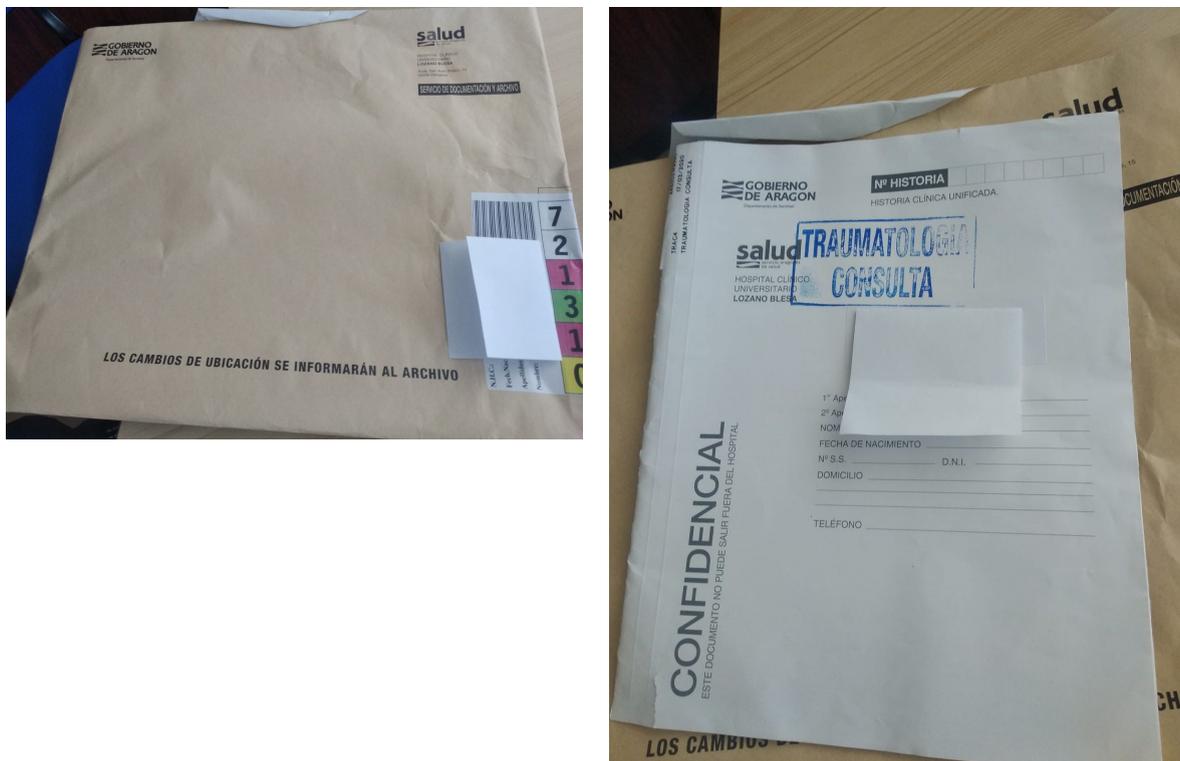


Figura 10: Ejemplo de historias clínicas

Este trabajo, requirió de la coordinación con el personal médico para su digitalización y análisis teniendo en cuenta los siguientes factores:

- Por un lado, la ausencia de sistemas informáticos sanitarios: las historias clínicas hasta antes del Covid-19 se recogían manualmente en este departamento hospitalario por lo cual no se precisa de toda la información digitalizada de los usuarios pero sí de un acceso especial a los datos. A raíz de la llegada del Covid-19

a España, se comienza a registrar de forma informática los episodios por los que ingresan los pacientes, con el objetivo de disminuir el riesgo de contagio entre profesionales sanitarios. A partir de este momento, la mayoría de los datos de los pacientes se registran en la llamada “*historia clínica electrónica*” . Por ello, al coger casos entre 2019 y 2020 se pueden encontrar historias registradas manualmente y otras a través de la historia clínica electrónica.

- De los datos recogidos de historias clínicas no informatizados, no todos se encuentran reflejados con exactitud en las mismas. Por ejemplo, al realizar una artroplastia total de cadera, la primera dosis de heparina de bajo peso molecular se debe iniciar a las 12 horas tras la intervención y aunque la administración de la misma sea en dicho momento, el registro será distinto, dado que manualmente no se escribe la hora exacta. Lo mismo ocurre con las dosis post-intervención de antibiótico que se administran a las 8 y 16 horas desde la intervención.
- Debido a la organización del propio hospital, tras operar a un paciente, se realiza una radiografía de control al día siguiente. En algunos casos, esta radiografía, no se realiza al día siguiente, por ejemplo, un paciente intervenido el viernes por la tarde, puede ocurrir que la radiografía de control no se haga el sábado dado que es festivo y se cuenta con menos personal que entre semana, teniendo que posponer dicha radiografía hasta el lunes. Esto puede implicar un alargamiento de la estancia hospitalaria y un mayor incremento de recursos sanitarios, dado que ningún paciente es dado de alta hospitalaria sin haber realizado la radiografía de control.
- Otro problema, es el desorden que hay dentro de la propia historia clínica. Los papeles se encuentran desordenados en algunas ocasiones, sin grapar y en diferentes carpetas. Esto provoca que en algunas ocasiones haya pérdida o trasapelamiento de información. Esto se soluciona con una revisión exhaustiva de toda la historia y con el uso de la historia clínica electrónica.
- Por otro lado, además de ser información a papel sin estructurar y sobredimensionada (historias clínicas de más de 20 páginas), se encuentran acrónimos y abreviaturas que un técnico no puede descifrar.

2.3.2 Transformación y carga de datos

Las hojas excel escritas por el equipo médico fueron 6 que se dividieron de la siguiente manera:

- **Datos analíticos:** Contienen la información que se utilizará posteriormente para la elaboración del panel de mandos en Power BI. Por un lado, contienen información general de paciente: Números de historial clínico, edad, sexo, procedencia, lado que se le opera, motivo y duración de la intervención, día de ingreso, día de operación, día de alta. Por el otro lado, información médica [Ver Anexo 3 apartado 3.3] donde se encuentra: el riesgo anestésico del paciente (ASA) y el tipo de anestesia aplicada, si tomaba previamente anticoagulantes (ACO), la prótesis que se le aplicó, en que

momento se le aplicó el antibiótico, si precisó de transfusión, indicadores de hemoglobina y hematocrito previo y posterior a la intervención y si el paciente tuvo patologías previas o complicaciones postoperatorias. Por temas de confidencialidad y anonimato del paciente no se permite trabajar con su nombre y apellidos.

- **Profesionales:** Se recogen datos asociados al personal que interviene: un identificador aleatorio, nombres y apellidos, si se trata de personal adjunto o médico residente, la especialidad en la que trabajan y el área de población que atienden.
- **Actividades:** Se diferencian las diferentes actividades a las que pueden estar sometidos los pacientes en su proceso hospitalario y los recursos necesarios para su ejecución.
- **Pacientes:** Contiene información propia de los pacientes que está expuesta en datos clínicos. Se ha realizado esta tabla para su posterior procesamiento en el sistema.
- **Datos de los procesos del paciente:** Se expone el historial clínico del paciente, el día postoperatorio de la actividad aplicada y el propio nombre de la actividad.
- **Recursos:** Contiene los diferentes recursos que se tienen en el sistema y el tipo que son (humano o material).

Se presentan parte de datos recogidos en la hoja Excel²:

	C	D	E	F	G	H	I	J
1	NHC	Edad	Sexo	Lado	ACO(Antcoagulantes)	ASA(Riesgo anestésico paciente)	Motivo Intervencin	Día de Ingreso
2	82886	54	M	Izquierdo	NO	II	Coxartrosis	09/02/2020
3	578407	49	M	Izquierdo	NO	II	Coxartrosis	23/06/2020
4	110711	61	M	Izquierdo	NO	II	Coxartrosis	10/02/2020
5	831852	60	M	Izquierdo	NO	II	Coxartrosis	28/06/2020
6	604154	56	F	Derecha	NO	II	Necrosis	31/05/2020
7	825224	53	M	Derecha	NO	II	Coxartrosis	20/01/2019
8	911965	53	M	Derecha	SI	III	Coxartrosis	17/01/2019
9	74676	65	M	Izquierda	NO	II	Coxartrosis	04/02/2019
10	757488	58	M	Izquierda	NO	III	Coxartrosis	06/02/2019
11	198464	40	F	Derecha	NO	III	Coxartrosis	14/02/2019
12	571898	28	M	Derecha	NO	I	Necrosis (perthes)	24/02/2019
13	24579	52	M	Derecha	NO	II	Coxartrosis	11/03/2019
14	91503	46	F	Izquierda	NO	I	Coxartrosis	21/03/2019
15	108266	59	F	Derecha	NO	I	Necrosis	21/03/2019
16	20860	83	M	Derecha	NO	II	Coxartrosis	26/03/2019
17	753531	46	M	Izquierda	NO	I	Coxartrosis	15/04/2019
18	155215	55	M	Derecha	NO	I	Coxartrosis	05/05/2019
19	89896	46	M	Derecha	NO	I	Necrosis	19/05/2019
20	247015	48	M	Izquierda	NO	II	Necrosis	29/05/2019
21	108486	50	F	Izquierda	NO	I	Necrosis	06/06/2019
22	28206	74	M	Derecha	NO	II	Coxartrosis	18/06/2019
23	881072	63	M	Derecha	NO	I	Coxartrosis	25/06/2019
24	570442	56	M	Derecha	NO	II	Coxartrosis	07/10/2019
25	82447	63	F	Derecha	NO	III	Coxartrosis	07/10/2019

² Consultable en el código de Github [/extra/pacientes_def.xlsx](#)

Día de Ingreso	Día de la semana iqx	Tipo de protesis	Cirujano Principal	Día Intervencion	Día Semana alta	Día de alta
09/02/2020	Miercoles	Evolution Ecofit	Torres	10/02/2020	Martes	16/02/2020
23/06/2020	Miercoles	Evolution Ecofit	Górriz	24/06/2020	Domingo	28/06/2020
10/02/2020	Miercoles	Evolution Ecofit	Albareda	11/02/2020	Domingo	15/02/2020
28/06/2020	Lunes	Evolution Ecofit	Albareda	29/06/2020	Jueves	02/07/2020
31/05/2020	Lunes	Evolution Ecofit	Lasierra	01/06/2020	Lunes	08/06/2020
20/01/2019	Lunes	Evolution Ecofit	Gomez	21/01/2019	Viernes	25/01/2019
17/01/2019	Viernes	Evolution Ecofit	Gomez	18/01/2020	Martes	22/01/2020
04/02/2019	Martes	Evolution Ecofit	Lasierra	05/02/2019	Sabado	09/02/2019
06/02/2019	Jueves	Evolution Ecofit	Torres	07/02/2019	Lunes	11/02/2019
14/02/2019	Viernes	Evolution Ecofit	Aznar	15/02/2019	Miercoles	20/02/2019
24/02/2019	Lunes	Evolution Ecofit	Gomez	25/02/2019	Viernes	01/03/2019
11/03/2019	Martes	Evolution Ecofit	Lasierra	12/03/2019	Viernes	15/03/2019
21/03/2019	Viernes	Evolution Ecofit	Torres	22/03/2019	Martes	26/03/2019
21/03/2019	Viernes	Evolution Ecofit	Gomez	22/03/2019	Martes	26/03/2019
26/03/2019	Miercoles	Evolution Ecofit	Gomez	27/03/2019	Lunes	01/04/2019
15/04/2019	Martes	Evolution Ecofit	Gomez	16/04/2019	Lunes	22/04/2019
05/05/2019	Lunes	Evolution Ecofit	Torres	06/05/2019	Viernes	10/05/2019
19/05/2019	Lunes	Evolution Ecofit	Lasierra	20/05/2019	Jueves	23/05/2019
29/05/2019	Jueves	Evolution Ecofit	Nicolau	30/05/2019	Domingo	02/06/2019
06/06/2019	Viernes	Evolution Ecofit	Lasierra	07/06/2019	Lunes	10/06/2019
18/06/2019	Miercoles	Evolution Ecofit	Albareda	19/06/2019	Domingo	23/06/2019
25/06/2019	Miercoles	Evolution Ecofit	Lasierra	26/06/2019	Sabado	29/06/2019
07/10/2019	Martes	Evolution Ecofit	Gomez	08/10/2019	Sabado	12/10/2019
07/10/2019	Martes	Evolution Ecofit	Gomez	08/10/2019	Domingo	12/10/2019

P	Q	R	S	T	U	V	W
Día de alta	Hora de entrada	Tiempo TOTAL QX	Tipo de anestesia	ATB(Antibiotico)	Momento	HB-PRE	HTO-PRE
16/02/2020	8:45	105	Intradural	Cefazolina 2g	Inducción Anestésica	16,6	48,8
28/06/2020	12:45	105	Intradural	Cefazolina 2g	Inducción Anestésica	14,4	40,8
15/02/2020	17:45	105	Intradural	Cefazolina 2g	Inducción Anestésica	15,4	45,3
02/07/2020	17:53	82	Intradural	Clindamicina 600 mg	Inducción Anestésica	14,7	42,7
08/06/2020	17:00	95	Intradural	Cefazolina 2g	Inducción Anestésica	14,4	43,9
25/01/2019	8:30	110	Intradural	Cefazolina 2g	Inducción Anestésica	15,3	44,6
22/01/2020	15:30	120	Intradural	Cefazolina 2g	Inducción Anestésica	15,9	47,9
09/02/2019	11:30	100	Intradural	Cefazolina 2g	Inducción Anestésica	16,6	49,5
11/02/2019	8:35	120	Intradural	Cefazolina 2g	Inducción Anestésica	15,6	44,5
20/02/2019	9:15	130	Intradural	Cefazolina 2g	Inducción Anestésica	14,2	43,1
01/03/2019	8:50	180	General	Cefazolina 2g	Inducción Anestésica	15,6	46,3
15/03/2019	17:25	95	Intradural	Cefazolina 2g	Inducción Anestésica	16,5	49,4
26/03/2019	9:05	105	General	Cefazolina 2g	Inducción Anestésica	14,7	43,3
26/03/2019	17:20	120	Intradural	Cefazolina 2g	Inducción Anestésica	13,3	39,2
01/04/2019	8:35	105	Intradural	Cefazolina 2g	Inducción Anestésica	15,4	45,4
22/04/2019	9:35	125	General	Cefazolina 2g	Inducción Anestésica	15,6	46,1
10/05/2019	8:45	130	Intradural	Cefazolina 2g	Inducción Anestésica	16,2	46,6
23/05/2019	15:35	100	General	Cefazolina 2g	Inducción Anestésica	15,5	45,4
02/06/2019	15:20	90	Intradural	Cefazolina 2g	Inducción Anestésica	15,8	46,8
10/06/2019	15:30	110	General	Cefazolina 2g	Inducción Anestésica	13,3	39,4
23/06/2019	17:30	110	Intradural	Cefazolina 2g	Inducción Anestésica	14,6	44
29/06/2019	15:15	100	Intradural	Cefazolina 2g	Inducción Anestésica	17,1	50,7
12/10/2019	8:30	120	General	Cefazolina 2 gr	Inducción Anestésica	16,6	49,5
12/10/2019	11:20	115	Intradural	Cefazolina 2gr	Inducción Anestésica	14,2	43,0

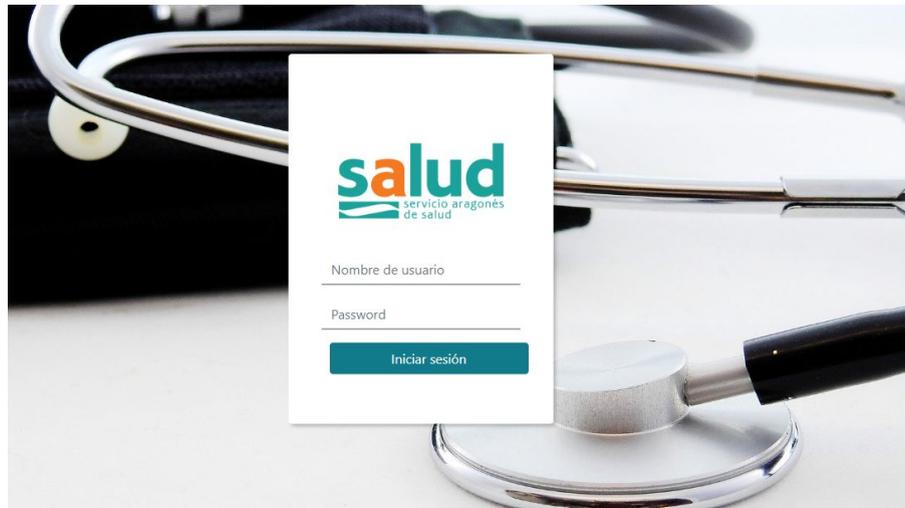
Figura 11: Ejemplo de recogida de datos en hoja Excel

Tras ello, se preparó un *script* que introdujera los datos dentro de la base de datos *SQLite* del sistema. Este *script* se puede encontrar dentro del código en *poblatedb.py*. Utilizando la librería *xlrd* (<https://xlrd.readthedocs.io/en/latest/>) de *Python* se trabajó hoja a hoja extrayendo las tablas. La inserción se hizo a través de la *API* que presenta *Django* para crear y actualizar objetos. Se encontró un problema con las fechas, ya que excel las guarda como enteros debido a que se cuentan los días desde el uno de enero de 1990. Sin embargo, la estructura para guardar fechas, *Datetime*, contiene una función para transformar estos enteros directamente.

2.4 Desarrollo del aspecto visual

Para otorgar un aspecto agradable a la vista se ha decidido desarrollar las diferentes vistas del sistema utilizando la herramienta *Bootstrap Studio* a través de una licencia de estudiante que permite trabajar con los componentes de *Bootstrap* y desarrollar una vista de forma estática en *HTML*. Más adelante, para añadir dinamismo a la web se ha utilizado el lenguaje de plantillas de *Django*. Para definir los diferentes menús de la aplicación web se ha reunido con el equipo médico para que resulte un aspecto bonito y funcional. A continuación, se puede observar algunas capturas de la aplicación:

Vista de Login



Vista de pacientes

Una captura de pantalla de la interfaz de usuario para la lista de pacientes. En la parte superior, hay un encabezado con el texto 'Bienvenido, Jorge' y un botón de 'Desconexión?'. A la derecha, hay un menú de navegación con los ítems: 'Procedimientos', 'Actividades', 'Recursos', 'Pacientes', 'Logs de pacientes' y 'Minería de datos'. El título principal de la sección es 'Lista de pacientes'. Debajo, hay una tabla con los siguientes datos:

NHC	Sexo	Origen del paciente	Edad	Fecha inclusión	Diagnostico del paciente	Procedimiento aplicado	Facultativo asociado
28206	M	Zaragoza	74	18 de Junio de 2019	Coxartrosis	Artroplastia cadera	Jorge
110711	M	Zaragoza	61	10 de Febrero de 2020	Coxartrosis	Artroplastia cadera	Jorge
831852	M	Zaragoza	60	28 de Junio de 2020	Coxartrosis	Artroplastia cadera	Jorge

En la parte inferior, hay un pie de página con el logo 'Hospital Clínico Universitario Lozano Blesa' y el logo 'salud servicio aragonés'. También se menciona 'Trabajo de Final de Grado de Ingeniería informática de David Urbide en la especialidad'.

Vista de procedimientos

Lista de procedimientos

ID	Nombre
1	Artroplastia cadera



Lista de actividades

Vista de actividades

Nombre actividad	Recurso 1	Recurso 2
Analtica	Enfermera	Recurso Farmaceutico
RX	Celador	Maquina RX
Antibiotico	Enfermera	Recurso Farmaceutico
Heparina	Enfermera	Recurso Farmaceutico
Sedestacion	Celador	
Momento caminar muletas	Celador	

Lista de recursos

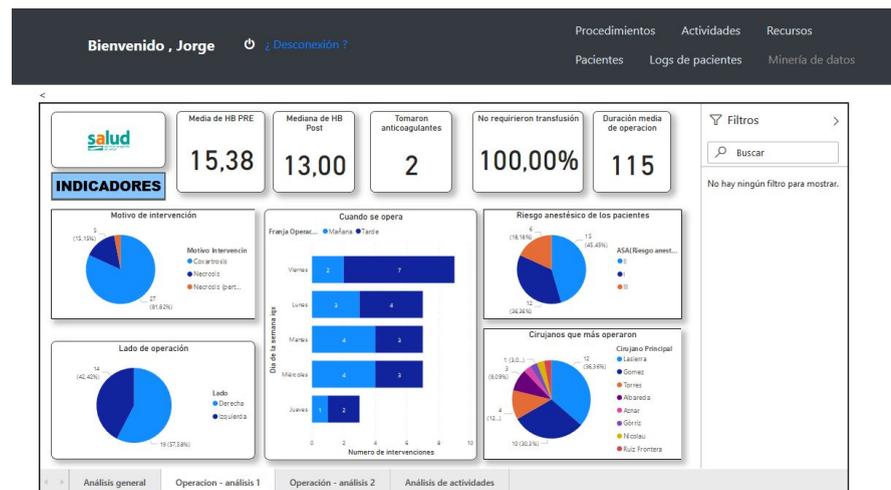
Vista de recursos

ID	Actividad	Recursos
1	Medico	Humano
2	Enfermera	Humano
3	Recurso Farmaceutico	Material
4	Celador	Humano
5	Cama	Material
6	Maquina RX	Material

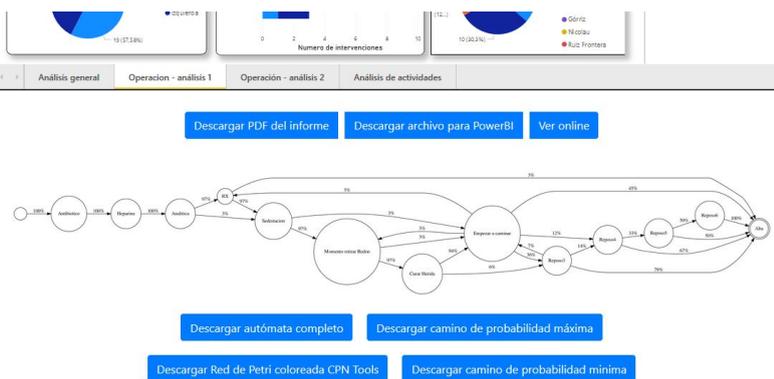
Vista de registro de actividades

Día postoperatorio	Nombre actividad	NHC paciente
1	Antibiotico	28206
1	Heparina	28206
1	Analitica	28206
1	RX	28206
1	Sedestacion	28206
1	Momento retirar Redon	28206
1	Curar Herida	28206
2	Empezar a caminar	28206
3	Alta	28206

Vista de minería de datos



Vista de minería de datos (continuación)



Administración de Django

Sitio administrativo

Panel de administración de la base de datos

AUTENTICACIÓN Y AUTORIZACIÓN	
Grupos	+ Añadir ✎ Modificar
Usuarios	+ Añadir ✎ Modificar
PROCESMINING	
Actividad_pacientes	+ Añadir ✎ Modificar
Actividades	+ Añadir ✎ Modificar
Areaprofesionales	+ Añadir ✎ Modificar
Pacientes	+ Añadir ✎ Modificar
Procedimientos	+ Añadir ✎ Modificar
Profesionales	+ Añadir ✎ Modificar
Recursos	+ Añadir ✎ Modificar

Figura 12: Vistas de la aplicación

La aplicación se ha dividido en 6 menús principales:

- **Lista de procedimientos:** Se podrán observar la lista de procedimientos con los que se tratarán a los pacientes. En nuestro caso únicamente se aplicarán un procedimiento para la sustitución total de cadera: *artroplastia de cadera*.
- **Lista de actividades:** Se registran las diferentes actividades que se pueden realizar a los pacientes y los recursos necesarios para las mismas.
- **Lista de pacientes:** Se mostrarán los pacientes con sus datos (sexo, origen del paciente, edad, fecha de ingreso, diagnóstico, procedimiento aplicado y el profesional a cargo) siendo estos identificados por el número de historia clínica. En esta pantalla, únicamente se verán los pacientes asociados al profesional que ha iniciado sesión en el sistema.
- **Visualización de logs:** Se visualizarán los diferentes registros de las actividades que se les ha hecho a los pacientes con la referencia del día postoperatorio asociado al mismo. Al igual que en pacientes, únicamente se visualizarán los registros de los pacientes cuyo profesional sea el que haya iniciado sesión en el sistema.

- **Minería de datos:** Se mostrarán los resultados del análisis de datos:
 - **Panel de mando:** Se mostrará un panel de mando con los indicadores de interés del personal médico pudiendo ver online, descargar en PDF o descargar el archivo *Power BI*.
 - **Vías clínicas:** Se mostrarán los resultados extraídos a través del autómata finito probabilista. Se mostrará el grafo global de probabilidades y los caminos de mínima y máxima probabilidad para la extracción de conclusiones.
- **Panel de administración:** En este apartado se puede gestionar la base de datos y los usuarios registrados del sistema. Se pueden modificar y añadir nuevos registros en el sistema.

El objetivo de esta aplicación es facilitar a los médicos una interfaz amigable para la visualización de resultados y para trabajar con ella adecuadamente.

3. Análisis de datos

A continuación, se va a exponer las diferentes técnicas que se han aplicado para el análisis de estos datos médicos

3.1 Autómatas finitos probabilísticos

En este caso se ha utilizado este modelo matemático para modelar las vías clínicas. Como se ha descrito anteriormente en la introducción, se modelaron los diferentes pasos por los que pasa un paciente a través de los FPA [Ver Anexo 5]. De tal forma, que se modelan como conjunto de estados todas los procesos clínicos a los que se somete un paciente.

Para esta modelización, se partió de los datos almacenados en la base de datos y se generó un fichero de salida. Este fichero, contiene para cada actividad (en este caso, proceso que pasa un paciente desde que se le opera hasta que le den el alta) asociadas una serie de acciones separadas por un carácter ‘;’ como se puede ver a continuación en estas líneas de ejemplo:

```
Antibiotico;Heparina;Analitica;RX;Sedestacion;Momento retirar Redon;Curar Herida;Empezar a caminar;Reposo4;Alta
Antibiotico;Heparina;Analitica;RX;Sedestacion;Momento retirar Redon;Curar Herida;Empezar a caminar;Alta
Antibiotico;Heparina;Analitica;RX;Sedestacion;Momento retirar Redon;Curar Herida;Empezar a caminar;Alta
Antibiotico;Heparina;Analitica;RX;Sedestacion;Momento retirar Redon;Curar Herida;Empezar a caminar;Reposo3;Alta
Antibiotico;Heparina;Analitica;RX;Sedestacion;Momento retirar Redon;Empezar a caminar;Reposo3;Reposo4;Alta
```

Figura 13: Salida fichero actividades_def.txt

Se ha asociado a cada un estado único dentro de nuestro autómata. Esto, nos permite generar el espacio de estados QA_k para cada vía clínica. Además, se creará un estado inicial q_0 , que no esté asociado a ninguna actividad que será quien dé pie a las diferentes posibilidades de inicio de actividad. A continuación, se expone una figura explicativa

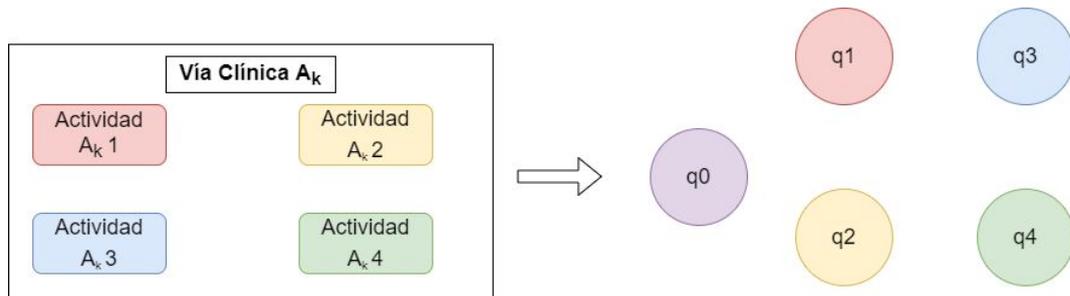


Figura 14: Ejemplo de creación de espacio de estados para una vía clínica A_k

→ En este caso, para crear las transiciones se debe de observar el estado previo posterior y el estado sucesor creando dos transiciones de tal manera que: si se tiene la secuencia de actividades k_{n-1}, k_n, k_{n+1} modeladas con los estados correspondientes q_{n-1}, q_n, q_{n+1} se añaden dos transiciones llamadas T_n y T_{n+1} y los eventos que las producen e_n y e_{n+1} de la siguiente forma $T_n = (q_{n-1}, e_n, q_n)$, y $T_{n+1} = (q_n, e_{n+1}, q_{n+1})$.

A continuación, se expone un ejemplo utilizando tres registros de tres pacientes con la misma patología y sus correspondientes secuencias de actividades y el autómata probabilista resultante tras su procesamiento:

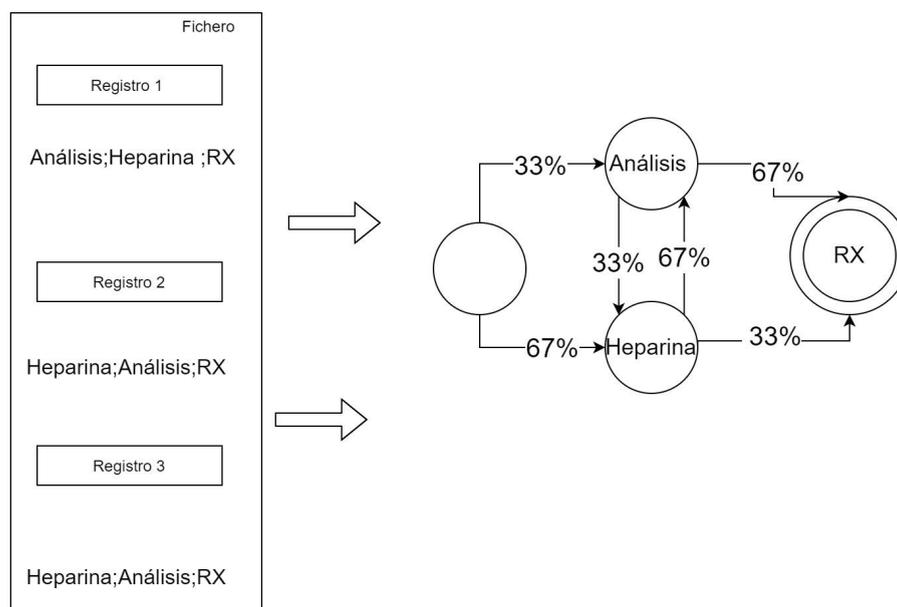


Figura 15: Generación de PFA

Y únicamente. faltaría añadir las probabilidades cada transición T_n que vienen determinadas de la siguiente manera:

$$P(T_n) = Pa(T_n) * (1 - Pe(T_n))$$

- **Pe: Probabilidad de estado final de una actividad** . Esta probabilidad para una actividad $a_{k,n}$ es el recuento de $a1$ como actividad final entre el número total de actividades. En nuestro caso, salvo el alta todas tienen probabilidad de estado final 0.
- **Pa: Probabilidad a priori**. Define la probabilidad a priori de salto. Si $T_n = (q_{n-1}, e_n, q_n)$. $Pa(T_n) = \text{recuento de transiciones } T_n \text{ con origen } q_{n-1}$.

Con los datos introducidos por el personal médico se desarrolló una aplicación en *Python* que lo realiza a través de la generación de listas con las diferentes transiciones. Se utilizaron diccionarios $\{clave, valor\}$ donde la clave sería el nombre de los estados origen y destino separados por ‘;’ y como la probabilidad de T_n explicada anteriormente. Con ello, se obtuvo el siguiente grafo:

Este grafo, es lo que es de valor para el equipo médico ya que permite ver los diferentes caminos posibles ante una determinada patología y generar posteriormente la vía clínica [Anexo 1].

A su vez, resulta de especial interés encontrar aquellos caminos que resultan más o menos probables a la hora de darles el alta por lo que se buscó el camino de mínima y máxima probabilidad.

Como bien es sabido, la probabilidad del camino $c_1 = q_1q_2q_4$ del estado inicial q_1 al estado final q_4 en el autómata finito probabilista de la figura 17 es el producto de las probabilidades de los diferentes probabilidades de las transiciones por la probabilidad de estado final de q_4 , que en este caso es igual a $P_5 = 1$ ya que desde q_4 no hay ninguna transición posible .

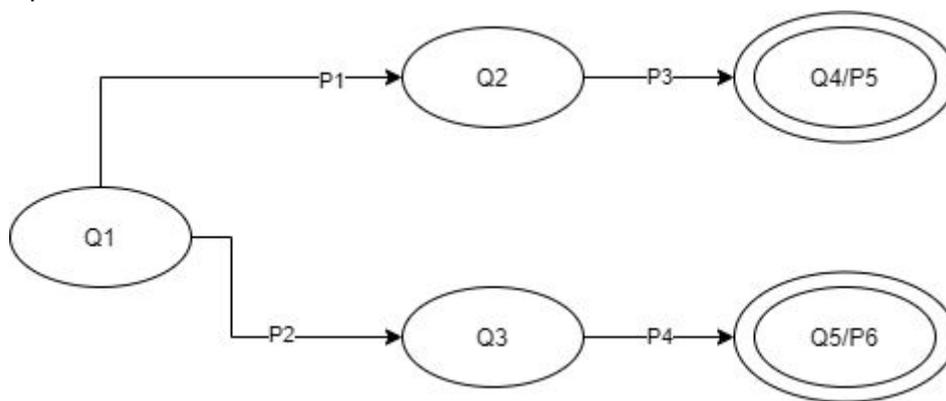


Figura 17: Ejemplificación de camino

$$P(c_1) = P_1 \cdot P_3 \cdot P_5$$

$$P(c_2) = P_2 \cdot P_4 \cdot P_6$$

Para poder utilizar un algoritmo de coste mínimo [Ver anexo 7] se utilizó la siguiente propiedad de la función logaritmo:

$$\log(P(c_1)) = \log(P_1) + \log(P_3) + \log(P_5)$$

$$\log(P(c_2)) = \log(P_2) + \log(P_4) + \log(P_6)$$

Como en el algoritmo de coste mínimo los costes deben ser positivos, y el logaritmo de una probabilidad P_i (un número entre 0 y 1) es un número entre $-\infty$ y 0, se trabajará con $-\log(P_i)$. A continuación se demuestra más detalladamente:

1. Sean P_1, P_2, \dots, P_k las probabilidades de todas las transiciones del autómata finito
2. P_1, P_2, \dots, P_k probabilidades que toman valores en el intervalo (0,1].

3. Sean los logaritmos de estas probabilidades: $l_1 = \log(P_1)$, $l_2 = \log(P_2)$, ..., $l_k = \log(P_k)$.
4. l_1, l_2, \dots, l_k toman valores en el intervalo $(-\infty, 0]$ tales que

$$l_j = -\infty ; \text{ si } P_j = 0$$

$$l_j = 0 ; \text{ si } P_j = 1$$

5. De donde es fácil ver que $-l_1, -l_2, \dots, -l_k$ toman valores en el intervalo $[0, +\infty)$ tales que

$$-l_j = \infty ; \text{ si } P_j = 0$$

$$-l_j = 0 ; \text{ si } P_j = 1$$

6. De esta forma, si se utiliza en autómata finito para obtener el camino de coste mínimo utilizando los costes $-l_1, -l_2, \dots, -l_k$ en las transiciones se obtiene el **camino de probabilidad máxima**.

Para el cálculo del camino de probabilidad mínima hay que continuar con los siguientes pasos:

1. Sea $maxLogProb = \max(-l_1, -l_2, \dots, -l_k)$ que en realidad corresponde al logaritmo de la probabilidad mínima
2. sean los costes $|-l_1 - maxLogProb|, |-l_2 - maxLogProb|, \dots, |-l_k - maxLogProb|$ que toman valores en el intervalo $[0, maxLogProb]$ Tal que

$$|-l_j - M| = 0 \text{ si } p_j = 0$$

$$|-l_j - M| = M ; \text{ si } p_j = \max(p_1, p_2, \dots, p_k)$$

Ahora si se aplica un algoritmo de camino mínimo en el grafo con los costes $|-l_1 - maxLogProb|, |-l_2 - maxLogProb|, \dots, |-l_k - maxLogProb|$. El resultado corresponde al camino de probabilidad mínima. Para agilizar el proyecto se utilizó una implementación del propio algoritmo de *Dijkstra* en *Python* utilizando la librería *Dijkstra* (<https://pypi.org/project/Dijkstra/>) y para la parte visual *Graphviz* (<https://pypi.org/project/graphviz/>).

3.2 Redes de Petri Coloreadas

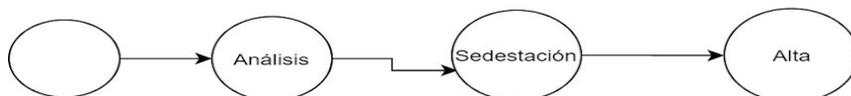
Para la transformación a un modelo de *red de Petri* [Ver Anexo 2] se utiliza como punto de partida el modelo de autómatas finitos probabilistas explicado anteriormente. Atendiendo a las siguientes indicaciones:

- **Transiciones:** Para pasar de un estado a otro se precisará de la creación de una transición con un nombre asociado. El nombre que se le dará será simplemente la concatenación del estado inicial y el final.
- **Recursos:** Para simular el uso de los recursos que se tienen asociados a cada actividad, se crea un lugar en la red de Petri para cada tipo de recurso y un arco en la transición que lleve a aquellas actividades que precisen de dichos recursos, para simular la asignación de los mismos. Posteriormente, en la transición de salida se creará un arco para modelar la liberación del recurso. Un pequeño ejemplo se muestra en la siguiente figura:

Tabla de actividades

Nombre estado	Recurso1	Recurso2
Análisis	Enfermera	Recurso Farmacéutico
Sedestación	Celador	

Autómata finito probabilista



Red de Petri

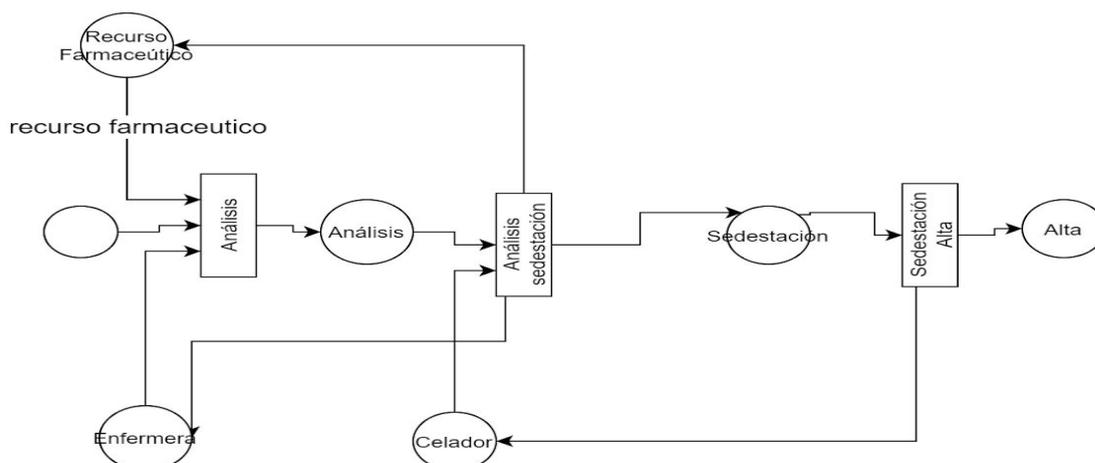


Figura 18 : Generación de red de Petri

- **Token:** Para simular los recursos y los pacientes se deberán de crear tokens y colores asociados a cada recurso además de un color propio para los pacientes. Los *tokens* se pueden crear de cualquier tipo pero en este caso serán cadenas de texto que se denominan: “*Medico1*” , “*Médico2*” , “*Paciente1*” , “*Paciente2*”...

Para poder simular la red de Petri se desarrolló un programa *Python (for_medical.py)* que genere un XML capaz de ser leído por *CPNTools* (<http://cpntools.org/2018/01/09/dtd-for-net-files/>). La motivación de uso de este software es por la capacidad de simulación que este posee ya que una vez introducida la red se le pueden introducir *tokens* con la estructura que interese, como la descrita arriba para simular el sistema ante diferentes escenarios.

Para su creación se han tenido en cuenta las siguientes indicaciones:

- **Colores:** Se creó un color inicial *Paciente* que permitiera crear tokens que representarán los pacientes dentro del sistema. Este proceso se repitió para todos los recursos que se tenían en la base de datos. A su vez, se precisa de la composición de colores cuando en un lugar se juntan varios tipos de tokens por lo que se extrajo de la base de datos la lista de actividades y los recursos que precisan y se crearon composiciones de estos colores para asociarlas posteriormente a los lugares.
- **Lugares:** Se extrajeron la lista de nodos del autómata y una lista con los recursos nuevos para generar la nueva lista de lugares. En el caso de la lista de nodos, se precisó de una lista que asociará a cada lugar la lista de recursos que tendrá para asociar la composición de colores necesaria para el lugar. Esto se hizo cogiendo la tabla *actividades*.
- **Transiciones:** En el autómata las transiciones se guardan utilizando diccionarios $\{clave, valor\}$. Donde la clave es el origen y el destino separado por ‘;’. Simplemente se creó una transición rescatando los lugares cuyo nombre coincide con origen y destino y asociando a la transición el color del origen.

A continuación se muestra una ejemplificación a pequeña escala de como sería el resultado de los colores basándonos en la figura anterior. Se ha utilizado el símbolo *X* para indicar la composición de colores:

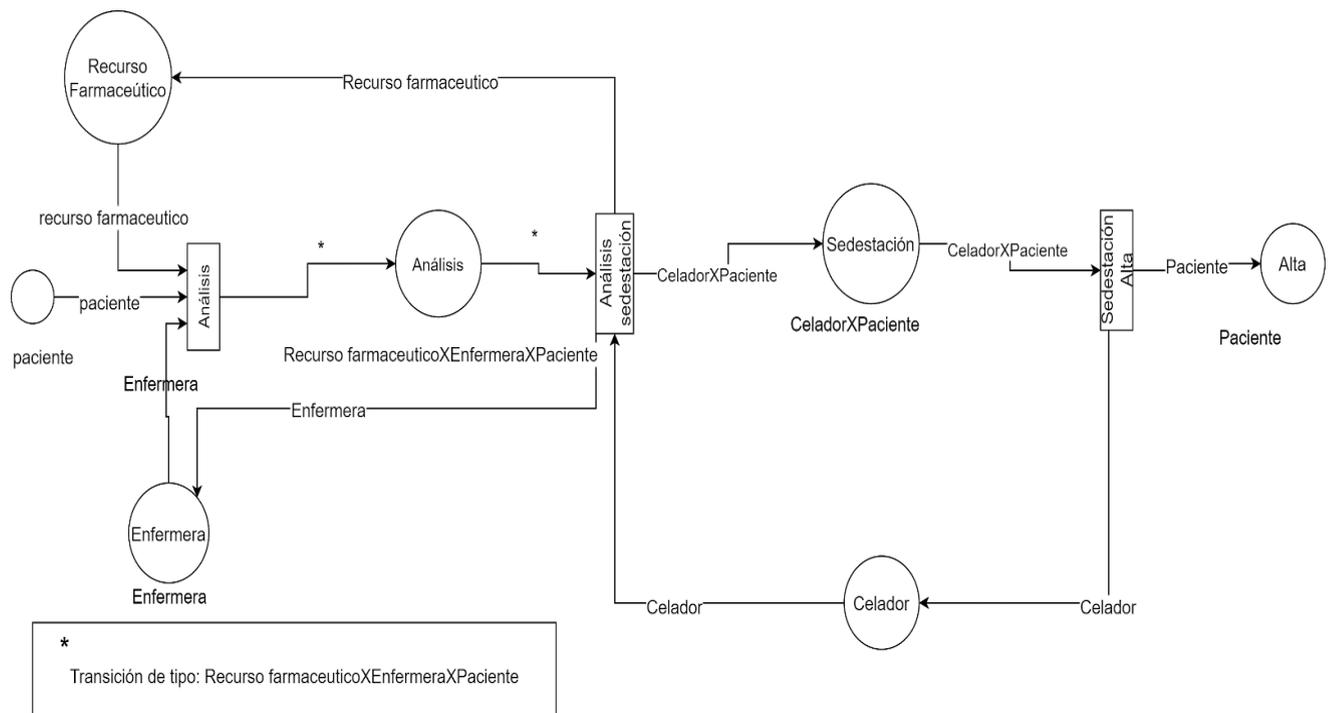


Figura 19: Ejemplo de red de Petri coloreada

El resultado se puede observar el resultado en *rdp_protetiscadera.cpn* dentro del proyecto *GitHub* o en la figura 25. Aunque, en el apartado 4 se pueden observar los problemas encontrados con la visualización en *CPN Tools*.

3.3 Panel de mando

Para realizar un análisis de datos orientado a las necesidades del personal se ha hecho un panel de mandos a través del software de *Power BI*. Este, es un software que permite generar cuadros de mando (<https://powerbi.microsoft.com/en-us/>) a través de diferentes orígenes de datos como tablas *Excel* o bases de datos. El uso de *Power BI* proporciona un cuadro de mando interactivo que permite al personal seleccionar diferentes días de la semana, franja horaria, profesional etc. para actualizar los indicadores que se mostrarán. El personal médico consideró importantes las siguientes métricas:

- 1) **Estancia media hospitalaria:** En base a los pasos que ha seguido el paciente se deberá analizar la media de estancia hospitalaria y analizar los casos anómalos a que es debido ya que después de que a un paciente comience a caminar debería dar el alta al día siguiente.
- 2) **Cirugía Mañana/tarde:** Métrica de especial interés ya que a los pacientes se les debería de atender solo por la mañana. La atención y apertura de un quirófano por las tardes conlleva una serie de gastos asociados debido a que se debe pagar a los trabajadores el plus correspondiente salarial o en días de vacaciones. Se explica más en detalle en las conclusiones.

- 3) **Lado de intervención:** Para analizar qué lado es más propenso a fracturación por parte de la población.
- 4) **ACO:** Análisis de necesidad de la toma anticoagulante. Se deberá analizar debido a que la toma de ACO está asociada a una mayor pérdida sanguínea durante la cirugía.
- 5) **Análisis de hemoglobina y hematocrito previo y posterior a la cirugía:** Indicadores hematológicos para valorar la pérdida sanguínea durante la intervención.
- 6) **Motivo de intervención:** Realización de análisis epidemiológico con aquellas patologías más frecuentes en la población.
- 7) **Origen de los pacientes y edad media:** Debido a que el hospital tiene asignado áreas rurales y con el objetivo de desarrollar un análisis demográfico se analizarán los pacientes que provienen del área rural con los del área urbana.
- 8) **TX:** Necesidad de transfusión de sangre durante la cirugía. Dato de especial interés ya que como se observará más adelante en el informe. El 100% de los pacientes no precisaron de transfusión. Sin embargo, el equipo médico debe valorar el tener reservas en las operaciones sin complicaciones debido a que el simple hecho de reservar sangre conlleva un gasto hospitalario de en torno a 150 euros.
- 9) **Riesgo anestésico:** Análisis del riesgo con el que acuden los pacientes a la intervención. Es muy importante para relacionar problemas operatorios a las patologías previas, la edad y el riesgo anestésico del paciente.
- 10) **Patologías previas:** Para realizar un posterior análisis por parte de los facultativos para una relación entre el éxito de la intervención y la situación del paciente.

A través de los datos extraídos del apartado 2.3.1, se generó el siguiente panel de mandos:

salud
INDICADORES

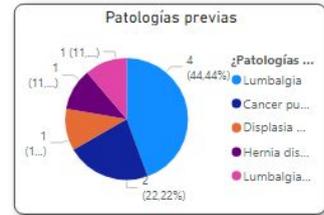
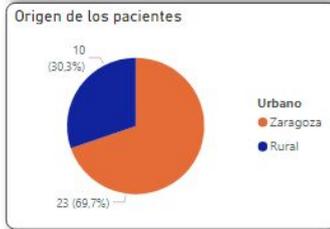
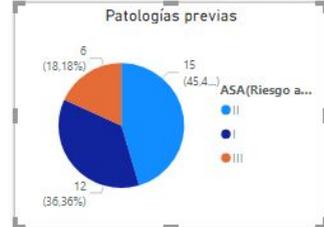
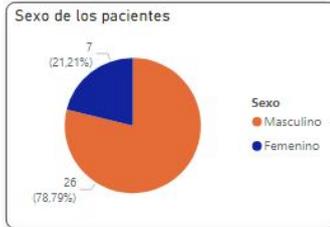
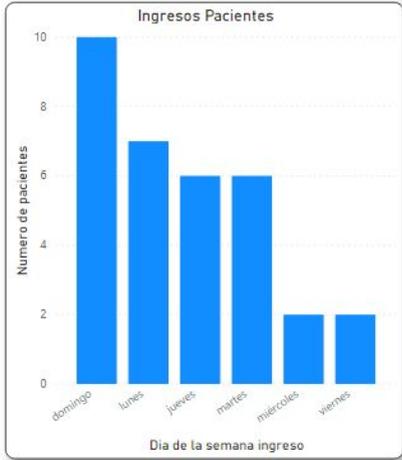
Pacientes atendidos
33

Mediana de edad
56

Mediana de estancia en días
5

Cirujanos Disponibles
8

Pacientes con patologías previas
11



salud
INDICADORES

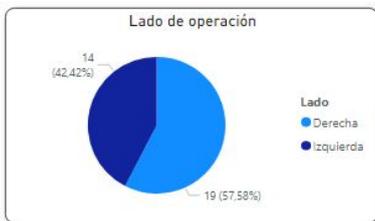
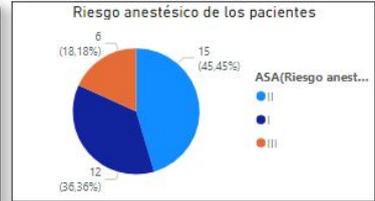
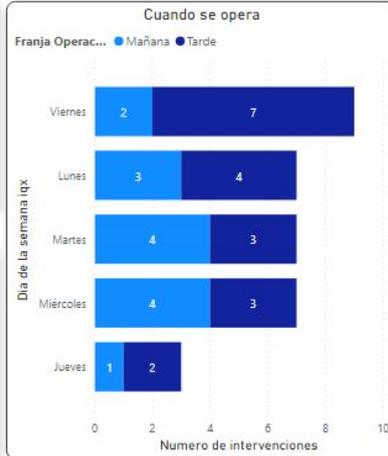
Media de HB PRE
15,38

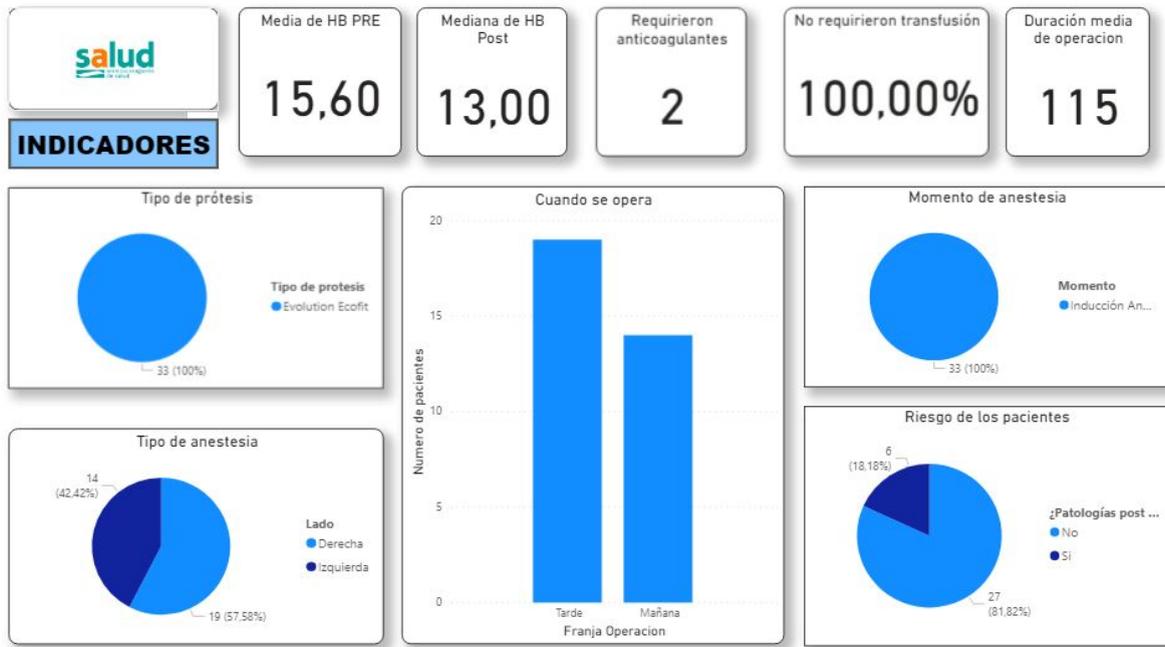
Mediana de HB Post
13,00

Tomaron anticoagulantes
2

No requirieron transfusión
100,00%

Duración media de operación
115





Redon primer día	47 %
RX primer día	96,88%

Accion	Mediana de Dia
Alta	4
Analitica	1
Antibiotico	1
Curar Herida	2
Empezar a caminar	2
Heparina	1
Momento caminar muletas	2
Momento retirar Redon	2
Reposo	3
RX	1
Sedestacion	1

Accion	Promedio de Dia
Alta	4,00
Analitica	1,00
Antibiotico	1,00
Curar Herida	1,52
Empezar a caminar	2,33
Heparina	1,00
Momento caminar muletas	2,36
Momento retirar Redon	1,53
Reposo	3,54
RX	1,06
Sedestacion	1,21

Figura 20: Paneles de mando BI

Este panel se encuentra embebido dentro de la página web en el apartado de *minería de datos*. Sin embargo, al trabajar con la cuenta proporciona con *Unizar* de ámbito académico solo permite visualizarlo iniciando sesión con dicha cuenta. Para poder embeber el panel de forma pública se precisaría de una cuenta *Pro* asociado a un correo profesional. Por ello, en la propia web, se ha facilitado la descarga en el apartado de minería de datos o en el código de *Github* en */extra/informe.pbix*

4. Despliegue

El objetivo de la aplicación web ha sido que pueda ser usada por los médicos de forma remota. Por ello se ha decidido que se despliegue en un servicio web en la nube. La plataforma elegida ha sido *Google Cloud* (<https://cloud.google.com/>) debido a que ofrece máquinas con un amplio crédito de uso a estudiantes para su uso y despliegue. La configuración de dicha máquina ha sido sencilla ya que se ha dividido en los siguientes pasos:

1. Se ha creado una máquina *host Linux Ubuntu* para despliegue del servidor *Django* a través del servicio *Computer Engine*. Debido a que el servidor no tendrá excesiva carga de trabajo se ha elegido la configuración más simple utilizando únicamente un *CPU* y *3,75 GB* de almacenamiento como se observa a continuación.

```
ID de instancia
5894470538600203885

Tipo de máquina
n1-standard-1 (1 vCPU, 3,75 GB de memoria)
```

Figura 21 : Máquina Cloud

2. El envío de código y posteriores revisiones del despliegue se ha utilizado el sistema de gestión de proyectos y versiones de *Github* importando el código de la máquina con el link descrito anteriormente.
3. Para la instalación de paquetes se puede utilizar el comando *pip freeze* para sacar todos los paquetes que se han instalado en local y posteriormente instalarlos en la máquina virtual. En este caso se sacó la salida de ese comando a un fichero y se hizo posteriormente un *install* leyendo de este fichero.
4. Una vez instalados los paquetes adecuados, se lanzó el servidor a través del puerto 80 para atender peticiones *HTTP*.
5. Para sustituir la IP de acceso al servicio, se obtuvo un dominio gratuito de *Freenom* (<https://www.freenom.com/en/index.html?lang=en>), *hlozablesa.tk*, que permite un acceso a la página más intuitivo que la IP de la máquina. Se tuvo que crear un espacio de nombres con el dominio adquirido como se puede observar:

Cloud DNS [+ CREAR ZONA](#)

Zonas [Políticas del servidor DNS](#)

Las zonas de DNS permiten definir tu espacio de nombres. Puedes crear zonas públicas o privadas. [Más información](#)

Filtrar recursos

<input type="checkbox"/>	Nombre de la zona ^	Nombre de DNS	DNSSEC	Descripción	Tipo de zona
<input type="checkbox"/>	hlozablesa	http://www.hlozablesa.tk.	Off ▾		Pública

Figura 22 : Espacio de nombres Google Cloud

6. Por último, una vez todo configurado adecuadamente se debió de poblar la base de datos con el script descrito en el apartado 2 y la creación de usuarios a través de la utilidad de administración que proporciona *Django*, *Django-admin*.

El resultado del despliegue es la aplicación que se puede acceder a través de:

<http://www.hlozablesa.tk/>

Pudiendo acceder con lo credenciales creados para *Jorge Albareda*:
https://docs.google.com/document/d/1FteBF10tZcX44K_Ecw_8NHqf1HMT1KbFaeY_4r4sP-M/edit?usp=sharing

5. Problemas encontrados

A lo largo de este proyecto, se han desarrollado diversos problemas asociados a la coordinación y gestión de un sistema de la información. A continuación, se numeran los más importantes:

1. **Covid-19:** El efecto de la pandemia, ha afectado más allá de tener que gestionar las reuniones a distancia. En este proyecto ha repercutido directamente en la disponibilidad de los facultativos y los procedimientos y espacios para ir al hospital a tener reuniones.
2. **Problemas de coordinación y gestión:** La obtención de historias clínicas, ha sufrido retrasos para la obtención de los 33 registros con los que se ha trabajado. Además, la necesidad de tener que trabajar en colaboración ha provocado problemas de inanición provocando que el proyecto parara durante unas semanas. Otro aspecto importante ha sido los malentendidos a tanto a la hora de explicar qué se necesitaba y cómo por parte del técnico de desarrollarlo, provocando tener que rehacer partes.
3. **Información desubicada:** Algunos registros se encuentran desubicados o en ubicaciones mixtas. Por un lado, se encontraban registros que estaban en

la plataforma de historial electrónico o en papel. Por el otro, para acceder a los valores de indicadores como el hematocrito previo y posterior a la intervención se precisó de conexión a la plataforma digital del SALUD.

4. **CPN tools:** El uso de CPN Tools como software de visualización y simulación ha provocado problemas en la visualización debido a que el programa se encuentra desactualizado (última versión 2018) y presenta problemas y ralentizaciones de carga al trabajar con redes demasiado grandes.

6. Resultados

Como ya se ha descrito en el apartado 3.1 se ha obtenido el autómata que registre todos los estados posibles en la patología. En base a dicho autómata, se ha analizado cómo obtener los caminos más y menos probables de actuación para la artroplastia de cadera. A continuación, se muestran ambos autómatas resultantes:

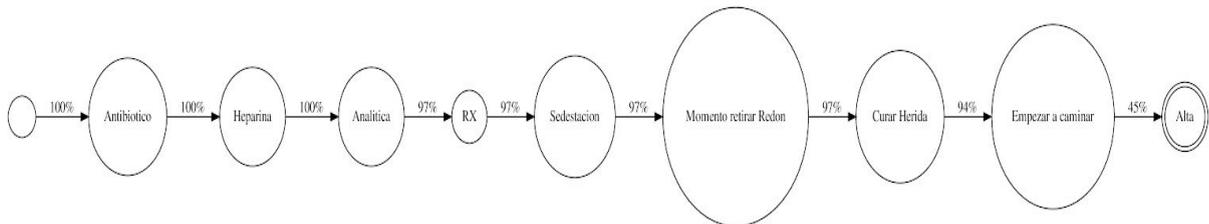


Figura 23 :Camino de máxima probabilidad³

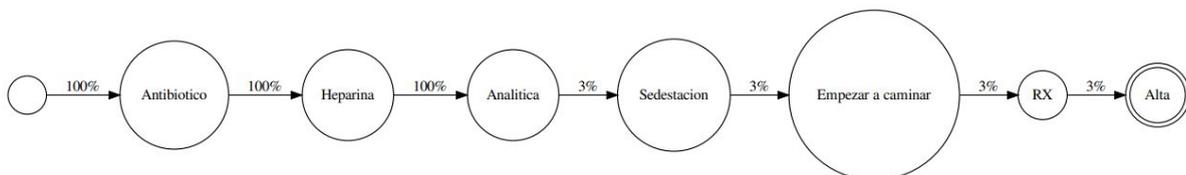


Figura 24 :Camino de máxima mínima⁴

En el apartado de la *red de Petri* se puede descargar un fichero para simular la *red de Petri* en *CPN Tools* pudiendo añadir *tokens* de tipo string en el inicio de la red para simular pacientes y *tokens* de tipo string en los diferentes recursos para simular la cantidad de recursos que tiene el sistema hospitalario. Sin embargo, debido a las limitaciones de

³ Descargable en la página web apartado minería de datos como camino de probabilidad máxima

⁴ Descargable en la página web apartado minería de datos como camino de probabilidad mínima

CPN Tools no soporta todos los elementos gráficos. A continuación, se muestra parte de la red en el software.

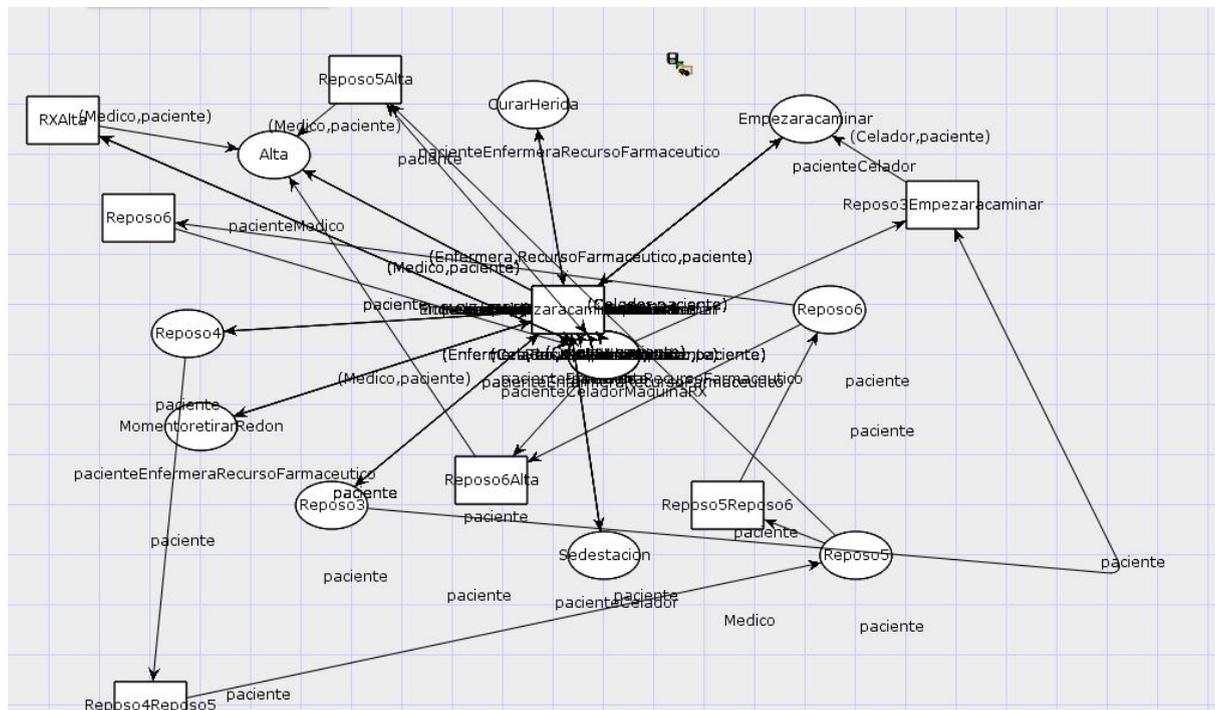


Figura 25 : red de Petri en CPN Tools

En base a los paneles del apartado 3.3 , se ha muestra el siguiente análisis de los datos [Ver Anexo 3]:

→ Análisis de datos epidemiológicos:

Se realiza un estudio descriptivo de pacientes intervenidos de prótesis total de cadera durante los años 2019/2020. Se obtienen un total de 33 pacientes intervenidos de prótesis totales de cadera no cementadas tipo Evolution/Ecofit con una mediana de edad de 56 años. El 78,79% de los pacientes eran del sexo masculino y la cadera más frecuente intervenida fue la derecha (57,58%). La causa más frecuente de cirugía fue la coxartrosis (81,82%), seguido de la necrosis avascular y de otras como la enfermedad de perthes. La mediana de la estancia hospitalaria fue de 5 días. 19 pacientes fueron intervenidos en turno de tarde y 14 pacientes en turno de mañana. El 70% de los pacientes intervenidos pertenecían a la ciudad de Zaragoza mientras que el 30% pertenecían a la zona rural de la provincia de Zaragoza. Todos los pacientes fueron intervenidos por 8 especialistas en cirugía ortopédica y traumatología.

→ Datos de manejo intrahospitalarios:

El tiempo medio de la intervención quirúrgica fue de 115 minutos, entendiendo el tiempo que transcurre desde que el paciente entra en el quirófano y se aplica la anestesia y cirugía correspondiente hasta que el paciente sale del quirófano. El 100% de los pacientes

recibió una dosis previa a la intervención quirúrgica y dos dosis post-intervención. También el 100% se reinició la heparina a dosis profilácticas a las 12 horas post-intervención y se mantuvo el tratamiento diario durante la estancia hospitalaria. A todos los pacientes se les realizó una radiografía post-intervención, algunos de ellos al primer día post-intervención (96,88%) y a otros (3,22%) en los días sucesivos. Esta variación se debe a que algunos pacientes son intervenidos el viernes y las radiografías no se realizan de forma habitual el sábado, pasando a realizarse el lunes.

Al 47% de los pacientes se les retiró el redón en el primer día postoperatorio mientras que al 53% al segundo día post-operatorio.

A todos los pacientes se les realizó una analítica post-operatoria para valorar la anemia aguda post-intervención (pérdida de sangre tras la intervención) y valorar la posibilidad de transfusión sanguínea si dicha pérdida era muy cuantiosa. Todo ello se mide a través del parámetro de la hemoglobina, *HB*. La hemoglobina media antes de someterse a una intervención fue de 15,38 mg/dl y tras la intervención descendió hasta 13 mg/dl. En ningún paciente hubo un descenso por debajo de 8 mg/dl por lo que no se transfunde a ningún paciente.

En cuanto a las complicaciones durante la estancia hospitalaria se observaron dos casos; uno de un síndrome febril y otro de un cuadro de gota en rodilla que no tuvieron relación con la cirugía pero sí incrementaron el tiempo de la estancia hospitalaria. La revisión al mes se observaron 3 casos de complicaciones relacionadas con la cirugía, una celulitis, molestias en la cadera con la deambulación y otro con sensación subjetiva de limitación funcional de la cadera.

7. Conclusiones

El desarrollo del sistema ha permitido un adecuado muestreo de información al equipo médico a través de un servicio web. A nivel de infraestructura, se ha conseguido desplegar un ecosistema web a través del desarrollo de la base de datos, aplicación web y despliegue de un sistema escalable y modificable a las necesidades del hospital. Esto permitirá en un futuro la aplicación a más áreas y departamentos lo que permitiría agrupar toda la información de forma digital para la extracción de conclusiones y garantizando la trazabilidad de la misma. Un futuro foco muy destacable sería el desarrollo de la red de *Petri* y la simulación hospitalaria ya que esta puede provocar muchas optimizaciones en el sistema.

Por otra parte, se han llegado a las siguientes conclusiones tras analizar los datos expuestos en el apartado anterior:

1. En los resultados se observa que en ningún caso fue necesaria la transfusión postoperatoria de sangre. En la actualidad, un acto que se realiza previo a la intervención quirúrgica es la reserva de sangre de cada paciente de su mismo grupo

sanguíneo para evitar complicaciones por un sangrado excesivo durante la cirugía o posterior a ella. Este acto tiene un coste aproximado de 150 euros. Se ha visto que en ningún caso ha sido necesario transfundir, por lo que sería importante plantearse si realmente es la reserva de sangre previa a intervención quirúrgica es necesaria ya que supone un gasto económico que luego no es útil.

2. El objetivo final de la implantación de una prótesis es que el paciente deambule sin dolor. Se ha visto que algunos pacientes tras comenzar a deambular continúan con el ingreso hospitalario durante más días (entre 1 y 4 días) por diversos motivos; médicos y no médicos como pueden ser: un manchado excesivo de la herida quirúrgica o no dar de alta por ser fin de semana. Una guía clínica elaborada podría ser una solución para establecer que una vez que el paciente camine, el paciente debería ser dado de alta. Ello supondría un ahorro económico importante, ya que cada día de ingreso supone un gasto económico medio de unos 600 euros.
3. El retraso en la realización de la radiografía implica que el paciente comience más tarde con la deambulación, ya que no puede empezar hasta tenerla hecha. Esto supone alargar la estancia hospitalaria y con ello, incrementar el gasto sanitario por lo que en una guía clínica adecuada debería aparecer la realización de la radiografía al día siguiente de la intervención. Esto es una medida de ahorro económico.
4. La informatización de la historias de los pacientes a través de la historia clínica electrónica y de este trabajo ha supuesto un registro ordenado sin pérdida de ninguna información de los datos de los pacientes. Además, también ha supuesto una medida adicional que evita el contagio de infección por COVID entre profesionales sanitarios.

8. Bibliografía

- [1] Aalst, W., 2011. *Process Mining : Discovery, Conformance And Enhancement Of Business Processes*. 1st ed. Heidelberg: Springer-Verlag, pp.1-65.
- [2] Bernardi, S., Mahulea, C. and Albareda, J., 2019. Toward a decision support system for the clinical pathways assessment. *Discrete Event Dynamic Systems*, 29(1), pp.91-125.
- [3] Cavarra, A., Riccobene, E. and Scandurra, P., 2004. A framework to simulate UML models. *Proceedings of the 2004 ACM symposium on Applied computing - SAC '04*,.
- [4] Viard, K., 2018. *Modelling and Recognition of Human Activities of Daily Living in a Smart Home. Automatic Control Engineering*. Université Paris-Saclay; Politecnico di Bari. Dipartimento di Ingegneria Elettrica e dell'Informazione (Italia) , pp 45-52.
- [5] Ciro, I., Basile, F. and Mahulea, C., 2017. *Modeling And Analysis Of Healthcare Systems Based On Clinical Pathways*. Master thesis. University of Salerno.
- [6] Novellón, M., Mahulea, C. and Albareda, J., 2018. *Búsqueda De Indicadores Médicos Mediante El Análisis De Datos Y Mejora De Los Modelos Matemáticos Existententes Aplicado A Las Vías Clínicas*. Graduado. Universidad de Zaragoza.
- [7] Jensen, K. and Kristensen, L., 2009. Coloured Petri Nets.
- [8] Bucholz R., 2014. Indicaciones, técnicas y resultados de reemplazo total de cadera en estado unidos. *Rev Med. Clin. Condes.*; pp.760 - 764
- [9] Lopez-Herranz G, Torres-Gomez O.,2017. Variabilidad de la clasificación del estado físico de la Sociedad Americana de Anestesiólogos entre los anestesiólogos del Hospital General de México. *Rev. Mex. Anest.*; pp 190-194.
- [10] Torrubia, G., & Terrazas, V. (2012). Algoritmo de Dijkstra. Un tutorial interactivo. *VII Jornadas de Enseñanza Universitaria de la Informática (JENUI 2001)*.

Anexos

Anexo 1: Vías clínicas

Las vías o guías clínicas [6] son los documentos elaborados conjuntamente entre el equipo médico y la dirección del hospital, que definen de forma detallada el tratamiento que ha de seguir el conjunto de los pacientes tratados por la misma dolencia.

Lo que pretende como su propio nombre indica es que sobre un proceso muy frecuente se realice a todos los pacientes el mismo tratamiento, exactamente el mismo realizando los mismos pasos en todos pretendiendo utilizar con precisión los recursos materiales y humanos reduciendo al máximo la posibilidad de complicaciones. Se define como hacer siempre todo lo que hay que hacer, pero nada más que lo que hay que hacer. Se denomina vía porque pretende definir y llevar el tratamiento de cada paciente como por una vía o rail. Que nadie haga lo que se le ocurra en cada momento sino seguir una vía o camino que le indique lo que hay que hacer en cada caso. Solo sirve para casos muy frecuentes, pues sobre casos esporádicos o excepcionales no puede crearse una vía clínica ya que no se tiene experiencia en el tratamiento complicaciones, recursos necesarios etc. Así mismo, sirven como protección para el personal médico frente a posibles demandas por mala praxis e, igualmente, permiten dar información más detallada a los pacientes de los tratamientos que van a seguir y los progresos que está previsto que tengan.

El proceso de elaboración de una vía clínica es bastante complejo, ya que, a la hora de modelarlas, cada hospital ha de tener en cuenta el periodo de tiempo que estas han de cubrir y las condiciones específicas de su zona de actuación. Las vías clínicas abarcan un periodo de tiempo que va desde el ingreso hasta que el paciente recibe el alta, contemplándose también en ellas, el tratamiento que habrá de seguir una vez abandonado el centro hospitalario. En el caso particular del Hospital Clínico “*Lozano Blesa*”, se debe tener en cuenta que un gran número de pacientes procede de zonas rurales más o menos alejadas de la ciudad de Zaragoza. Esta circunstancia conlleva que la duración de estas vías sea ligeramente superior a la que se da en otros hospitales para los mismos procedimientos, dado que es necesario poder cerciorarse de que no se presenta ningún tipo de complicación derivada de la intervención. Este aumento del plazo fijado para la vía evita posibles traslados sucesivos del paciente, ya que, de presentarse complicaciones, este puede tener que someterse a un segundo ingreso a pesar de haber sido dado inicialmente de alta.

A continuación se muestra una vía clínica para una artroplastia primaria de cadera

	<p align="center">PLAN DE MEJORA DE LA CALIDAD</p> <p align="center">Servicio de Cirugía Ortopédica y Traumatología</p>	<p align="center">"HOSPITAL CLINICO UNIVERSITARIO "LOZANO BLESA" Avda. San Juan Bosco, 15 50009-ZARAGOZA</p>
---	---	--

PROTOCOLO-GUIA DE PRACTICA CLINICA
ARTROPLASTIA PRIMARIA DE CADERA

Elaborado	Elaborado	Aprobado
Fecha: Octubre 2013	Fecha: Octubre 2013	Fecha: Diciembre 2013
Jefe de Servicio	Responsable de Calidad	Comisión de Dirección del Servicio
Jorge Albareda Albareda	Jose M. Lasierra Sanromán	
Revisión :		A

	<p align="center">PLAN DE MEJORA DE LA CALIDAD</p> <p align="center">Servicio de Cirugía Ortopédica y Traumatología</p>	<p align="center">"HOSPITAL CLINICO UNIVERSITARIO "LOZANO BLESA" Avda. San Juan Bosco, 15 50009-ZARAGOZA</p>
---	---	--

DIAS PREVIOS AL INGRESO

- ✓ Verificar documentación completa procedente de la consulta
- ✓ Verificar que ha sido valorado por Anestesia y posibles incidencias
- ✓ Verificar si tiene firmado el Consentimiento Informado

DIA DEL INGRESO

- ✓ Verificar documentación completa procedente de la consulta
- ✓ Verificar que ha sido valorado por Anestesia
- ✓ Verificar si tiene firmado el Consentimiento Informado
- ✓ Valorar la existencia de procesos patológicos intercurrentes durante su periodo de LEQ (incluidas nuevas alergias)
- ✓ Atención especial ante infecciones urinarias en las 4-6 semanas previas
- ✓ Realizar en su caso el estudio preoperatorio urgente
- ✓ Verificar si está incluido en programa de Autotransfusión
- ✓ *En caso de no autotransfusión iniciar VENOFER 200 mg IV (que se repetirá a las 48 horas)*

DIA DEL INGRESO

- ✓ Solicitar las Unidades establecidas al Banco de Sangre
- ✓ Pautar Profilaxis Antibiótica
- ✓ Pautar Profilaxis con Heparina de BPM
- ✓ Señalar-Marcas el lado que se va a intervenir (rotulador indeleble)
- ✓ El paciente se duchará un mínimo de 6 horas antes de la IQ

	<p align="center">PLAN DE MEJORA DE LA CALIDAD</p> <p align="center">Servicio de Cirugía Ortopédica y Traumatología</p>	<p align="center">"HOSPITAL CLINICO UNIVERSITARIO "LOZANO BLESAS"</p> <p align="center">Avda. San Juan Bosco, 15 50009-ZARAGOZA</p>
---	---	---

DIA DE LA INTERVENCIÓN

- ✓ Comprobar que el paciente baje a quirófano con la Hª Clínica Completa
- ✓ Comprobar preparación por Enfermería antes de bajar a quirófano
- ✓ Verificar que se acompaña la dosis correspondiente de Profilaxis AB
- ✓ Comprobar la correcta identificación del paciente
 - Postcirugía
 - Control de Analgesia
 - Control de Analítica
 - Vigilancia de drenajes/diuresis/trofismo distal
 - Asegurarse de que lleva oxígeno en gafas nasales

1º DIA POSTOPERATORIO

- ✓ Verificar controles analíticos
- ✓ Retirar Sonda Urinaria y Drenaje/s
- ✓ Comprobar cumplimiento de protocolo de Quimioprofilaxis
- ✓ Mantener 24 horas la oxigenoterapia en gafas nasales
- ✓ Levantar de la cama y sentar en silla alta (no sillón)
- ✓ Solicitar Rx de control
- ✓ 2ª dosis de VENOFER 200 IV

2º DIA POSTOPERATORIO

- ✓ Retirada de vía venosa
- ✓ Realizar revisión-cura de la herida
- ✓ Autorizar "primeros pasos" con bastones ingleses / andador
- ✓ Comprobar disponibilidad de suplemento para WC

	<p align="center">PLAN DE MEJORA DE LA CALIDAD</p> <p align="center">Servicio de Cirugía Ortopédica y Traumatología</p>	<p align="center">"HOSPITAL CLINICO UNIVERSITARIO "LOZANO BLESA" Avda. San Juan Bosco, 15 50009-ZARAGOZA</p>
---	---	--

3º DIA POSTOPERATORIO

- ✓ Deambulación progresiva con bastones ingleses
- ✓ Comprobar / Reclamar Rx de control

4º DIA POSTOPERATORIO

- ✓ Además de lo correspondiente al día 3º
- ✓ Ver Rx de control
- ✓ Valorar si cumple criterios para alta en 24 horas

5º DIA POSTOPERATORIO

- ✓ Revisión herida quirúrgica
- ✓ Alta en Hospital (salvo contraindicación)
- ✓ Cita para retirar agrafes de sutura en las consultas externas / Centro de Salud
- ✓ Citación para revisión en consultas externas en el plazo de 4-6 semanas.

Figura 26 : Vía Clínica artroplastia de cadera [6]

Anexo 2: Redes de Petri y redes de Petri coloreadas

Antes de hablar de redes de Petri Coloreadas se debe hablar de redes de Petri [1], una de las formas más consolidadas de modelado de procesos. Estas han sido diseñadas para el análisis de la concurrencia.

Una red de Petri es una tripleta $N = (P, T, F)$ donde P es un conjunto finito de lugares, T es otro conjunto finito de transiciones tal que $P \cap T = \emptyset$ y F es el conjunto finito de arcos dirigidos tal que $F \subseteq (P \times T) \cup (T \times P)$. Además una red de Petri marcada es aquella definida por el par (N, M) , donde $N = (P, T, F)$ y M es una función que asigna a cada lugar un número positivo de tokens. Un ejemplo de red de Petri marcada se puede ver a continuación:

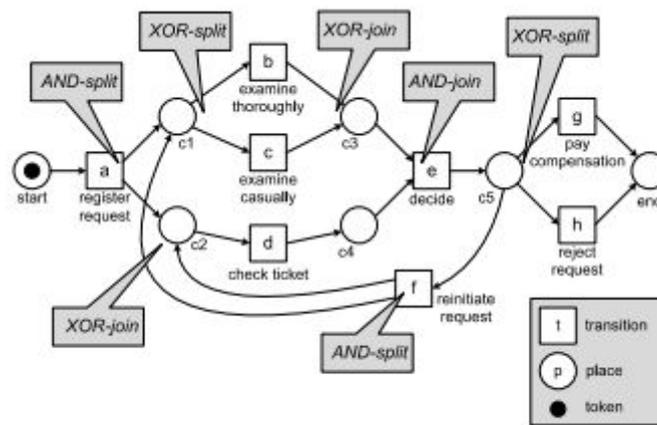


Figura 27 : red de Petri

Por otro lado tenemos las redes de Petri Coloreadas[7] CPN siendo estas una extensión de las redes de Petri. Estas redes se pueden definir de la siguiente manera:

$$N = (P, T, A, \Sigma, C, N, E, G, I)$$

Como se ha descrito anteriormente P representa los lugares, T las transiciones y A los arcos dirigidos siendo los conjuntos disjuntos entre sí: $P \cap T = P \cap A = T \cap A = \emptyset$. A esto se le añaden las siguientes características:

- Σ Es el conjunto de todos los colores, es decir de todas las *marcas* posibles dentro de nuestra red.
- C Es la función de *coloreado*. Coloca en los lugares P las marcas correspondientes del conjunto Σ .
- N corresponde a la función de nodos. Coloca A en el conjunto $(P \times T) \cup (T \times P)$.
- E corresponde a la función de los arcos. Coloca en cada arco $a \in A$ en la expresión e . La entrada y salida de datos en la función de los arcos deben ser del mismo tipo.
- G es la función de guarda. Transforma cada transición $t \in T$ en una guarda g . Esta expresión se evalúa como verdadero o falso.

→ I es una función de inicialización. Aplica a cada lugar p una expresión de inicialización i para colocar las marcas en él.

Esta extensión de las redes de Petri otorga a la red una mayor capacidad al poder tener cualquier tipo de marca en el sistema y el poder tener funciones de evaluación *guardas* para las transiciones o funciones en los arcos para complementar la transformación de los diferentes *tokens*. A continuación, se adjunta un ejemplo⁵

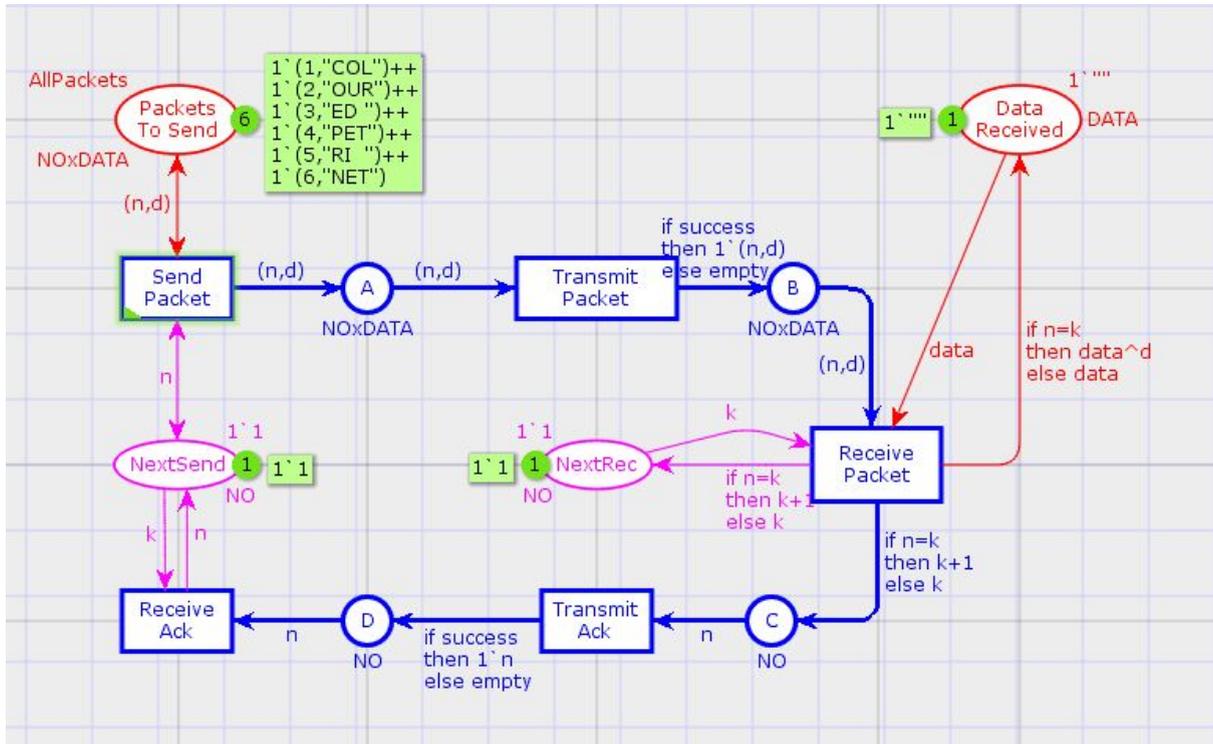


Figura 28 : red de Petri coloreada en el software CPN Tools

En nuestro campo de estudio resultan especialmente útiles para analizar la gestión de recursos hospitalarios. Tienen la ventaja aparte de tratarse de un modelo formal que permiten ser ejecutadas y analizadas con diferentes sistemas software como CPN Tools. Este software se puede dividir en tres zonas para su explicación:

⁵ Práctica 2 asignatura de evaluación y control de sistemas, Universidad de Zaragoza

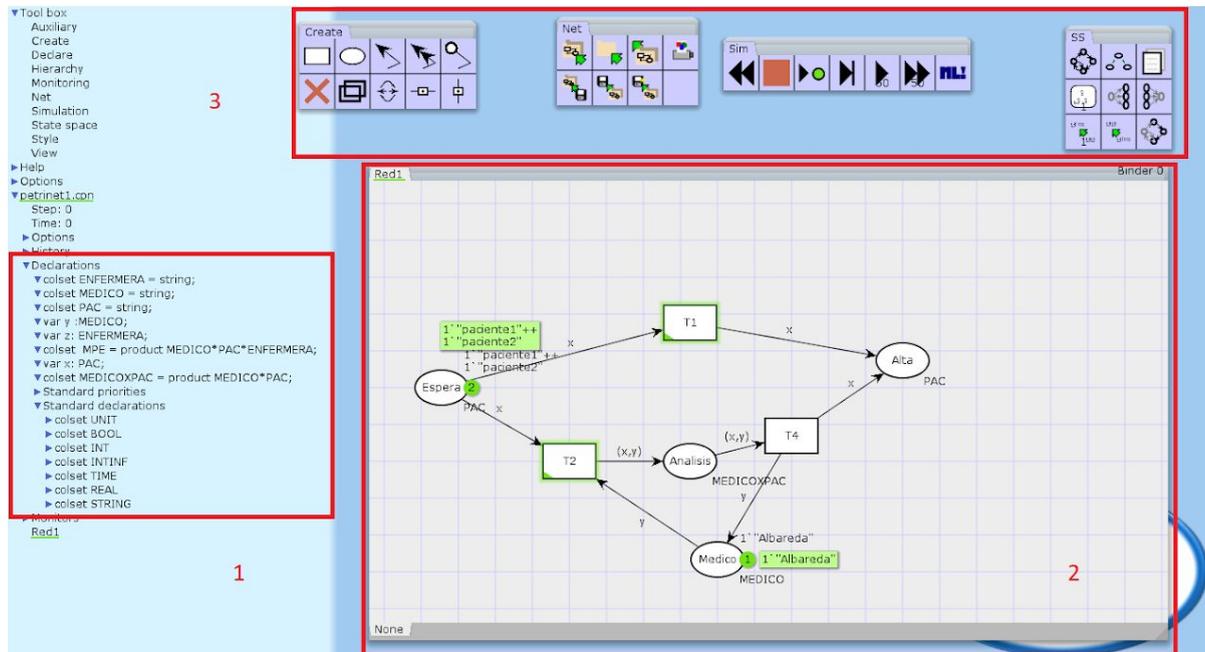


Figura 29 : red de Petri coloreada en el software CPN Tools

En la zona uno se pueden observar los diferentes colores que se han generado y las variables asociadas a dichos colores con las que se trabajará en la red.

En la zona dos se observa el lienzo de la red de Petri. En ella se deberá trabajar con los elementos de creación de elementos de la zona 3. Además de ello, se podrá colocar los token adecuados en los nodos que se desee, asignado el número y el tipo que se desee.

En la zona tres se observan las diferentes herramientas con las que se puede trabajar para dibujar la red. Se pueden distinguir cuatro recuadros con funciones:

- **Primer recuadro:** Se pueden encontrar utilidades para crear los nodos, transiciones y arcos.
- **Segundo recuadro:** Se encuentran las utilidades para carga y descarga de redes de Petri en el sistema.
- **Tercer recuadro:** Herramientas de simulación. Dónde se pueden ejecutar las diferentes transiciones del sistema. En la zona dos se puede observar en verde aquellas transiciones que se pueden ejecutar debido a que contienen la cantidad de *token* adecuada para ser ejecutada.
- **Cuarto recuadro:** Utilidades de la red de Petri. Permite generar el espacio de estados, nodos sucesores y reportes entre otras utilidades.

El objetivo en este trabajo será la automatización de este proceso, generado automáticamente el fichero para su importación. Para ello, nos hemos basado en la estructura XML (<http://cpntools.org/2018/01/09/dtd-for-net-files/>) de los ficheros que genera CPN Tools.

Anexo 3: Apéndice médico

Anexo 3.1 ¿Qué es una prótesis de cadera? ¿Por qué hay que poner una prótesis de cadera?

La indicación de implantar una prótesis de cadera en un paciente viene definida por la presencia de un dolor moderado-grave de cadera que no se controla con medidas conservadoras; reposo y medicación analgésica[8]. Las causas de dolor en dicha cadera pueden ser muy diversas siendo las más frecuentes la artrosis (en la cadera se denomina coxartrosis), necrosis avascular u otras enfermedades menos frecuentes que deforman la articulación de la cadera y produce dolor como pueden ser la enfermedad de *Legg Calve Perthes* entre otras. Las prótesis de cadera permiten a los pacientes una deambulación sin dolor con muy buenos resultados a largo plazo.

Existen diversos tipos de prótesis, con diferentes características que se adaptan a las necesidades del paciente. En el presente estudio se han seleccionado prótesis de cadera de tipo *Evolution Ecofit*. Se tratan de prótesis totales no cementadas, es decir, se realiza una sustitución de las dos partes que componen la articulación de la cadera; el acetábulo (o cotilo) y el fémur. Es un tipo de prótesis donde los propios componentes se fijan biológicamente al hueso.

- El acetábulo de la pelvis es sustituido por una cúpula de titanio rugoso (llamado cotilo) que se integra en el propio acetábulo del paciente.
- El fémur proximal del paciente es sustituido por un vástago de titanio recubierto de hidroxiapatita la cual permite la fijación biológica en el hueso. A dicho vástago se une una cabeza femoral de titanio.
- Entre el cotilo y la cabeza femoral se coloca un polietileno que permite la movilidad entre ambos componentes. Todo ello está reflejado en la siguiente imagen:

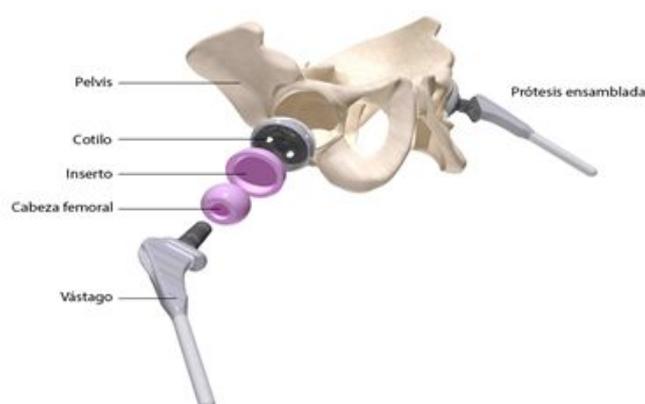


Figura 30 : Ejemplo de prótesis de cadera
(<https://www.mba.eu/blog/tipos-de-protesis-de-cadera/>)

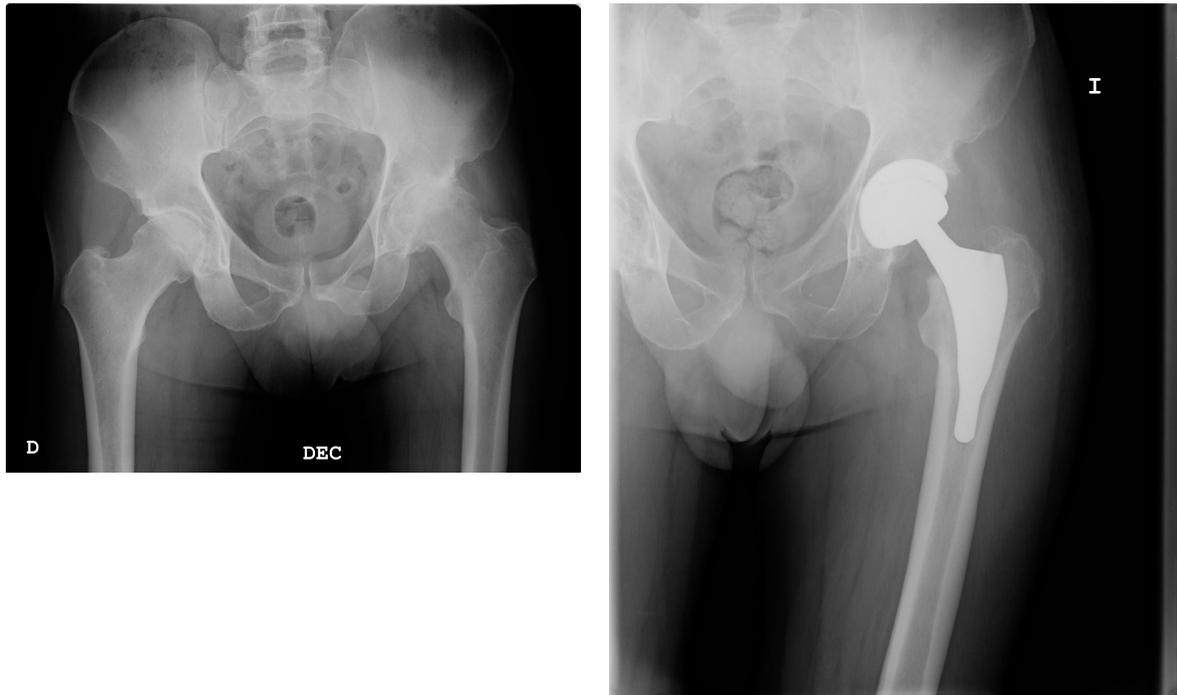


Figura 31 : Radiografía de paciente diagnosticado de artrosis de cadera izquierda (imagen 1) y radiografía de control tras implantación de prótesis total de cadera izquierda (imagen 2).

Anexo 3.2 Proceso hospitalario de intervención quirúrgica

Previo a cualquier intervención quirúrgica es necesario una valoración del riesgo anestésico que tiene un paciente que va a ser sometido a la intervención[9]. Esta valoración es realizada por el servicio de Anestesia. Para ello se realiza una serie de pruebas que incluye un cuestionario de preguntas, analítica, electrocardiograma y radiografía de tórax. Cuando este cuestionario se realiza, al paciente se le otorga una categoría según la clasificación ASA, la cual permite evaluar el grado de estado físico del paciente antes de someterse a una intervención quirúrgica. Esta escala tiene 6 categorías, siendo la I la de menor riesgo quirúrgico y la V la de mayor riesgo quirúrgico (la VI se aplica a pacientes con muerte cerebral).

Para llevar a cabo una intervención de implantación de prótesis de cadera, en el hospital "Lozano Blesa" es necesaria la participación de 3 cirujanos ortopédicos, una enfermera que instrumente la intervención, otra enfermera "circulante" que abastezca a la enfermera instrumentista del material que necesite. Un auxiliar de enfermería que prepare al paciente. Un anestésico y una enfermera de anestesia son los encargados de anestesiarse y mantener las constantes vitales del paciente mientras se realiza la intervención quirúrgica. El tipo de anestesia utilizada con mayor frecuencia en la cirugía de prótesis de cadera es la locoregional (intradural) o en algunos casos anestesia general.

Una vez terminada la intervención el paciente es trasladado a la sala de despertar donde se comprueba si el paciente está estable para ser trasladado a la planta. En la planta comienza el tratamiento post-operatorio. Este tratamiento incluye una serie de medidas y actos que se realizan en los días sucesivos. Entre estas actividades, se incluyen realización de una radiografía y analítica de control, administración dosis de antibiótico correspondiente, reinicio de tratamiento con heparina a dosis profilácticas, la retirada de redón y primera cura de herida, momento de sentar en el sillón y momento de iniciar la deambulación con muletas.

Algunos de los actos expuestos anteriormente, se llevan a cabo para evitar posibles complicaciones derivadas de la intervención quirúrgica. Dentro de las complicaciones más frecuentes, destacan, las propias relacionadas con la técnica quirúrgica, la infección protésica, la enfermedad tromboembólica y la anemia aguda post-operatoria. Se exponen a continuación algunas de las soluciones ante estas complicaciones:

- Para evitar la infección protésica, se administra una dosis de antibiótico (Cefazolina 2gr IV y en alérgicos a ésta, clindamicina 600 mg IV) en la inducción anestésica (momento previo a comenzar la intervención). Una vez finalizada la intervención quirúrgica, se administran otras dos dosis a las 8 y 16 horas post-intervención de Cefazolina 1 gr (o clindamicina 600 mg IV en pacientes alérgicos). Un factor que se ha visto que favorece la infección protésica es el elevado tiempo quirúrgico.
- Para evitar la enfermedad tromboembólica es imprescindible iniciar tratamiento con heparina a dosis profilácticas a las 12 horas post-intervención y mantener dicho tratamiento de forma diaria durante 30 días.
- Para evitar las complicaciones de una anemia aguda post-operatoria se realiza una analítica de control al día siguiente de la intervención, valorando la pérdida sanguínea. La pérdida sanguínea marca el parámetro de la hemoglobina. Si la hemoglobina es menor de 8 mg/dl estará indicado la transfusión de sangre.
- Para evitar las complicaciones propias de la técnica quirúrgica, se realiza una radiografía de la cadera al día siguiente de la intervención para ver que los componentes de la prótesis tienen una orientación adecuada y permiten iniciar la deambulación del paciente desde el momento que el paciente lo tolere.

Anexo 3.3 Definiciones y siglas médicas

A continuación, se definen algunos de los términos utilizados durante el desarrollo de este proyecto durante la recogida de datos para el sistema:

- **NHC (número de historia clínica):** Número que permite la identificación de un paciente. Cada paciente tiene asignado un número exclusivo que le identifica de tal forma que no puede existir un mismo número para dos pacientes.
- **ACO (anticoagulantes):** Fármacos que toman pacientes con enfermedades cardíacas. Esto implica que si estos pacientes se someten a cualquier cirugía, incluida una cirugía de cadera, tienen un riesgo mayor de padecer alguna complicación durante la misma. Además la toma de estos anticoagulantes favorece un mayor sangrado durante la intervención pudiendo incrementar el riesgo de transfusión.
- **ASA:** Se trata de una clasificación que permite estimar el riesgo que plantea la anestesia para el paciente si se somete a una intervención quirúrgica.
- **Momento de inducción:** Periodo de tiempo que transcurre desde que el paciente llega al bloque quirúrgico hasta realizar la anestesia. Durante este momento el paciente es valorado por una enfermera de anestesia y un anestesista y realizan una serie de medidas para realizar la anestesia al paciente. Entre estas medidas destacan la administración de fármacos previos a intervención (fármacos), canalización de vías periféricas o centrales, revisar pasos críticos de la anestesia.
- **Anestesia:** Es un acto médico que permite llevar a cabo una intervención quirúrgica eliminando el dolor. Existen diversos tipos de anestesia utilizados en cirugía de cadera
 - **Anestesia general:** consiste en la administración de fármacos por vía intravenosa o inhalada para provocar en el paciente un estado muy similar a un sueño profundo, sin alterar las constantes vitales del paciente.
 - **Anestesia raquídea:** consiste en la administración de fármacos aplicados mediante una inyección en la columna vertebral consiguiendo eliminar la sensibilidad en ambas extremidades inferiores.
- **Antibiótico:** Fármacos que se administran durante el tiempo peri-operatorio y que han demostrado disminuir el riesgo de infección protésica.
- **HB y HTO :** Hemoglobina y hematocrito. Son parámetros obtenidos de un análisis de sangre y permiten calcular la pérdida sanguínea. Para ello se realiza la diferencia entre los parámetros previos y tras la cirugía y ello indica la necesidad de transfusión.

Anexo 4: Servidor Web Django

Django es un framework web de desarrollo de código abierto que implementa el patrón arquitectural *modelo vista controlador*. Es utilizado para desarrollar aplicaciones web de forma sencilla a través del uso de múltiples conectores y facilidades. Se encarga de procesar las diferentes peticiones HTTP que se le realizan apoyándose en los datos que se le pasan a través de estas *URL*.

La estructura principal de procesamiento *Django* la podemos observar a continuación:

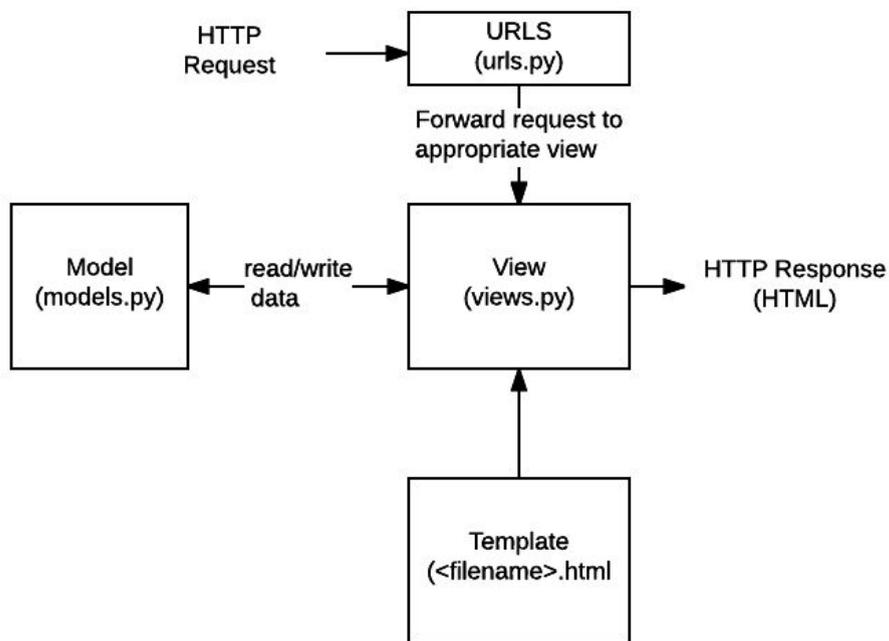


Figura 32 : Estructura de proyecto

Django(<https://developer.cdn.mozilla.net/es/docs/Learn/Server-side/Django/Introducci%C3%B3n>)

- **URLs:** Es el módulo encargado del procesamiento y enrutamiento de las diferentes peticiones HTTP hacia la vista correspondiente. Este además, permite patrones de dígitos y caracteres para unificar en uno las diferentes posibilidades de petición. Un ejemplo de enrutamiento se puede observar a continuación:

```
path(r'iniciarsesion/', views.iniciar_sesion,
      name='iniciar_sesion')
```

- **Views:** Es el módulo encargado de contener las vistas con la lógica de negocio. Dentro de este módulo se procesan las peticiones y se realizan los cálculos pertinentes a través del uso de modelos y se encargan de generar la respuesta HTML de vuelta. Se precisan de
- **Models:** Se definen los modelos utilizando la API de Django. Este permite conectarse a cualquier base de datos pero por defecto, viene integrado con una

base de datos *SQLite*. Esta API , nos proporciona una forma estándar de generar los modelos ya que ella misma se encarga posteriormente de traducir a las sentencias SQL. Un ejemplo de modelo se puede observar a continuación:

```
class RECURSO(models.Model):
    nombre_recurso = models.CharField(
        null=False,
        max_length=50 )

    tipo = models.CharField(
        null=False,
        max_length=50
    )
```

Para poder trabajar con estos modelos , se facilita una API que trabaja con los *queries* a través de la API *QuerySet* (<https://docs.djangoproject.com/en/3.0/ref/models/querysets/>).

- **Templates:** Son los ficheros con la plantilla HTML que se utilizan para renderizar la respuesta dinámicamente. Con el objetivo de aplicar dinamismo, se incorpora a esta estructura tiene su propio lenguaje de plantillas ,Django template language (DTL) que permite incluir y trabajar la lógica que se le pasa desde las vistas en estas plantillas. Este lenguaje permite entre otras cosas añadir filtros,variables,tags,bucles entre otras cosas. Un ejemplo de parte de una plantilla , que permite sacar errores pasados previamente en las vistas:

```
{% if errormessage %}
    <script>
    alert('{{ errormessage }}');
    </script>

{% endif %}
```

Anexo 5: Autómatas finitos probabilistas (PFA)

Se entiende como autómata finito probabilista A_k como una tupla tal que [4]:

$$A_k = \langle Q_{Ak}, \Sigma_{Ak}, \delta_{Ak}, I_{Ak}, F_{Ak}, P_{Ak} \rangle,$$

Teniendo en cuenta que:

- Q_{Ak} es el conjunto de estados finito no vacío q .
- Σ_{Ak} es el alfabeto no vacío de eventos e .
- $\delta_{Ak} \subseteq Q_{Ak} \times \Sigma_{Ak} \times Q_{Ak}$ es el conjunto de transiciones.
- $I_{Ak} : Q_{Ak} \rightarrow [0, 1]$ son las probabilidades de estado inicial.
- $P_{Ak} : \delta_{Ak} \rightarrow [0, 1]$ son las probabilidades de transición.
- $F_{Ak} : Q_{Ak} \rightarrow [0, 1]$ son las probabilidades de estado final.

Donde I_{Ak} , P_{Ak} y F_{Ak} son funciones tales que :

- $\sum_{q \in Q_{Ak}} I_{Ak}(q) = 1$
- $\forall q \in Q_{Ak}, F_{Ak}(q) + \sum_{e \in \Sigma_{Ak}, q' \in Q_{Ak}} P_{Ak}(q, e, q') = 1$

Por todo esto, los autómatas finitos probabilistas se pueden representar como grafos etiquetados. A continuación, se muestra un ejemplo de un autómata finito determinista : $Q_{Ak} = \{q_0, q_1, q_2, q_3\}$, con $I_{Ak}(q_0) = 1$ y $I_{Ak}(q_i) = 0$ para $i \in \{1, 2, 3\}$, y utilizando un alfabeto de cuatro símbolos: $\Sigma_{Ak} = \{a, b, c, d\}$ y cada transición con el símbolo del evento que ocurre y probabilidad de ocurrencia:

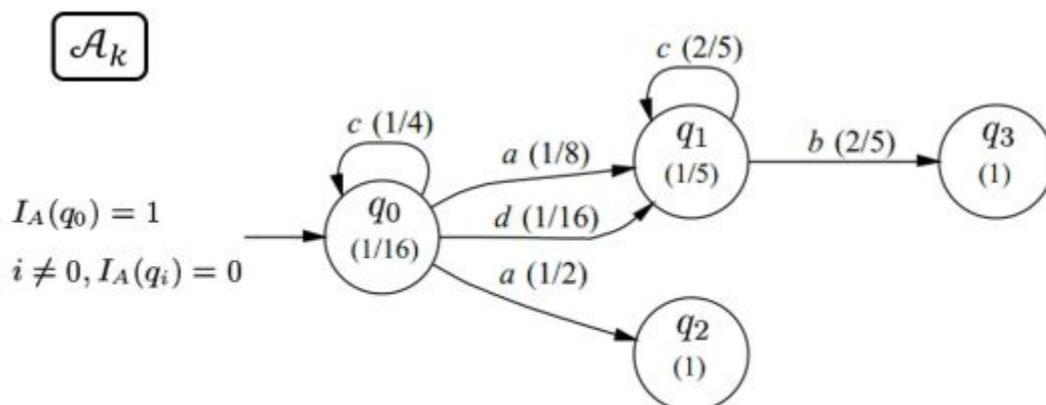


Figura 33 : Ejemplo de Autómata probabilista [4]

Para entender más detenidamente la probabilidad de la transición (q_0, a, q_1) que va desde q_0 a q_1 teniendo como probabilidad $P_{Ak}(q_0, a, q_1) = 1/8$ esta puede ser leída como la probabilidad del generador de eventos modelado por el autómata finito probabilista A_k de la figura superior de alcanzar q_1 desde el estado q_0 generando el evento a .

Dentro de los PFA, la operación clave de generación de cadenas o secuencias de estados a partir del conjunto de eventos Σ , es la concatenación. Esta se puede ver definida

fácilmente en la cadena de eventos abb : es la concatenación de una cadena ab con un evento. Así mismo, ab es la concatenación del evento b a la cadena a . La cadena vacía ϵ es la identidad de la concatenación teniendo la siguiente propiedad :

$$\epsilon u = u \epsilon \text{ para cualquier cadena } u$$

Anexo 6: Modelado de procesos y Análisis de datos

A continuación, se exponen dos tipos de modelado de procesos que se han investigado y pueden ser útiles en algunas situaciones para la modelización discreta de eventos:

Anexo 6.1 YAWL

Es un sistema de modelado y un sistema de trabajo de código abierto cuyo objetivo es como se describe en el título la modelación de procesos. Este sistema cubre una amplia gama de patrones de modelado y diseño desde patrones de control, datos, recursos y excepciones. El objetivo de YAWL es ofrecer un amplio abanico de opciones para poder así actuar en todos los ámbitos manteniendo un lenguaje simple. Aunque en un inicio se planteó como un lenguaje de modelado de procesos y como una forma de construir sistemas, lo cierto es que se ha ido entrelazando. Tanto es así que la comunidad académica muchas veces ha acabado por utilizar YAWL para designar ambas cosas.

YAWL utiliza los operadores lógicos conocidos, y trabajar mediante tareas (*tasks*) que se entrelazan con dichos operadores. A continuación se adjuntan los operadores y un ejemplo de modelado en YAWL:

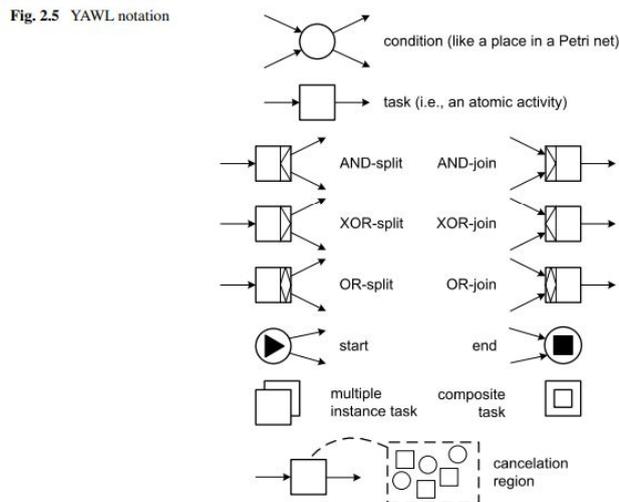


Figura 34: Simbología YAWL[1]

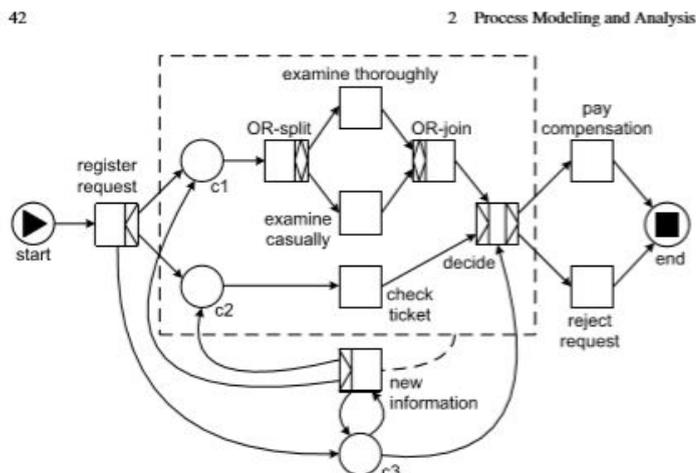


Figura 35: Ejemplo de sistema YAWL[1]

Este modelo tiene una ventaja y es que permite pasar de redes de Petri a YAWL de forma directa lo que permite ofrecer una visión más fácil del proceso.

Anexo 6.2 Business Process Modeling Notation

BPMN(*Business Process Modeling Notation*) es una notación gráfica estandarizada que permite el modelado de procesos de negocio. Esta utiliza la idea de flujo de trabajo (*workflow*) para modelar un proceso de negocio. El objetivo de esta notación es modelar el paso de mensajes que ocurre entre los diferentes elementos dentro de la misma. Tiene la ventaja frente a YAWL que al ser un estándar está muy desarrollado y cuenta con herramientas de todo tipo para análisis , desarrollo y validación. Un ejemplo de esta es camunda(<https://camunda.com/>).

Muy similar a los anteriores, se pueden clasificar los objetos del *workflow* en tres tipos :

- **Actividades:** Determina las diferentes acciones que se van realizando durante el proceso. Se simboliza con un cuadrado
- **Eventos:** Determina las ocurrencias que se dan dentro del flujo , como que un usuario empieza o interrumpe una acción u ocurre un paso de mensajes.
- **Gateways:** Define los puntos de desvío que determinarán la forma en que va a seguir el proceso , simbolizados por diamantes y suelen corresponder a los operadores lógicos.
- **Recursos:** Como bases de datos, ficheros ,servidores que se representa con un dibujo del objeto.

A continuación se muestran los diferentes operadores con los que cuenta y un ejemplo de diagrama:

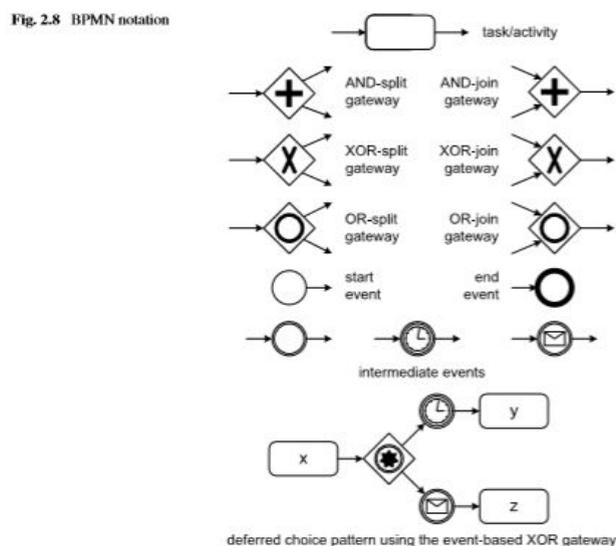


Figura 36: Notación BPMN [1]

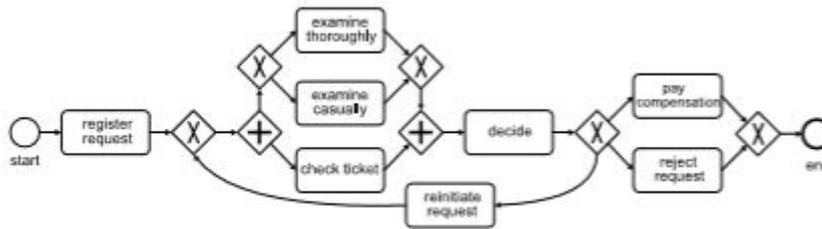


Fig. 2.7 Process model using the BPMN notation

Figura 37: Ejemplo de modelado BPMN [1]

Este sistema de modelado tiene la ventaja de aparte de estar estandarizado, cuenta con una amplia gama de elementos para la descripción de paso de mensajes y de eventos aunque a veces , puede dar lugar a diagramas muy complejos.

Anexo 6.3 Event-Driven Process Chains

Es un tipo de diagrama de flujo diseñado para gestión de recursos en el ámbito de la empresa y los procesos asociados a ella. Nace junto a *SAP 3 (Sistema de planificación de recursos empresariales)* y englobado dentro del marco de trabajo AIS (*Architecture of Integrated Information Systems*).

En el marco AIS, el modelo de negocio a modelar se puede observar desde cinco puntos de vista:

- **Function view:** Las actividades y relaciones que existen entre ellas se describen en este ámbito a través de un modelado específico.
- **Organization view:** Se otorga una visión general del negocio y de los diferentes aspectos que lo componen como recursos humanos, dispositivos, hardware etc.
- **Data view:** Todos los eventos que generen datos y trabajen con ellos.
- **Product/Service view:** Se muestra un resumen del modelo de negocio con el que se trabaja.
- **Process view:** La vista del proceso dentro del modelo de negocio , que en este caso puede ser *EDPC*⁶.

Los elementos de los que se compone un *EDPC* son :

- ➔ **Eventos:** Son elementos que describen los requisitos de entrada a un proceso o la salida del mismo.
- ➔ **Funciones:** Elementos activos de los *EDPC* que describen las transformaciones de un proceso desde un estado inicial a un estado final.
- ➔ **Dueño de un proceso:** Forma de modelado explícita de modelar quien debe hacer qué tarea.
- ➔ **Unidad organizativa:** Determina que departamento dentro de una empresa es el responsable de esa tarea.

⁶ Event-Driven Process Chains

- **Información, material, u objeto de recursos (Términos Técnicos):** Determina los recursos que precisan para llevar a cabo ese proceso en concreto.
- **Relaciones lógicas:** Como en los diagramas anteriores entre diferentes eventos puede surgir la necesidad de relacionarlo mediante los operadores lógicos, *EDPC* cuenta con elementos para ello.

A continuación, se muestra un esquema de modelado de un *EDPC* y un ejemplo del mismo:

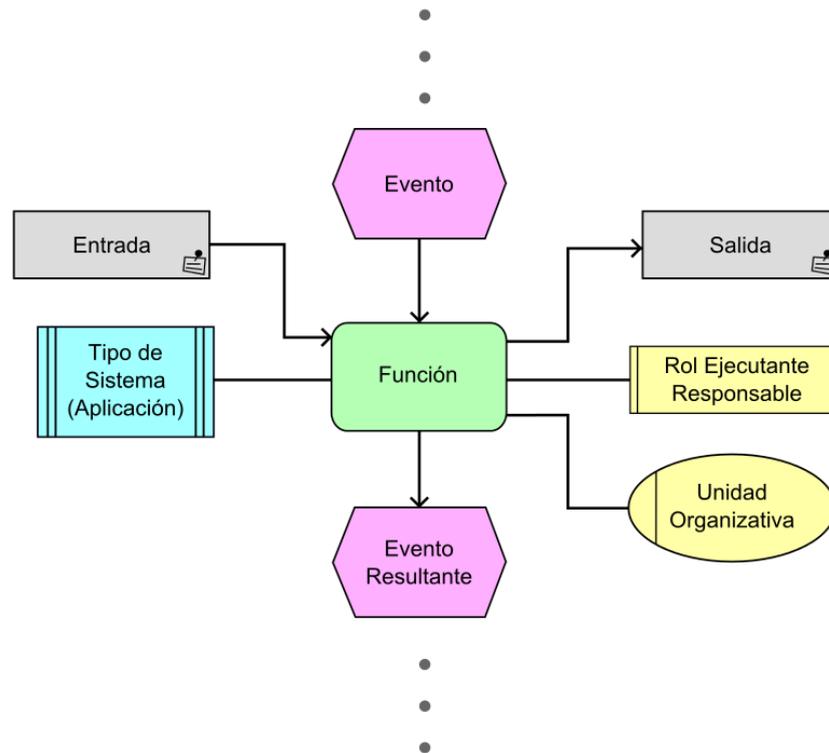


Figura 38: Ejemplo diagrama de EDPC

([https://es.wikipedia.org/wiki/Event-driven_process_chain#Elementos_de_una_Cadena_de_Procesos_Impulsada_por_Eventos_\(CPE\)](https://es.wikipedia.org/wiki/Event-driven_process_chain#Elementos_de_una_Cadena_de_Procesos_Impulsada_por_Eventos_(CPE)))

Anexo 6.3 Estado del arte en la minería de datos

La minería de datos engloba toda una disciplina de análisis de datos siendo la minería de procesos un subconjunto de esta [1]. Por ello, se van a explicar las diferentes técnicas que se han barajado para la solución de este dominio de problema concreto en el ámbito de la minería de datos:

- **Aprendizaje supervisado:** Se denomina aprendizaje supervisado al cual se le proporcionan datos etiquetados. Estos datos contienen los *inputs* del sistema y el resultado que finalmente ha sido. En el siguiente ejemplo de datos sobre tumores se puede ver el etiquetado de cada uno:

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000025	5	1	1	1	2	1	3	1	1	benign
1002945	5	4	4	5	7	10	3	2	1	benign
1015425	3	1	1	1	2	2	3	1	1	malignant
1016277	6	8	8	1	3	4	3	7	1	benign
1017023	4	1	1	3	2	1	3	1	1	benign
1017122	8	10	10	8	7	10		7	1	malignant
1018099	1	1	1	1	2	10	3	1	1	benign
1018561	2	1	2	H	2	1	3	1	1	benign
1033078	2	1	1	1	2	1	1	1	5	benign
1033078	4	2	1	1	2	1	2	1	1	benign

Figura 39: Ejemplo de aprendizaje supervisado con datos etiquetados (<https://stackoverflow.com/questions/19170603/what-is-the-difference-between-labeled-and-unlabeled-data>)

Dentro de este apartado se pueden distinguir dos tipos de técnicas:

❖ **Clasificación:** En este ámbito entrarían los **árboles de decisión** y las regresiones.

➤ **Árboles de decisión:** En base a los datos etiquetados se genera un diagrama en forma de *árbol no binario* (aunque puede haber veces que sea binario) con todas sus posibilidades. En el siguiente ejemplo se puede observar una modelización de los mismos para saber que tipo de coche utilizan en base a la edad y a si está casado o no:

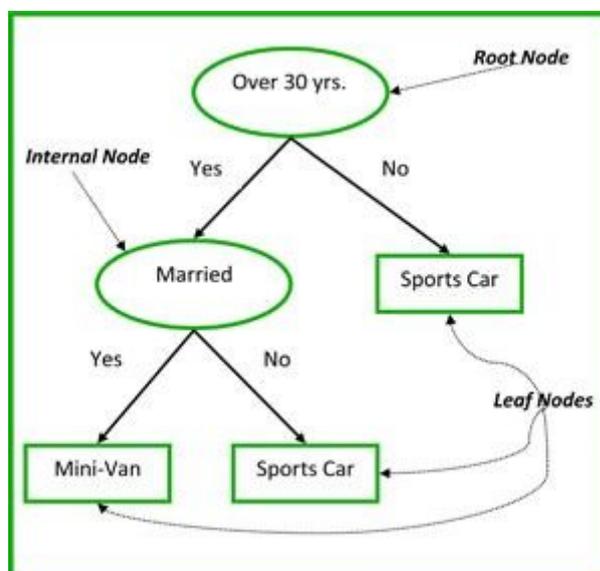


Figura 40: Ejemplo de árbol de decisión (<https://conectasoftware.com/analytics/arboles-de-decisiones-en-la-mineria-de-datos>)

➤ **Regresiones:** Esta técnica es muy utilizada en el mundo del aprendizaje automático y el objetivo de esta es encontrar una función matemática que encaje con los datos que nos proporcionan. En esta

técnica se intenta buscar qué variables(*predictores*) influyen más en el modelo. Se intenta ajustar con funciones de poca complejidad como las lineales. Un ejemplo de regresión se puede ver en la página de *scikit* (librería de *Python* para minería de datos):

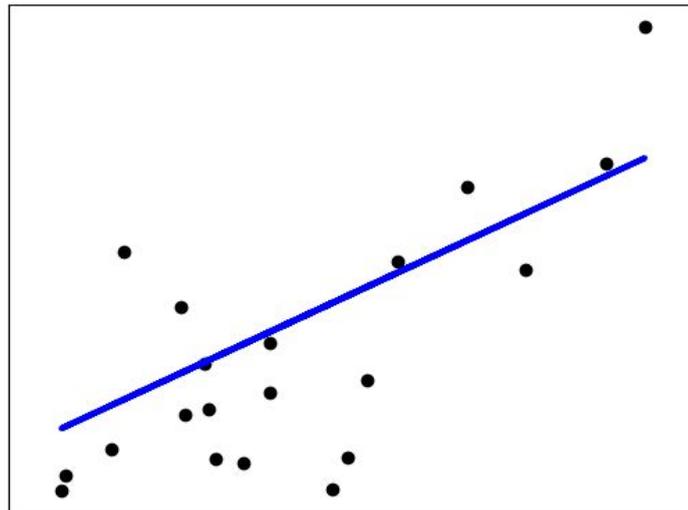


Figura 41: Ejemplo de regresión lineal

(https://scikit-learn.org/stable/auto_examples/linear_model/plot_ols.html)

- **Aprendizaje no supervisado:** Continuando con lo anterior, se denomina aprendizaje no supervisado al conjunto de técnicas que trabajan con datos no etiquetados. Por ejemplo, en la fotografía de a continuación se pueden observar datos de clientes, sin especificar una clase o conjunto al que pertenecen:

Customer Id	Age	Edu	Years Employed	Income	Card Debt	Other Debt	Address	DebtIncomeRatio
1	41	2	6	19	0.124	1.073	NBA001	6.3
2	47	1	26	100	4.582	8.218	NBA021	12.8
3	33	2	10	57	6.111	5.802	NBA013	20.9
4	29	2	4	19	0.681	0.516	NBA009	6.3
5	47	1	31	253	9.308	8.908	NBA008	7.2
6	40	1	23	81	0.998	7.831	NBA016	10.9
7	38	2	4	56	0.442	0.454	NBA013	1.6
8	42	3	0	64	0.279	3.945	NBA009	6.6
9	26	1	5	18	0.575	2.215	NBA006	15.5
10	47	3	23	115	0.653	3.947	NBA011	4
11	44	3	8	88	0.285	5.083	NBA010	6.1
12	34	2	9	40	0.374	0.266	NBA003	1.6

unlabeled

Figura 42: Ejemplo de aprendizaje con datos no etiquetados

(<https://stackoverflow.com/questions/19170603/what-is-the-difference-between-labeled-and-unlabeled-data>)

Dentro de este ámbito se encuentran dos grandes técnicas:

- ❖ **Clustering:** Agrupa el conjunto de algoritmos y técnicas cuyo objetivo es agrupar los datos. Cuando se tienen los datos dispersos en una gráfica se intentan descubrir las relaciones que existen entre estos grupos como se puede observar a continuación donde mediante el algoritmo de *k-medias* se ha dividido en 3 grupos:

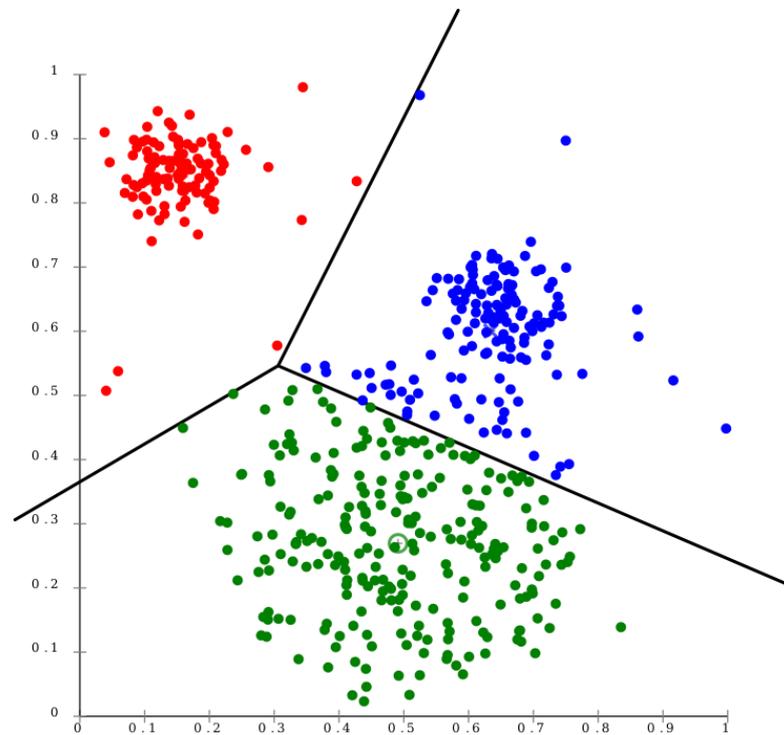


Figura 43: Ejemplo de aplicación de clustering
(https://en.wikipedia.org/wiki/Cluster_analysis)

- ❖ **Reglas de asociación:** Es una técnica que busca el encontrar eventos que ocurren tras sucederse otro. A este conjunto de eventos que realiza un usuarios se le conoce como transacciones. Basándonos en el ámbito médico el objetivo es encontrar relaciones del tipo que si alguien se le ha realizado una analítica y una radiografía se le hará una biopsia de piel según el conjunto de datos. Esto se consigue modelando todas las transacciones en una tabla y aplicando los diferentes algoritmos sobre esta. A continuación se expone un ejemplo en un supermercado⁷:

⁷ https://en.wikipedia.org/wiki/Association_rule_learning

Example database with 5 transactions and 5 items

transaction ID	milk	bread	butter	beer	diapers
1	1	1	0	0	0
2	0	0	1	0	0
3	0	0	0	1	1
4	1	1	1	0	0
5	0	1	0	0	0

Figura 44: Ejemplo de datos para reglas de asociación
(https://en.wikipedia.org/wiki/Association_rule_learning)

El objetivo en este caso es conseguir inferir que productos se van a comprar a la vez. Un ejemplo de regla de asociación $\{butter, bread\} \Rightarrow \{milk\}$

Anexo 7: Búsqueda del camino mínimo en grafo: Algoritmo de Dijkstra

Un grafo se define como un conjunto de vértices o nodos interconectados entre sí mediante aristas que permiten representar relaciones binarias entre elementos de un conjunto. Una relación binaria R es el subconjunto de los elementos del producto cartesiano $A_1 \times A_2$ que cumplen una determinada condición⁸:

$$R = \{(a_1, a_2) : (a_1, a_2) \in A_1 \times A_2 \wedge R(a_1, a_2) = Verdadero\}$$

Figura 45: Condicion de relaciones binarias

[https://es.wikipedia.org/wiki/Relaci%C3%B3n_binaria]

El algoritmo de Dijkstra es un algoritmo para el cálculo del camino mínimo en un grafo en base a unos costes en las aristas entre dos vértices. El algoritmo de *Dijkstra* es de complejidad $O(n^2)$ donde n es el número de vértices que sirve para encontrar el camino de coste mínimo desde un nodo origen a todos los demás nodos del grafo siendo este uno de los más rápidos.

El fundamento en el que se basa este algoritmo es un principio de optimalidad: si el camino más corto entre los vértices u y v pasa por el vértice w , entonces la parte del camino que va de w a v debe ser el camino más corto entre todos los caminos que van de w a v como se puede ver a continuación [10]:

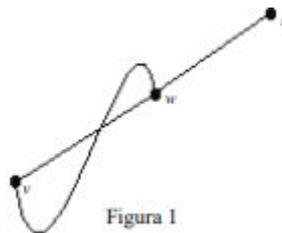


Figura 1

Figura 46: Principio de optimalidad[10]

A través de este principio, se pueden inferir el resto de caminos de coste mínimo desde un vértice inicial hasta los diferentes vértices del grafo utilizando los caminos conseguidos como parte de los nuevos caminos.

Se propone el siguiente pseudocódigo para facilitar su entendimiento:

- **G** = Grafo (V,E) donde V es el conjunto de vértices y E el de arcos.
- **S** : Conjunto de vértices cuyos caminos más cortos al origen han sido ya determinados.

⁸ <https://es.wikipedia.org/wiki/Grafo>

- **V-S** : Resto de vértices del grafo.
- **d**: ARRAY que asocia los costes de cada transición (El coste puede estar asociado a la estimación hasta el nodo final. Pero, en el caso de este dominio será la probabilidad o el valor propuesto en el apartado 3.1 según las necesidades).
- **pr**: ARRAY de vértices predecesores para cada vértice

1. Inicialización de d y pr
2. $S = \emptyset$ (conjunto vacío)
3. While $\{V-S\} \neq \emptyset$
 - a. Ordenar los vértices en V-S y analizar de acuerdo al menos coste de transición d
 - b. Añadir u, el vértice con el menor coste en V-S, a S
 - c. Recalcular el coste de todos V-S restantes si fuese necesario

Anexo 8: Gestión del proyecto

A continuación, se muestra la distribución de esfuerzos en la siguiente tabla:

Tarea	Tiempo (Horas)
Análisis de los modelos actuales de procesos	20
Reuniones con el equipo médico y la universidad	49
Diseño de pantallas HTML estáticas	16
Diseño de base de datos	20
Implementación de servidor web	40
Obtención y procesado de datos	15
Módulo de autómatas finitos deterministas	25
Estudio de CPN Tools y desarrollo de módulo	31
Panel de mando	12
Estudio de plataformas <i>Cloud</i> y despliegue	18
Análisis de resultados	7
Memoria	50
Total	303