



UNIVERSIDAD  
DE GRANADA

# Desarrollo de un Sistema Basado en Conocimiento

10 de Junio 2022

*Ingeniería del Conocimiento*

*Gallego Menor, Francisco Javier*

*javigallego@correo.ugr.es*

*74745747W*

## Índice

<b>1. Resumen del funcionamiento</b>	<b>3</b>
<b>2. Descripción del proceso seguido</b>	<b>3</b>
2.1. Procedimiento para el desarrollo de la base del conocimiento . . . . .	3
2.2. Procedimiento de validación y verificación del sistema . . . . .	4
2.2.1. Procedimiento de validación del sistema . . . . .	4
2.2.2. Procedimiento de verificación del sistema . . . . .	5
<b>3. Descripción del sistema</b>	<b>6</b>
3.1. Variables de entrada del problema. Como se representan . . . . .	6
3.1.1. Apartado A) . . . . .	6
3.1.2. Apartado B) . . . . .	6
3.2. Variables de salida del problema. Como se representan . . . . .	6
3.2.1. Apartado A) . . . . .	6
3.2.2. Apartado B) . . . . .	6
3.3. Conocimiento global del sistema (hechos y relaciones que se cargan inicialmente)	7
3.3.1. Apartado A) . . . . .	7
3.3.2. Apartado B) . . . . .	7
3.4. Especificación de los módulos se han desarrollado . . . . .	9
3.4.1. Estructura en módulos . . . . .	9
3.4.2. Descripción de cada módulo. Hechos que utiliza, hechos que deduce, y reglas. . . . .	9
<b>4. Manual de usuario</b>	<b>11</b>

## 1 Resumen del funcionamiento

En esta práctica se ha implementado un sistema basado en conocimiento que tiene el fin de recomendar a un estudiante de Ing. Informática. Este incluye dos tipos de asesoramiento:

- En la primera parte, se aconseja qué rama debería de escoger el alumno. Esta parte no maneja ningún tipo de incertidumbre.
- De entre dos asignaturas que se escogen, cual es la más apropiada para el alumno y por qué. En esta parte, al contrario que en la anterior, si se maneja incertidumbre. Para tratarla, hemos hecho uso de factores de certeza.

Para llevar a cabo los dos asesoramientos, se le van a realizar una serie de preguntas al estudiante. Se puede solicitar parar las preguntas en el momento que el alumno desee.

En base a las respuestas del alumno, el sistema va a proporcionarle una respuesta intentando aconsejarle. Así mismo, incluirá una justificación del proceso de razonamiento llevado a cabo.

En cuanto a lo general, el sistema ha sido dividido en dos subsistemas, cada uno correspondiente a uno de los apartados. A su vez, dentro de cada subsistema hemos dividido en módulos. Todo esto con el fin de que sea fácilmente modificable y entendible por alguien ajeno a la implementación.

## 2 Descripción del proceso seguido

En esta sección voy a detallar como ha sido el proceso que se ha llevado a cabo para la implementación del sistema.

### 2.1 Procedimiento para el desarrollo de la base del conocimiento

Por una parte, para el desarrollo de la base del conocimiento de esta práctica me he apoyado en mis propios conocimientos, como alumno del grado que soy. En mi caso, he buscado cierta información a cerca de las asignaturas de otras ramas optativas. Esto debido a que no disponía de la suficiente información como para tener un conocimiento completo sobre el tema.

Por otro lado, también hemos hecho uso de diversos ejercicios que han sido realizados para la parte de teoría.

- Primero, se llevo a cabo un estudio del problema. Esto es, realizamos un ejercicio entre diversos compañeros de clase en el que analizábamos como uno de nosotros (como

experto) aconsejaría al usuario que utilice el sistema. Para ello, se replicó un modelo de entrevista entre un experto y varios ingenieros de conocimiento.

- En segundo lugar, se realizó un ejercicio de Rejilla De Repertorio de las Asignaturas, el cual me ayudó a observar cuáles eran los atributos importantes a la hora de recomendar una asignatura u otra.
- Por último, realizamos un ejercicio práctico de adquisición del conocimiento en el que mediante una estructura de árbol hicimos una posible calificación dependiendo de los atributos que pudiera presentar un usuario.

## 2.2 Procedimiento de validación y verificación del sistema

Para el procedimiento de validación y verificación del sistema, tuvimos que realizar un ejercicio para la teoría. Este nos pedía determinar la planificación del proceso de verificación y validación que se va a aplicar durante el desarrollo del sistema de la unidad de prácticas 6. Los pasos de verificación y validación que hemos considerado son los siguientes:

- Verificar si el sistema es completo, correcto y consistente
- Evaluar si el sistema cumple especificaciones del modelo de diseño.
- Diseñar un plan de validación aplicando metodologías apropiadas.
- Valorar en función de criterios de validación. Entre otros los requisitos funcionales definidos en la fase de identificación del problema.

### 2.2.1 Procedimiento de validación del sistema

Como bien sabemos de la teoría, la validación consiste en verificar si el sistema es correcto. Para ello, necesitaremos de la participación del experto.

- La interfaz es comprensible para el usuario. Dado que nuestra interfaz no presenta una gran complejidad no debe haber problema alguno.
- La comunicación del sistema con otros sistemas (transferencias) es adecuada. Gracias a la implementación y el diseño, es posible conseguirlo. Además, es fácil y posible añadir o modificar el conocimiento.
- La explicación del razonamiento del sistema es suficiente. El sistema es capaz de justificar las respuestas consistentes.

- Cumple los requisitos de ejecución en tiempo real pedidos. En nuestro caso, no se nos han proporcionado dichos requisitos de tiempo. Sin embargo, es lógico que el SBC no debería de tardar mucho en dar esta respuesta. (En nuestro caso, la da al instante).
- El sistema cumple las especificaciones de seguridad. Por ejemplo, en ningún momento un usuario tiene a su disposición datos personales de otro usuario que haya hecho uso previamente del sistema.
- Satisfacción y utilidad de los resultados finales e intermedios comparados con: resultados conocidos, prestaciones de un experto o de un modelo algorítmico. Para ello, se han realizado varios casos de prueba. Se ha contactado con el/los experto/s al que realizamos una entrevista en el pasado para verificar que funciona correctamente, que el conocimiento es válido y suficiente. Además, distintos usuarios también han podido comprobar y verificar que las respuestas son concisas y correctas. Se obtiene una adecuación al problema y se evitan los errores por comisión (se responde incorrectamente) o por omisión (no se responde)

### 2.2.2 Procedimiento de verificación del sistema

Procedimiento de verificación del sistema El ingeniero del conocimiento verificará que el sistema se ha construido correctamente. Los criterios a verificar en un SBC que se han tenido en cuenta son los siguientes.

- Consistencia: no se llega a conclusiones incoherentes. Es decir, no puede pasar que aconseje la rama de Ingeniería de Computadores, si el estudiante ha indicado que no le gusta el hardware.
- Corrección: hay corrección en la sintaxis.
- Completitud: no hay lagunas en capacidad deductiva (casos donde el sistema no respondería y que se puedan dar en la práctica). Debemos tener especial cuidado entonces, en situaciones en las que el estudiante responda que no quiere seguir haciendo preguntas, y asegurarnos que aún así, siempre recibe una respuesta. De tal manera que continúe la interacción.

### 3 Descripción del sistema

#### 3.1 Variables de entrada del problema. Como se representan

##### 3.1.1 Apartado A)

He usado los siguientes hechos para recoger las respuestas del alumno.

- (gusta [matematicas | programacion | hardware] SI | NO | NS): para representar la respuesta del usuario a las preguntas sobre si le gusta o no algo.
- (r[Nota | Trabajo | Trabajador | TipoClases] ?): para representar la respuesta del usuario a las preguntas que no tienen como respuesta un simple si o no.

##### 3.1.2 Apartado B)

He usado los siguientes hechos para recoger las respuestas del alumno.

- (asignatura\_i <nombre>): para representar la respuesta del usuario a la elección de las asignaturas, con  $i \in \{1, 2\}$ .
- (Evidencia [matematicas | programacion | hardware | teoria | bases\_datos] ?): para representar la respuesta del usuario a las preguntas sobre sus gustos por las matematicas.

#### 3.2 Variables de salida del problema. Como se representan

##### 3.2.1 Apartado A)

He usado los siguientes hechos para recoger la respuesta del sistema, con su correspondiente justificación.

- (Consejo [CSI | IC | IS | SI | TI] ?motivo FJGallego).

##### 3.2.2 Apartado B)

Para la salida en este apartado, hemos hecho uso de los hechos proporcionados en el ejercicio de los factores de certeza. Dicha sintaxis es la que sigue:

- (Motivo FactorCerteza ?asignatura [SI | NO] ?factor ?explicación): así mismo, este hecho también es usado para sacar por pantalla las evidencias proporcionadas por el usuario.

### 3.3 Conocimiento global del sistema (hechos y relaciones que se cargan inicialmente)

#### 3.3.1 Apartado A)

Hechos y relaciones que se cargan inicialmente. Estos primeros, hacen referencia a las distintas ramas optativas que ofrece el grado de Ing. Informática.

---

```
(deffacts Ramas
  (Rama Computacion y Sistemas Inteligentes)
  (Rama Ingenieria del Software)
  (Rama Ingenieria de Computadores)
  (Rama Sistemas de Informacion)
  (Rama Tecnologias de la Informacion)
)
```

---

Los siguientes, hacen referencia a las distintas preguntas que se le van a realizar al alumno.

---

```
(deffacts Preguntas
  (Modulo ModPreg)
  (pregunta sobre matematicas)
  (pregunta sobre trabajo)
  (pregunta sobre nota)
  (pregunta sobre hardware)
  (pregunta sobre trabajador)
  (pregunta sobre programacion)
  (pregunta sobre tipoClases)
)
```

---

#### 3.3.2 Apartado B)

Los hechos hacen referencia a cada una de las asignaturas del grado en Ing. Informática. Incluyo solo una primera parte en la siguiente imagen:

```

; Asignaturas del Grado de Ing. Informática
(deffacts Asignaturas
  ; Asignaturas Basicas -> Primer Curso todas
  (Asignatura Basica FP "Fundamentos de programacion")
  (Asignatura Basica TOC "Tecnologia y organizacion de computadores")
  (Asignatura Basica CAL "Calculo")
  (Asignatura Basica ALEM "Algebra lineal y estructuras matematicas")
  (Asignatura Basica MP "Metodologia de la programacion")
  (Asignatura Basica FFT "Fundamentos fisicos y tecnologicos")
  (Asignatura Basica EST "Estadistica")
  (Asignatura Basica IES "Ingenieria, empresa y sociedad")
  (Asignatura Basica FS "Fundamentos del software")
  (Asignatura Basica LMD "Logica y metodos discretos")

  ; Asignaturas Obligatorias -> Segundo y tercer curso
  (Asignatura Obligatoria DDSI "Diseño y desarrollo de sistemas de informacion")
  (Asignatura Obligatoria PD00 "Programacion y diseño orientado a objetos")
  (Asignatura Obligatoria ISE "Ingenieria de servidores")
  (Asignatura Obligatoria SO "Sistemas operativos")
  (Asignatura Obligatoria SCD "Sistemas concurrentes y distribuidos")
  (Asignatura Obligatoria AC "Arquitectura de computadores")
  (Asignatura Obligatoria ED "Estructuras de datos")
  (Asignatura Obligatoria FBD "Fundamentos de bases de datos")
  (Asignatura Obligatoria FIS "Fundamentos de ingenieria del software")
  (Asignatura Obligatoria EC "Estructura de Computadores")
  (Asignatura Obligatoria FR "Fundamentos de redes")
  (Asignatura Obligatoria IG "Informatica grafica")
  (Asignatura Obligatoria IA "Inteligencia artificial")
  (Asignatura Obligatoria MC "Modelos de computacion")
  (Asignatura Obligatoria ALG "Algoritmica")

  ; Asignaturas Optativas -> Tercer y cuarto Curso

  ; Rama Optativa de Ingenieria de Computadores
  (Asignatura Optativa AS "Arquitectura de sistemas")
  (Asignatura Optativa ACAP "Arquitectura y computacion de altas prestaciones")
  (Asignatura Optativa DHD "Desarrollo de hardware digital")
  (Asignatura Optativa DSE "Diseño de sistemas electronicos")
  (Asignatura Optativa SCM "Sistemas con microprocesadores")
  (Asignatura Optativa CPD "Centros de procesamiento de datos")
  (Asignatura Optativa SE "Sistemas empotrados")
  (Asignatura Optativa TR "Tecnologias de red")

  ; Rama Optativa de Sistemas de la Informacion
  (Asignatura Optativa ABD "Administracion de bases de datos")
  (Asignatura Optativa ISI "Ingenieria de sistemas de informacion")
  (Asignatura Optativa PW "Programacion web")

```

Figura 1:



### 3.4 Especificación de los módulos se han desarrollado

Los dos asesoramientos se realizan de forma secuencial, es decir, cuando termina uno comienza el otro. Entonces, los módulos se ejecutan de forma similar.

#### 3.4.1 Estructura en módulos

Para el primer apartado la estructura en módulos es la que sigue:

- ModuloPreguntas
- ModuloCalcular
- ModuloConsejo

En cuanto al segundo apartado de la práctica, el de las asignaturas, la estructura en módulos es como sigue:

- ModuloPregAsig
- ModuloCalcAsig
- ModuloConsAsig

#### 3.4.2 Descripción de cada módulo. Hechos que utiliza, hechos que deduce, y reglas.

En cuanto a la primera parte:

- ModuloPreguntas: este módulo es el encargado de hacer las preguntas al usuario, y de extraer la información necesaria para llevar a cabo el cálculo de la rama apropiada. En este módulo es posible detener el cuestionario de preguntas en cualquier momento, respondiendo PARAR. Además, se lleva a cabo una comprobación de las respuestas del usuario, invalidándolas y pidiendo una nueva en caso de no ser correctas. Este módulo finaliza y da paso al siguiente cuando o bien se han realizado todas las preguntas, o bien se ha pedido parar las preguntas. En este modulo no se deduce nada. Las reglas que usa son:
  - `preg_<caracteristica>`: son las reglas encargadas de preguntar al resultado a cerca de una característica concreta. Así mismo, recogen la respuesta del usuario.

- `respuesta_<caracteristica>`: son las reglas encargadas de comprobar que la respuesta introducida por el usuario para la pregunta sobre la característica `<caracteristica>` es válida.
  - `parar_pregunta_<caracteristica>`: encargada de parar las preguntas en el caso de que la respuesta a la pregunta de `<caracteristica>` haya sido PARAR.
  - `ajustar_Nota_<tipo>`: se encarga de ajustar el tipo de nota media de un alumno, en el supuesto caso de que haya respondido con un número a la pregunta a cerca de las notas.
  - `no_queda_pregs`: mediante esta regla comprobamos si no quedan mas preguntas por responder.
  - `preg_parar`: mediante esta regla comprobamos si el usuario ha decidido parar las preguntas.
- **ModuloCalcular**: es el encargado de calcular (deducir) la rama optativa mas apropiada para el usuario, segun las respuestas que nos haya proporcionado a nuestras preguntas. Las reglas usadas en este módulo son las que siguen:
- `<Rama>_[1,2,3 ...]`: estas reglas nos sirven para deducir los supuestos en los que la rama `<Rama>` es la adecuada para el usuario. El nº de reglas para una misma rama dependerá de las mas o menos posibilidades que tenga dicha rama de ser escogida.
  - `solo_[1,2,3, 4]_resp`: mediante estas reglas hacemos el calculo de la rama apropiada cuando se han dejado preguntas sin responder.
  - `noHayMasCalculos`: mediante esta regla terminamos si ya hemos decidido que rama aconsejar.
- **ModuloConsejo**: es el encargado de mostrar por pantalla el consejo final, junto con su correspondiente explicación. En este módulo no se deduce nada nuevo. La regla que usa es la siguiente:
- `consejo_final`: mediante esta regla mostramos por pantalla el consejo final, con su justificación.

En cuanto al segundo apartado:

- **ModuloPregAsig**: este módulo es el encargado de hacer las preguntas al usuario, y de extraer la información necesaria para llevar a cabo el cálculo de la rama apropiada. En este módulo es posible detener el cuestionario de preguntas en cualquier momento, respondiendo PARAR. Además, se lleva a cabo una comprobación de las respuestas del

usuario, invalidándolas y pidiendo una nueva en caso de no ser correctas. Este módulo finaliza y da paso al siguiente cuando o bien se han realizado todas las preguntas, o bien se ha pedido parar las preguntas. En este modulo no se deduce nada. Las reglas que usamos en este modulo son:

- preguntar\_asignaturas: mediante esta regla le pedimos al usuario que introduzca dos asignaturas de las disponibles en el grado.
  - comprobar\_entrada\_asignaturas: comprobamos que la respuesta es correcta.
  - asignar\_asignaturas\_correctas: si se han introducido bien las asignamos.
  - certezas\_evidencias: convertimos cada evidencia en una afirmación sobre su factor de certeza.
  - reglas relacionadas con los factores de certeza: cogidas del ejemplo proporcionado por el profesor.
  - preguntar\_evidencias\_<caracteristica>: mediante esta regla preguntamos a cerca de los gustos del alumno, relacionados con la característica <caracteristica>.
  - no\_quedan\_pregs: mediante esta regla comprobaremos si no quedan mas preguntas por responder.
  - preg\_parar: mediante esta regla comprobamos si el usuario ha decidido parar las preguntas.
- ModuloCalcAsig: se calcula la asignatura apropiada. En este módulo, para el **manejo de incertidumbre pedido** se han utilizado **factores de certeza**. De esta forma, se le asigna un factor a cada asignatura. Nos quedamos con aquella asignatura que mayor certeza proporciona. Se han utilizado las siguientes reglas:
    - Ri: donde para cada  $i \in \{1...22\}$  se recomienda una asignatura por x motivo.
    - hipotesis\_mayor\_certeza: tras razonar nos quedamos con la hipotesis que mayor certeza tenga.
    - fin\_calculos: no hay mas calculos.
  - ModuloConsAsig: mostramos por pantalla el consejo final del experto, con su debida explicación.

## 4 Manual de usuario

Cuando se ejecuta el sistema se empieza ejecutando la primera parte y después la segunda. Si no interesa ejecutar la primera parte, se puede parar con las preguntas respondiendo a alguna de ellas con PARAR

Aparte de lo ya mencionado la ejecución es muy sencilla, solo se debe responder a las preguntas realizadas por el sistema en el formato especificado.