

# Sistema Experto de recomendación de lenguaje de programación

Juan Carlos Fernández Durán and Victor Gualdrás de la Cruz.

**Expertos:** Ismael Caballero Muñoz-Reja. Pascual Julián Iranzo. David Villa Alises.

**Profesor y director del proyecto:** Manuel Emilio Prieto Méndez.

Universidad Castilla la Mancha, Escuela Superior de Informática  
Ciudad Real, España

**Abstract** In this document is exposed the goal that aims to accomplish the programming language recommender system. It would be analyzed the requirement of a system like that, what it wants to cover and the utilities it will have. After this, the experts whose knowledge we will take of will be presented. The skills and knowledge of these experts will be shown, and it will show which give each other to the project. Also the potential users will be presented.

After that, it will be shown in which situation it's found this kind of systems, analyzing if there are other similar works, or what helpful information we can take. Next some related concepts are going to be explained, referred to the application field and so to the field of Expert System. To warrant and check the viability and whether it is worth to go away with the project the Slagle Test is going to be done.

## 1. Introducción

En esta sección se va a realizar una breve introducción acerca del Sistema Experto de recomendación de lenguaje de Programación en el que quedará definido el problema, definiendo a su vez el alcance que abarcará dicho sistema experto y de cuáles son los objetivos que satisfará el mismo. Posteriormente se presentarán los expertos de los cuales se obtendrá el conocimiento y a qué tipo de usuario estará dirigido este Sistema Experto

### 1.1. Planteamiento del Problema

El mundo de la informática es demasiado grande y muchas veces la tecnología y tendencias actuales suelen variar con rapidez, esto puede ser desconcertante para las personas que dedican su vida diaria a este campo, entre ellos, los informáticos. Muchas veces, hay quien se puede encontrar en la situación de querer abordar un nuevo problema, o bien uno quiere especializarse en un campo específico, pero no conoce las herramientas apropiadas para ello, en este caso, los lenguajes de programación, los cuales se caracterizan porque no existe uno que predomine sobre todos, ya que cada uno se adapta mejor a unos problemas que otros abordan peor.

### 1.2. Objetivos y Alcance

El objetivo de este sistema experto es proporcionar al usuario un lenguaje de programación que se adapte a sus problemas y necesidades, pudiendo estas diferir de varias formas.

### 1.3. Los Expertos

El conocimiento será adquirido principalmente de 3 expertos, siendo todos profesores con actividades docentes en la Escuela Superior de Informática de la UCLM

- **Pascual Julián Iranzo:** Doctorado en Ciencias de la Computación, con actividad docente en Lógica y Programación declarativa durante 18 años, experto en Sistemas Inteligentes proporcionará el conocimiento necesario acerca de la rama de la Programación Declarativa, sus ventajas frente a la Programación Imperativa así como de Sistemas Inteligentes.
- **David Villa Alises:** Doctorado en Ingeniería Informática, miembro del grupo de investigación **Arco** con actividad docente en Redes de Computadores y Sistemas Distribuidos.
- **Ismael Caballero Muñoz-Reja:** Experiencia profesional en el mundo de la informática, proporcionará la visión más empresarial acerca de los lenguajes de programación, actualmente desempeña su actividad docente en la Escuela Superior de Informática de la UCLM en asignaturas como Ingeniería del Software II

#### 1.4. Los usuarios finales del sistema

Los usuarios serán, en su mayoría, personas que tienen un interés especial en la informática para realizar un determinado desarrollo, pudiendo ser para fines lucrativos, mero aprendizaje, práctica, etc.

Tras recibir las necesidades del usuario, el sistema responderá realizando una simulación del juicio del experto recomendando el lenguaje de programación que crea más apropiado

## 2. Estado del Arte

En el estado del arte se presenta un breve estudio sobre la presencia de cómo se ha llevado este problema al campo de los sistemas expertos por parte de otras personas, así como artículos en los que se describa esta problemática. Se abordará brevemente cada uno de estos artículos con un breve resumen.

También se introducirán los conceptos más importantes tanto del sistema experto como de los lenguajes de programación.

### 2.1. Revisión Bibliográfica

Tras una búsqueda exhaustiva, solo se ha encontrado un único sistema experto [1] el cual en base a unas preguntas cortas determina que lenguaje de programación es el que debes usar, su código fuente está publicado, y se puede observar que es bastante sencillo.

Se puede encontrar un artículo[2] acerca de qué lenguaje de programación elegir como primer lenguaje de programación para aprender a programar. Recomienda como primer lenguaje Python.

Dado que en el sistema experto se abordará el tema empresarial, es interesante ver una comparativa de los lenguajes de programación mejor pagados [3] ya que algunas personas pueden interesarles únicamente el punto de vista económico, visto desde esta perspectiva, los desarrolladores de Ruby on Rails, Objective C y Python son los lenguajes mejor pagados de la industria.

Otra de las grandes áreas de aplicación de la programación es la programación web, en este artículo [4] se puede observar una comparativa de algunos de los lenguajes de programación más interesantes para afrontar un desarrollo web, siendo HTML y CSS los más básicos, luego se ve diferenciada por los que se ejecutan desde la aplicación cliente y desde la servidora.

Es especialmente interesante este artículo [5]. Permite ver de una forma clara y sencilla que se espera de cada perfil además de una breve descripción de los lenguajes, por ejemplo, los desarrolladores de videojuegos tienden a usar más los lenguajes C, C++, C# y Java, trabajan bien bajo presión, y suelen tener otras habilidades relacionadas con el diseño.

### 3. Estudio de viabilidad. Test de Slagle

Se ha realizado un estudio de viabilidad para determinar si este proyecto lo es. Se ha utilizado la técnica conocida como Test de Slagle, y se ha obtenido una puntuación del 91.38

### 4. Adquisición del conocimiento

La adquisición del conocimiento es la extracción o recogida del conocimiento de la fuente experta para obtener el conocimiento heurístico y la experiencia en la resolución de problemas de un determinado dominio por parte del ingeniero, con el fin de que sea usado por un programa. Requiere de una larga e intensa interacción. En este caso las fuentes de las que se extraerá dicho conocimiento serán las fuentes citadas en las referencias y los expertos nombrados antes.

#### 4.1. El Proceso de las Entrevistas

En el caso de este proyecto se ha decidido optar por un método manual, como son las entrevistas, como método principal de extracción del conocimiento a los expertos, si bien, se ha recurrido también a otras fuentes como complemento a estas entrevistas.

#### 4.2. Entrevistas y sus resultados

En este caso, se han utilizado tanto entrevistas no estructuradas como estructuradas. Las entrevistas han sido determinantes en el proceso de extracción del conocimiento. Sin estas entrevistas no se hubiese podido llevar a cabo gran parte del trabajo ya que existen pocos recursos sobre este tema, y mucho menos opiniones y experiencias de personas con experiencia en esta materia. Las entrevistas se han realizado de la siguiente manera:

- Primera reunión con el experto en la que se ha explicado en que iba a consistir del sistema, y se han establecido los principales puntos que se deberían tratar. Esta entrevista correspondería al grupo de no estructuradas.
- Se le han proporcionado al experto una serie de preguntas que iban a ser la espina dorsal de la entrevista para que pudiese preparárselas.
- Se ha procedido a una entrevista estructurada.
- Finalmente se ha contrastado lo comentado por los expertos con otras fuentes de conocimiento.

Respecto a las variantes de entrevistas utilizadas, decir que han sido mayoritariamente orientadas a una discusión focalizada. También se ha usado el método de la reclasificación o descomposición de metas, se le han puesto casos generales de como lograr un objetivo, y el experto ha planteado distintas posibilidades.

### 4.3. El proceso de conceptualización

Antes de pasar a la etapa de representación del conocimiento es necesario una etapa intermedia denominada conceptualización, que consiste en hacer explícitos los conceptos claves y las relaciones relevantes entre estos. En esta etapa se establecen que conocimientos maneja el experto, cómo los utiliza, dónde los emplea y cuándo los usa, permitiendo de esta manera obtener una descripción detallada del problema mediante modelos conceptuales. Estos modelos conceptuales son una especificación de lo que debe hacer el sistema pero no especifican el cómo.

### 4.4. Conceptualización y sus resultados

Durante este proceso se determinan tres tipos de conocimientos, los Estratégicos, los conocimientos Tácticos de Acción u Operativos, y los conocimientos Fáticos o Declarativos.

**Conocimientos Factuales** Especifican lo que es, o se cree que es verdad acerca del mundo en general y del caso particular para el cual se está ejecutando la tarea.

Primero de todo se ha procedido a la elaboración de un glosario, en el que se recogen los conceptos mas importantes del campo que nos ocupa que es el de la informática. Se presenta un glosario resumido

#### A

Aplicación: Programa diseñado para una determinada función

#### B

Base de datos relacional: Colección de datos organizada y relacionada, para evitar duplicaciones y permitir la obtención de datos combinados, satisfaciendo la necesidad de usuarios con diferentes necesidades de información.

#### C

Cliente: Programa que demanda servicios de otra computadora llamada servidor, y se hace cargo de la interacción necesaria con el usuario.

Código fuente: Programa escrito en un lenguaje de programación de alto nivel por un programador.

Compilar: Generar un programa en lenguaje de máquina a partir de un lenguaje de programación de alto nivel.

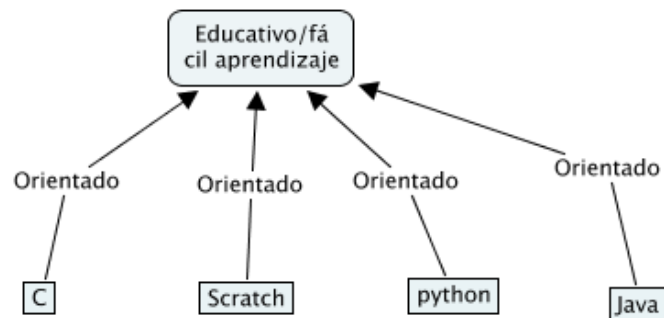
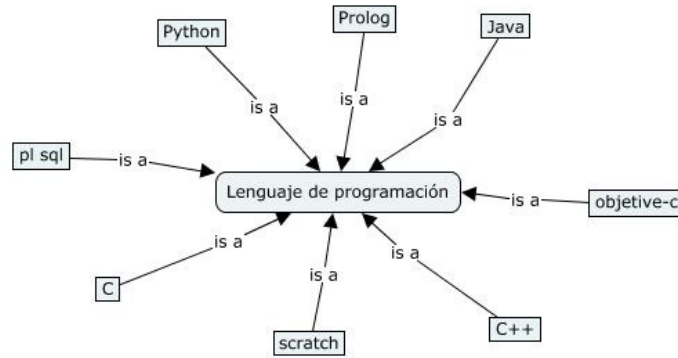
#### L

Lenguaje de programación: Conjunto de sentencias utilizadas para escribir secuencias de instrucciones para ser ejecutadas en una computadora.

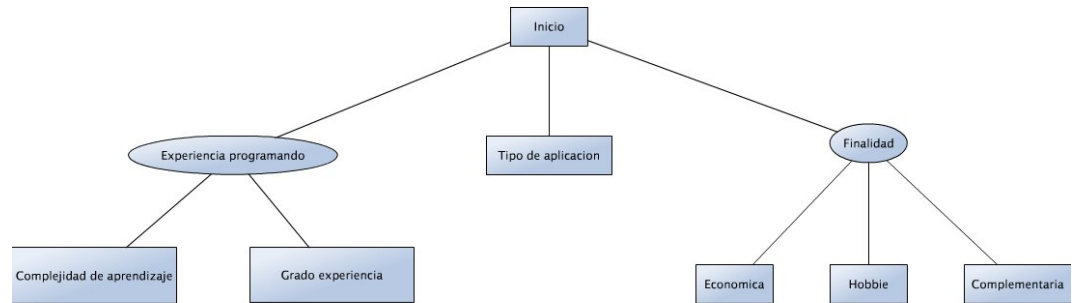
#### S

Software: Programas escritos en un lenguaje que la computadora entiende y puede ejecutar para realizar una tarea.

Aquí se muestran también algunas ontologías sobre los lenguajes de programación que se usarán.



**Conocimientos Estratégicos** Especifican qué hacer, dónde y por qué hacerlo, es decir, fijan la secuencia de pasos que el SE deberá seguir para ejecutar su tarea. A continuación se especifican una serie de pseudoreglas que definen como funciona el sistema.



**Conocimientos Tácticos** De acción u operativos, especifican cómo y cuándo el SE puede añadir a sus conocimientos genéricos información actual acerca del caso.

Se muestran algunos ejemplos de las pseudoreglas usadas.

<b>R1</b>	
Si	tiene experiencia
Entonces	Exp
<b>R2</b>	
Si	no tiene experiencia
Entonces	Aprender
<b>R2</b>	
Si	Aprender
Y	curva de aprendizaje muy fácil
Entonces	AppInventor

## 5. Implementación

Tras haber llevado a cabo todo el proceso de conceptualización toca el de implementación. Primero se mostrará un primer boceto de como se ha planteado el sistema de manera genérica haciendo uso de una herramienta de maquetado, y posteriormente se analizara la tecnología utilizada para el desarrollo de la propia interfaz y finalmente la interfaz propiamente dicha.

### 5.1. Interfaz Gráfica de Usuario

La interfaz de usuario es una parte muy importante de cualquier programa informático. Es la parte que el usuario aprecia de manera directa, y muchas veces sirve para juzgar cualquier sistema antes de verlo en funcionamiento. Algunos

datos interesantes son *"La interfaz constituye entre el 47 % y el 60 % de las líneas de código"* (McIntyre) o *"Actualmente más del 70 % del esfuerzo de desarrollo de las aplicaciones interactivas está dedicado a la interfaz"* (Gartner Group).

## 6. Conclusiones y trabajo futuro

Como conclusiones, destacar que se ha conseguido llevar a cabo el proyecto previsto, proporcionando una herramienta que de manera sencilla aconseja o recomienda a un público muy variado el lenguaje de programación que mejor se adapte a sus necesidades. De la evaluación del estado del arte se ha podido comprobar que no existe ningún trabajo similar (al menos serio). Casi toda la información relacionada con este tipo de sistemas que una persona sin un alto grado de experticidad puede tener a su disposición esta recogida en blogs y es mayoritariamente de opiniones de usuarios amateurs. Por el contrario este proyecto cuenta con la participación de grandes expertos y con las fuentes mas fiables de las que se ha podido disponer.

Por supuesto al ser un mercado tan cambiante el mercado tecnológico con el que tan ligados están los lenguajes de programación, este sistema necesita de una actualización continua. Además estas actualizaciones se podrían automatizar extrayendo de alguna fuente de información fiable que lenguajes se usan mas y para que, aunque este seria un cambio completo de orientación del proyecto.



## Referencias

1. Kish Waukee College. Expert System for Choosing a Programming Language
2. Vladyslav Kruglyk and Michael Lvov. Kherson State University. Ukraine. <http://ceur-ws.org/Vol-848/ICTERI-2012-CEUR-WS-paper-37-p-188-198.pdf>
3. Lisa Eadicicco Business Insider. (2014). <http://www.businessinsider.com/best-tech-skills-resume-ranked-salary-2014-11>
4. Yoshitaka Shiotsu. (2014). <https://www.odesk.com/blog/2014/03/web-development-101-top-web-development-languages-2014/>
5. Kasia Mikoluk. (2013). <https://blog.udemy.com/best-programming-language/>