

**UNIVERSIDAD NACIONAL**  
**“SANTIAGO ANTÚNEZ DE MAYOLO”**  
**FACULTAD DE CIENCIAS**  
**ESCUELA ACADÉMICO - PROFESIONAL DE**  
**INGENIERÍA DE SISTEMAS E INFORMÁTICA**



**“SISTEMA EXPERTO DETERMINÍSTICO EN PROLOG DE**  
**ORIENTACIÓN VOCACIONAL PARA EL APOYO A LA DIRECCIÓN**  
**DE BIENESTAR UNIVERSITARIO - UNASAM”**

**SISTEMAS EXPERTOS E INTELIGENCIA ARTIFICIAL**

**ELABORADO POR:**

ARAUCANO MINAYA, César Iván

EVANGELISTA FLORES, María Soledad

GRANADOS HERNÁNDEZ, Eduardo

PARIACHI DUEÑAS, Doris Liseth

**DOCENTE:**

Ing. SILVA ZAPATA, Miguel Ángel

**HUARAZ – PERÚ**

**2021**

## INTRODUCCIÓN

En la actualidad la UNASAM no cuenta con las herramientas necesarias para el apoyo en la orientación vocacional de los futuros postulantes , estamos hablando precisamente de La dirección de bienestar universitario.

La mayoría de los alumnos no tienen una idea de qué profesión elegir, debido a que a veces no se tiene una orientación vocacional necesaria que les ayude a decidir una toma de decisión tan importante pero se debe tener muy en cuenta este aspecto ya que la carrera que el alumno elegirá será importante para su vida, también conocer cuales son las habilidades y el potencial que tienen les hará más fácil el estudio.

Al carecer de una orientación vocacional, el estudiante puede tomar decisiones erróneas, que muchas veces puede repercutir en el estado emocional, económico y familiar, ya que una mala decisión provoca un malestar que influye en todos estos aspectos.

Si el alumno toma una buena elección, este tendrá un impacto positivo en su vida estudiantil, para ello pensamos implementar un sistema experto utilizando prolong con su programación declarativa y funcional, en los siguientes apartados, describiremos brevemente los conceptos más importantes de los sistemas expertos, la metodología que se usará para la programación y daremos a conocer los resultados que obtendremos.

## ÍNDICE

INTRODUCCIÓN	2
ÍNDICE	3
RESUMEN	5
ABSTRACT	6
1. El problema de la investigación.	7
1.1. Planteamiento del problema	7
1.2. Formulación del problema.	8
1.2.1. Problema General	8
1.2.2. Problemas Específicos.	8
1.3. Objetivos de la investigación.	9
1.3.1. Objetivo General	9
1.3.2. Objetivos específicos.	9
1.4. Justificación de la investigación.	10
1.4.1. Justificación teórica.	10
1.4.2. Justificación práctica.	10
1.4.4. Justificación Operativa.	10
1.4.5. Justificación Económica.	10
1.4.6. Justificación Tecnológica.	11
2. Marco Teórico.	12
2.1. Antecedentes de la investigación.	12
2.1.1. Antecedentes Internacionales.	12
2.1.2. Antecedentes Nacionales.	12
2.2. Bases Teóricas.	13
2.2.1. Orientación Vocacional	13
2.2.2. Sistema Experto	14
2.2.3. Prolog	21
2.3. Definición de términos.	25
2.4. HIPÓTESIS.	26
2.4.1. Hipótesis General	26
2.4.2. Hipótesis Específicas.	26
2.5. Variables.	27
2.5.1. Variable Independiente.	27
2.5.2. Variable Dependiente.	27
2.5.3. Operalización de variables	28

<b>3. IMPLEMENTACIÓN</b>	29
<b>3.1. Instalación de Librerías</b>	29
<b>3.2. Implementación de la Interfaz Principal</b>	29
<b>3.3. Implementación de Interfaz</b>	30
<b>3.4. Reglas para tomar la ruta</b>	32
<b>3.5. Creación del diálogo de preguntas</b>	32
<b>3.6. Botones de verificación</b>	33
<b>3.7. Agregar Label y botones de cuadro de diálogos</b>	34
<b>3.8. Llamada de sentencias para determinar respuestas</b>	34
<b>3.9. Vocación respecto a respuestas dadas</b>	34
<b>Bibliografía</b>	36

## **RESUMEN**

En el presente trabajo planteamos la creación de un sistema experto que nos ayudará en el proceso de orientación vocacional para los futuros estudiantes de la Universidad Nacional Santiago Antunez de Mayolo - UNASAM. Lo que busca este sistema realizado en Prolog es contar con una herramienta que permita al alumno identificar su verdadera vocación según sus habilidades y competencias.

Este sistema experto de orientación vocacional para el apoyo en la Dirección de Bienestar Universitario busca brindar la herramienta necesaria para el profesional que trabaja en el servicio de orientación vocacional, de esta manera ayudar a los futuros estudiantes con una buena elección para que ellos tengan un impacto positivo en su carrera universitaria.

## **ABSTRACT**

In this paper we propose the creation of an expert system that will help us in the vocational guidance process for future students of the Santiago Antúnez de Mayolo National University - UNASAM. What this system made in Prolog seeks is to have a tool that allows the student to identify their true vocation according to their skills and competencies.

This expert system of vocational guidance for support in the University Welfare Directorate seeks to provide the necessary tool for the professional who works in the vocational guidance service, in this way to help future students with a good choice so that they have an impact positive in his college career.

## **1. El problema de la investigación.**

### **1.1. Planteamiento del problema**

Actualmente uno de los problemas más frecuentes en la educación es la deserción universitaria, esta es influenciada por diferentes factores ya sean sociales, económicos y/o académicos, y dentro de ellos la falta de orientación vocacional juega un rol importante en todo este proceso. Puesto que los postulantes en muchas ocasiones no saben cuál sería la carrera correcta dependiendo de sus aptitudes y facultades.

La Universidad Nacional Santiago Antúnez de Mayolo cuenta con la oficina de Dirección de Bienestar Universitario, que tiene como una de sus funciones orientar a los postulantes a descubrir cuál es la carrera perfecta para ellos.

En muchas ocasiones la mala elección de la carrera nos lleva a observar diferentes problemas ocasionados por este, como bien pueden ser el relevo de deserciones, el bajo rendimiento, el cambio de carreras cuando ya este haya avanzado, entre otros.

Para enfrentar el problema en particular como es la elección correcta de la carrera profesional, se puede apoyar en la construcción de un Sistema Experto, siendo este una rama de la Inteligencia Artificial, se podrá emular la inteligencia del humano experto para poder realizar un sistema tal que pueda ayudar a que todo este proceso se realice de manera tanto óptima como eficiente.

## **1.2. Formulación del problema.**

### **1.2.1. Problema General**

¿En qué medida la implementación de un Sistema Experto determinístico en Prolog mejorará el servicio de Orientación Vocacional brindado por la Dirección de Bienestar Universitario en la UNASAM, 2021?

### **1.2.2. Problemas Específicos.**

- ¿En qué medida la implementación de un Sistema Experto determinístico en Prolog influye en el número de solicitudes para utilizar el servicio de Orientación Vocacional brindado por la Dirección de Bienestar Universitario en la UNASAM, 2021?
- ¿En qué medida la implementación de un Sistema Experto determinístico en Prolog influye en el tiempo en el que se realiza el test de Orientación Vocacional brindado por la Dirección de Bienestar Universitario en la UNASAM, 2021?
- ¿En qué medida la implementación de un Sistema Experto determinístico en Prolog influye en el nivel de satisfacción del servicio de Orientación Vocacional brindado por la Dirección de Bienestar Universitario en la UNASAM, 2021?



### **1.3. Objetivos de la investigación.**

#### **1.3.1. Objetivo General**

Determinar en qué medida la implementación de un Sistema Experto determinístico en Prolog mejorará el servicio de Orientación Vocacional brindado por la Dirección de Bienestar Universitario en la UNASAM, 2021.

#### **1.3.2. Objetivos específicos.**

- Determinar en qué medida la implementación de un Sistema Experto determinístico en Prolog influye en el número de solicitudes para utilizar el servicio de Orientación Vocacional brindado por la Dirección de Bienestar Universitario en la UNASAM, 2021.
- Determinar en qué medida la implementación de un Sistema Experto determinístico en Prolog influye en el tiempo en el que se realiza el test de Orientación Vocacional brindado por la Dirección de Bienestar Universitario en la UNASAM, 2021.
- Determinar en qué medida la implementación de un Sistema Experto determinístico en Prolog influye en el nivel de satisfacción del servicio de Orientación Vocacional brindado por la Dirección de Bienestar Universitario en la UNASAM, 2021

#### **1.4. Justificación de la investigación.**

##### **1.4.1. Justificación teórica.**

La presente investigación se justifica teóricamente, ya que la implementación del sistema sentará bases teóricas para poder realizar futuros trabajos de investigación relacionados con la mejora de los servicios brindados por la UNASAM, utilizando Sistemas Expertos.

##### **1.4.2. Justificación práctica.**

La presente investigación tiene justificación práctica, ya que la implementación del sistema nos permitirá servir de guía al momento de implementar sistemas relacionados, para automatizar servicios brindados en la UNASAM.

##### **1.4.4. Justificación Operativa.**

La justificación operativa del presente proyecto se basa en brindar una alternativa para optimizar el servicio de manera virtual a través de Sistemas Expertos, de esta manera mejorar el servicio de Orientación Vocacional brindado por la Dirección de Bienestar Universitario en la UNASAM.

##### **1.4.5. Justificación Económica.**

La implementación de un Sistema Experto determinístico en Prolog en Inteligencia Artificial para gestionar incidentes y requerimientos busca reducir los costos de transporte hacia el local del Ministerio Público, ahorrando tiempo y dinero.

#### **1.4.6. Justificación Tecnológica.**

Se proporcionará un Sistema Experto determinístico en Prolog para mejorar el servicio de Orientación Vocacional brindado por la Dirección de Bienestar Universitario en la UNASAM, la cual se implementará en el portal web y permitirá mejorar la calidad de la atención.

## **2. Marco Teórico.**

### **2.1. Antecedentes de la investigación.**

#### **2.1.1. Antecedentes Internacionales.**

(Kolovós Ortega and Guerrero Rayas, 2015) En su tesis para obtener su título profesional de ingeniería titulado “Sistema experto para orientación vocacional de educación media-superior” nos dice que plantean que la finalidad de dicho sistema es contar con una herramienta que permita al alumno identificar sus verdaderas competencias y habilidades, para dicho problema recolecta la información de las carreras en Celaya, creación de la base de conocimientos, base de datos y creación de reglas logrando disponer de la información necesaria acerca de las carreras que actualmente se ofrecen en la región, a su vez será una herramienta de apoyo para el orientador vocacional en su labor de asesoramiento a los alumnos sobre su futuro profesional.

El aporte de esta tesis al plan es que los alumnos quedaron satisfechos con la respuesta del sistema pues de acuerdo a sus habilidades y conocimientos el sistema logró tener una predicción correcta sobre la carrera de su agrado.

#### **2.1.2. Antecedentes Nacionales.**

(Tapia Castillo, 2009) En su tesis para obtener el título profesional de ingeniería titulado “Sistema experto para el apoyo del proceso de orientación vocacional para las carreras de ingeniería en la Pontificia Universidad Católica del PERÚ” plantea la creación de un sistema experto que sirva de apoyo al proceso de orientación vocacional de las carreras de Ingeniería de la Pontificia Universidad

Católica del Perú. La finalidad de dicho sistema es contar con una herramienta que permita al alumno identificar sus verdaderos intereses y habilidades así como disponer de la información necesaria acerca de las opciones vocacionales existentes, a su vez será una herramienta de apoyo para el orientador vocacional en su labor de asesoramiento a los alumnos sobre su futuro profesional.

Esta tesis aporta al plan, ya que gracias a este sistema lograremos que una mayor cantidad de alumnos puedan recibir a tiempo una adecuada orientación sobre su futuro profesional.

## **2.2. Bases Teóricas.**

### **2.2.1. Orientación Vocacional**

El proceso de orientación vocacional tiene por fin conocer los factores que influyen en la elección de carrera, definiendo el concepto de orientación vocacional, se conocerá la labor del orientador vocacional y se detalla el proceso. La elección de carrera es una de las decisiones más importantes que el individuo debe afrontar, y muchas veces este proceso puede experimentarse con tensión ante la inseguridad de los futuros cambios y el miedo al fracaso. Esta decisión es trascendente en el desarrollo del ser humano para que este no termine siguiendo una profesión que no está relacionada con sus intereses y habilidades y que lo puede llevar al fracaso. La orientación vocacional busca que el ser humano pueda afrontar con éxito la elección profesional. Para asegurar una buena elección se tiene que evaluar la habilidad o capacidad que tiene el individuo de realizar alguna actividad y los intereses que posee, logrando así delimitar campos de interés que finalmente podrán ser relacionados con campos profesionales. En conclusión, son necesarios dos elementos claves para tomar una adecuada decisión: conocerse a uno mismo y

conocer las opciones vocacionales existentes: Estos dos elementos son contemplados en el proceso de orientación vocacional y es por esta razón que es de gran ayuda en el proceso de toma de decisión profesional, ya que ayuda a que el individuo tome una decisión de manera racional, analizando las alternativas existentes y conociendo sus habilidades, intereses y fortalezas. (Jackeline Tapia, 2011)

### **2.2.2. Sistema Experto**

Se puede entender como una rama de la inteligencia artificial. Estos sistemas imitan las actividades de un humano para resolver problemas de distinta índole (no necesariamente tiene que ser de inteligencia artificial). También se dice que un S.E se basa en el conocimiento declarativo (hechos sobre objetos, situaciones) y el conocimiento de control (información sobre el seguimiento de una acción). Un sistema experto es un sistema de cómputo que emula la habilidad de tomar decisiones de un especialista humano. El término emular significa que el sistema experto tiene el objetivo de actuar en todos los aspectos como un especialista humano. Una emulación es mucho más fuerte que una simulación, que en algunos aspectos solo requiere que se actúe como en la realidad. A pesar de que una solución a problemas de propósito general aún nos elude, los sistemas expertos funcionan bien en sus dominios restringidos. Como prueba de su éxito, solo se necesita observar las muchas aplicaciones que tienen hoy en los negocios, la medicina, la ciencia y la ingeniería, además de todos los libros, diarios, conferencias y productos dedicados a ellos. (Unidad de Desarrollo Tecnológico en Inteligencia Artificial Universidad Autónoma de Barcelona España, 2015)

Un sistema experto está conformado por:

**Base de conocimiento (BC):** Representación del conocimiento del dominio para la solución de problemas específicos, normalmente dicho conocimiento se estructura en forma modular en forma declarativa. Es una base de datos que almacena todo el conocimiento del sistema experto en forma de reglas. Este conocimiento comprende los datos que describen el problema, las reglas utilizadas, la forma de combinar estas reglas, los nuevos datos deducidos y las propuestas de solución. En la creación de una base de conocimientos se debe de tener en cuenta qué objetos serán definidos, cómo son las relaciones entre estos objetos, cómo se formularán y procesarán las reglas.

**Base de Hechos:** Contiene los datos del problema, así como los elementos y hechos relativos a la solución de un problema en particular. A su vez almacena la información dada por el usuario en respuesta a las preguntas del sistema.

**Motor de Inferencia** - Proceso que efectúa el razonamiento a partir de los datos y utilizando el conocimiento de la base de conocimiento. Es "genérica", es decir, que se puede aplicar a diferentes dominios sólo cambiando la Base de conocimiento.

Simula la estrategia de solución de un experto, obtiene conclusiones aplicando las reglas sobre los hechos presentes. Determina qué acciones se realizarán, en qué orden y cómo las realizarán las diferentes partes del sistema experto. Está compuesto de tres elementos:

1. **Solucionador de Problemas:** Contiene mecanismos de valoración de conocimientos, como mecanismos de inferencia, los cuales permiten deducir nuevos hechos o establecer nuevas hipótesis. Debe disponer de

técnicas de búsqueda para recorrer la base de conocimientos y de técnicas de selección.

2. **Módulo generador de explicaciones:** Es un subsistema que tiene la capacidad de explicar el razonamiento que conduce a cierta conclusión, para lo cual requiere de una interface con el usuario. El sistema debe acceder al registro de conocimientos que se emplearon durante el procesamiento de información y traducirlo en forma aceptable para el usuario.

3. **Módulo generador de conocimiento:** Contiene las funciones necesarias para mejorar el conocimiento del sistema experto. Debe ser capaz de actualizar la base de conocimientos ya sea comunicando nuevos conocimientos o modificando los existentes, verificando las entradas para identificar errores o inconsistencias.

**Interfaz de Usuario:** Entrada/Salida al usuario del sistema, incluyendo, normalmente, mecanismos de pregunta (por qué) y de explicación (cómo). Establece la forma en que el sistema experto se presentará al usuario, se debe establecer un diálogo en términos del problema y con construcciones del lenguaje humano correctas. Debe cumplir con los siguientes requisitos: el aprendizaje del manejo debe ser rápido, debe evitar la entrada de datos erróneos, las preguntas y resultados deben de presentarse en forma comprensible para el usuario.

Un sistema experto posee tres tipos diferentes de interfaces de usuario:

1. **Interface de componente de adquisición.** Usada por el ingeniero del conocimiento y por el usuario experto.



2. **Interfaz del componente explicativo.** Usada por el ingeniero del conocimiento, el usuario experto y el usuario del sistema.
3. **Interface de consulta.** Encargada de plantear preguntas, recibir respuestas y emitir resultados.

**Usuarios del Sistema Experto:** Los sistemas expertos poseen dos tipos de usuarios:

1. **Usuario Experto.** Se encarga de añadir nuevos conocimientos a la base de conocimientos o de modificar el conocimiento existente en el sistema.
2. **Usuario del Sistema.** Ejecuta el sistema experto.

**Arquitectura del Sistema Experto:** La arquitectura de los sistemas expertos es determinada por el motor de inferencias de acuerdo a las técnicas de razonamiento usadas:

- **Algoritmo de ejecución de un motor de inferencias:** El funcionamiento general de un motor de inferencias se da en dos etapas: Fase de evaluación y fase de ejecución.
- **Fase de evaluación:** En esta fase el motor de inferencia determina qué reglas pueden ser aplicadas de acuerdo a la situación actual de la base de hechos. Consta de tres etapas:
  1. **Restricción:** Consiste en determinar el conjunto de reglas y de hechos que serán usadas para resolver el problema. Para esto hace uso de los metaconocimientos sobre los hechos y las reglas.
  2. **El filtrado:** Se analiza el conjunto de reglas obtenidas en la etapa de restricción y se escogen aquellas cuya condición es verdadera si se toman en cuenta el conjunto de hechos obtenido. Se espera obtener de

este filtrado una sola regla, en el caso de que se obtenga más de una ocurre lo que se conoce como conflicto.

3. Resolución de conflictos. En esta última etapa se realiza la selección final de una regla. Esta selección puede estar basada en criterios relacionados directamente con la regla. Como, por ejemplo: fiabilidad y costo, y en criterios independientes de la regla como, por ejemplo: realizar una ordenación del conjunto de reglas y escoger la primera. El motor de inferencia después de ejecutar los criterios de selección puede obtener una regla, en cuyo caso el conflicto se soluciona, varias reglas, en cuyo caso el conflicto sigue o se pueden eliminar todas las reglas escogidas.

- Fase de ejecución: En esta fase se procede a ejecutar la(s) regla(s) escogidas en la primera fase, modificando la base de hechos. Si se eliminaron todas las reglas el motor de inferencia detiene la resolución, si el conflicto continúa se trata de activar otra regla.
- Encadenamiento entre los ciclos de base: Un ciclo base de un motor de inferencia está compuesto por las fases de evaluación y ejecución, se requiere de la ejecución de varios ciclos base para poder solucionar un problema. Es por esta razón que se requiere un encadenamiento de los ciclos base para que la resolución sea eficiente.

Se cuenta con tres estrategias de encadenamiento:

- Encadenamiento efectuado hacia adelante. Recorre la base de conocimientos de los hechos a las conclusiones y procede a seleccionar las reglas cuyas condiciones son verdaderas teniendo en cuenta los hechos.

- Encadenamiento hacia atrás. Recorre la base de conocimientos de las conclusiones a los hechos. Se determina una lista de objetivos y se procede a probar si los hechos correspondientes a estas conclusiones son verdaderos o falsos.
- Encadenamiento mixto. Es una combinación de las dos primeras estrategias. Se usa cuando el espacio de búsqueda es grande y en sistemas que trabajan en tiempo real.

**Estrategias de búsqueda:** Se cuenta con tres estrategias de búsqueda:

- Vertical. Solo se desarrolla una alternativa en un momento dado, tratando de ver si conduce a una solución correcta.
- Horizontal. Se desarrollan en paralelo todas las alternativas existentes en un momento dado. Esta estrategia debe ser usada cuando el espacio de búsqueda es pequeño, dicha búsqueda se realizará de manera exhaustiva.
- Ordenada. Se asocia una prioridad a las diferentes alternativas y se desarrolla la de mayor prioridad en un momento dado.

**Ventajas de los Sistemas Expertos:** Los Sistemas Expertos son programas de computadora en los que, a diferencia de los programas tradicionales, no se ha vertido una solución dado a un problema, sino el conjunto de conocimientos y reglas de operación de un experto humano, en torno a un problema específico, que le permiten al programa, a semejanza del experto humano, buscar la mejor solución, entre un gran número de posibilidades, atendiendo a la naturaleza del problema y a la situación contextual en el que éste se da. Además, los Sistemas Expertos tienen incorporadas facilidades de explicación de cómo llegaron a determinada solución y, en ciertos casos, mecanismos para aprender de la experiencia.

Las ventajas del uso de Sistemas Expertos son:

- **Autonomía.** Una vez que el Sistema Experto ha sido diseñado y completado, se hace autónomo, es decir, independiente de la presencia física del especialista.
- **Reproductibilidad.** El Sistema Experto en sí y el conocimiento que abarca son reproducibles a voluntad de su autor. Esto implica que, en caso de ser necesario, el producto puede reproducirse para dar servicio a miles de usuarios. Por otro lado, en contraposición con la lentitud del proceso de formación de mentores especialistas, la reproducción de un Sistema Experto es del orden de unos minutos solamente.
- **Bajo costo de adquisición y operación.** En tanto que disponer permanentemente de un experto supone altos costos, un Sistema Experto requiere únicamente pagar por su diseño y construcción. Este costo después se distribuye entre todos los usuarios del sistema, por lo que el costo por copia es muy reducido.
- **Facilidad de distribución.** Por la naturaleza del sistema, pueden cubrirse necesidades en localizaciones geográficas muy dispersas y bajo condiciones de trabajo difíciles.
- **Mínimos requerimientos de hardware.** Los requerimientos de hardware se limitan a una computadora PC.
- **Flexibilidad para modificaciones y expansión.** El Sistema Experto puede ser actualizado en el contenido de información, con fines de ampliación a profundización, con solo acceder a sus archivos de texto, ya que estos no forman parte integrante de las bases de conocimiento.

### 2.2.3. Prolog

Sancho (2014) nos dice que: Habitualmente, programar un ordenador significa dar una sucesión de tareas que, al ejecutarlas paso a paso, permiten resolver un problema concreto. Sin embargo, el proceso que se sigue en la programación lógica es completamente distinto: un programa en un lenguaje de programación lógica está formado por un conjunto de hechos, junto con un conjunto de condiciones que debe verificar la solución... el ordenador, usando un motor de inferencia, debe "deducir" la solución a partir de los hechos y las condiciones dadas.

Prolog (del francés, PROgrammation en LOGique) fue el primer lenguaje de programación basado en el paradigma de la programación lógica. Se implementó por primera vez a principios de los años setenta en la Universidad de Marsella (Francia), por un equipo dirigido por A. Colmeraeur y utilizando resultados de R. Kowalski (Universidad de Edimburgo). Aunque con ciertas dificultades iniciales, debido principalmente a la novedad del paradigma y a la escasa eficiencia de las implementaciones disponibles, el lenguaje se fue expandiendo rápidamente, sobre todo en Europa y en Japón (en este último país la programación lógica se incluyó como parte central del proyecto de ordenadores de quinta generación de los años ochenta) y en 1995 se normaliza con el correspondiente estándar ISO.

Teniendo en cuenta que es un lenguaje de programación que se utiliza para resolver problemas en los que existen objetos y relaciones entre objetos, la programación en Prolog consiste simplemente en:

- Declarar hechos sobre los objetos y sus relaciones.
- Definir reglas sobre dichos objetos y relaciones.
- Hacer preguntas.

Uno de los posibles usos de Prolog es como lenguaje de programación interactivo, lo que quiere decir que el ordenador y el programador sostienen una especie de conversación donde Prolog espera que se le introduzcan los hechos y reglas que definen el problema que se quiere resolver y, a continuación, si se hacen las preguntas adecuadas, buscará las respuestas y las presentará por pantalla.

Pese a la impresión inicial de tener capacidades limitadas, Prolog es un lenguaje muy versátil que puede implementar cualquier tipo de algoritmo, no únicamente aquellos para los que fue diseñado, por lo que no es menos potente que lenguajes como C++, Java o Haskell, y en algunos aspectos podría considerarse más potente. La elección de un lenguaje u otro depende, como siempre, del tipo de problema que se quiere resolver y del objetivo que se tenga al resolverlo.

Antes de comenzar a programar en Prolog se ha de tener en cuenta una importante consideración: es, con mucha probabilidad, radicalmente distinto a cualquier otro lenguaje de programación con el que hayamos trabajado; en consecuencia, cuando se quiere resolver un problema con él, no debe intentar resolverse previamente en otro lenguaje con el fin de traducirlo luego a Prolog (algo que es relativamente fácil y común entre otros lenguajes)... aquí no hay que buscar un algoritmo que resuelva el problema, basta proporcionar las bases lógicas suficientes para que el motor de inferencia de Prolog lo resuelva por nosotros... pero hay de reconocer bien qué información básica va a permitir que Prolog realice su tarea (cuando se entra en más profundidad en Prolog, se descubre que también puede influir el orden en que se proporciona esa información, pero ese aspecto no lo debatiremos aquí).

Los elementos de un programa en Prolog:

## Los hechos

Un hecho es un predicado (relación) entre objetos. Su sintaxis en Prolog es relación (objeto, objeto, ...). Ha de tenerse en cuenta lo siguiente:

- Los nombres de las relaciones deben comenzar con una letra minúscula.
- Los objetos se escriben separados por comas y encerrados entre paréntesis.
- Al final del hecho debe ir un punto.

## Las reglas

Las reglas funcionan como las fórmulas condicionales habituales en lógica.

Reflejan que la verdad de un hecho depende de la verdad de otro hecho o grupo de hechos. Consta de una cabeza y un cuerpo, donde este último puede estar formado por varios hechos (también llamados objetivos).

## Las variables

Los nombres de las variables deben comenzar con letra mayúscula o con el carácter (`_`). Existe una variable especial, la variable anónima o blanca, que se utiliza de la misma manera que las demás variables, pero nunca toma ningún valor.

La estructura de un programa en Prolog

La mayoría de los programas Prolog están organizados en cuatro secciones principales:

- Dominio: donde se declaran los argumentos que utilizarán los predicados.
- Predicados: donde se declaran todos los predicados no predefinidos que se utilizarán en las siguientes secciones.
- Objetivos: esta sección permite ejecutar los programas de forma no interactiva, y por tanto, buscará la solución deseada tan pronto como se ejecute el programa. Como también es habitual usar Prolog de forma interactiva es frecuente ejecutar un programa y luego esperar a que se nos pregunte por los objetivos.
- Cláusulas: donde se escriben los hechos y las reglas que conocemos del dominio.



### 2.3. Definición de términos.

- **Algoritmo:** Conjunto ordenado y finito de operaciones que permite hallar la solución de un problema.
- **Alternativa:** Opción entre dos cosas o más.
- **Carrera:** Estudios universitarios repartidos en una serie de años con los que se obtiene un título profesional.
- **Decidir:** Dar una solución o juicio definitivo sobre un asunto.
- **Experto:** Persona que tiene los conocimientos necesarios y suficientes para ser la base del sistema experto.
- **Inferencia:** Deducción de una cosa a partir de otra, conclusión.
- **Orientar:** Informar a uno de lo que ignora acerca de un asunto o negocio, o aconsejarle sobre la forma más acertada de llevarlo a cabo.
- **Racional:** Perteneciente o relativo a la razón, la lógica y la ciencia. Teoría.- Conocimiento especulativo considerado con independencia de toda aplicación.
- **Vocación:** Inclinação a cualquier estado, profesión o carrera.

## **2.4. HIPÓTESIS.**

### **2.4.1. Hipótesis General**

La implementación de un Sistema Experto determinístico en Prolog mejora el servicio de Orientación Vocacional brindado por la Dirección de Bienestar Universitario en la UNASAM, 2021.

### **2.4.2. Hipótesis Específicas.**

- La implementación de un Sistema Experto determinístico en Prolog influye en el número de solicitudes para utilizar el servicio de Orientación Vocacional brindado por la Dirección de Bienestar Universitario en la UNASAM, 2021.
- La implementación de un Sistema Experto determinístico en Prolog influye en el tiempo en el que se realiza el test de Orientación Vocacional brindado por la Dirección de Bienestar Universitario en la UNASAM, 2021.
- La implementación de un Sistema Experto determinístico en Prolog influye en el nivel de satisfacción del servicio de Orientación Vocacional brindado por la Dirección de Bienestar Universitario en la UNASAM, 2021.

## **2.5. Variables.**

### **2.5.1. Variable Independiente.**

X = Implementar un Sistema Experto determinístico en Prolog.

### **2.5.2. Variable Dependiente.**

Y= Servicio de Orientación Vocacional

### 2.5.3. Operalización de variables

Variable	Tipo de variable	Definición	Dimensión	Indicador	Instrumento de medición	Escala
Implementar un Sistema Experto determinístico en Prolog	Variable independiente	Se puede entender como una rama de la inteligencia artificial. Estos sistemas imitan las actividades de un humano para resolver problemas de distinta índole	Implementación de un Sistema Experto Determinístico	Número de ingresos realizados al portal web.	Análisis del flujo de control del Portal Web	Número
Servicio de orientación vocacional	Variable dependiente	El servicio brindado por la Dirección de Bienestar Universitario, el cual se desea mejorar	Satisfacción con el servicio brindado.	Satisfacción del Usuario.	Encuesta	Likert

### 3. IMPLEMENTACIÓN

#### 3.1. Instalación de Librerías

Se procede a instalar las librerías necesarias para que el IDE Prolog pueda activar el modo gráfico, estilos de la librería, la carga de imagen y el directorio de la imagen.

```
:-use_module(library(pce)). % libreria para trabajar modo gráfico
:-use_module(library(pce_style_item)). % activa los estilos de la libreria
:-pce_image_directory('./programa'). % Indica directorio donde se tiene la imagen para el programa
resource(imagen_inicio, image, image('im_inicio.jpg')). % cargue de imagen
:- dynamic color/2.
```

#### 3.2. Implementación de la Interfaz Principal

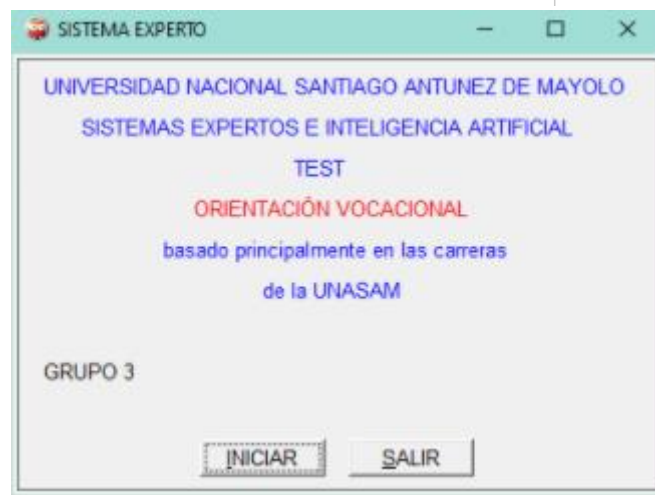
Se procede al diseño e implementación de la Interfaz Principal

```
comenzar:- new(D,dialog('SISTEMA EXPERTO',size(560,400))),% crea la ventana principal

new(Label1,label(text,'UNIVERSIDAD NACIONAL SANTIAGO ANTUNEZ DE MAYOLO')),% label contenido texto información inicial

    send(Label1,colour,blue),
new(Label2,label(text,'      SISTEMAS EXPERTOS E INTELIGENCIA ARTIFICIAL')),
    send(Label2,colour,blue),
new(Label3,label(text,'                                TEST')),
    send(Label3,colour,blue),
new(Label4,label(text,'                                ORIENTACIÓN VOCACIONAL ')),
    send(Label4,colour,red),
new(Label5,label(text,'                                basado principalmente en las carreras')),
    send(Label5,colour,blue),
new(Label6,label(text,'                                de la UNASAM')),
    send(Label6,colour,blue),
new(Label7,label(text,'')),
    send(Label7,colour,blue),
new(Label8,label(text,'GRUPO 3')),
    send(Label8,colour,black),

new(Label12,label(text,'')),
    send(Label12,colour,blue),
```



### 3.3. Implementación de Interfaz

- Se llama a los labels declarados inicialmente:

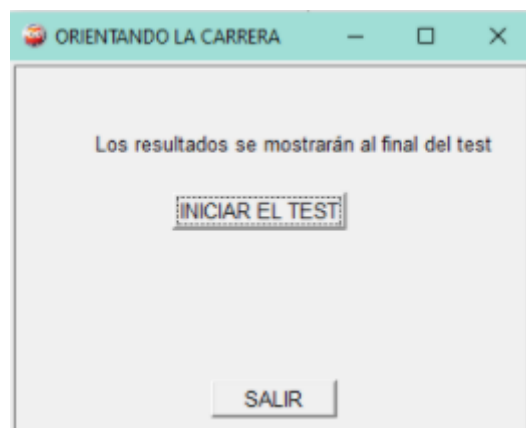
```
send(D, append(Label1)),  
send(D, append(Label2)),  
send(D, append(Label3)),  
send(D, append(Label4)),  
send(D, append(Label5)),  
send(D, append(Label6)),  
send(D, append(Label7)),  
send(D, append(Label8)),  
send(D, append(Label12)),
```

- Se declaran los botones de la ventana principal

```
new(Boton1, button('INICIAR', and(message(@prolog, main),  
and(message(D, open), message(D, free))))),  
send(Boton1, colour, black),  
new(Bcancelar, button('SALIR', and(message(D, destroy), message(D, free))))),  
send(Bcancelar, colour, black),  
send(D, append(Boton1)),  
send(D, append(Bcancelar)),  
send(D, open_centered).  
:-comenzar.
```

- Se hace un llamado al método principal

```
main:-  
new(D2, dialog('ORIENTANDO LA CARRERA', size(500, 400))),  
new(Label10, label(nombre, '')), send(Label10, colour, red),  
  
new(@texto, label(text, 'Los resultados se mostrarán al final del test')),  
new(@respl, label(text, '')),  
new(Salir, button('SALIR', and(message(D2, destroy), message(D2, free))))),
```



- Se crea el botón para dar inicio al TEST

```
new(@boton,button('INICIAR EL TEST',message(@prolog,botones))),

send(D2, append(Label10)),
new(@btncarrera,button('¿diagnostica?')),
send(D2, display,Label10,point(10,20)),
send(D2, display,@boton,point(100,80)),
send(D2, display,@texto,point(50,40)),
send(D2, display,Salir,point(125,200)),
send(D2, display,@respl,point(70,90)),
send(D2,open_centered).
```

- Se especifican las hipótesis de las vocaciones, se está haciendo uso del corte (!.) para que una vez validada una hipótesis se detenga y muestre el resultado.

```
hipotesis('Ingeniería de Sistemas e Informática'):-ingenieria_de_sistemas,!.
hipotesis('Enfermería'):-enfermeria,!.
hipotesis('Economía'):-economia,!.
hipotesis('Administración'):-administracion,!.
hipotesis(desconocido). /* no existe */
```

- Se detallan las profesiones, declarando todas las preguntas para estas.

### Para Ingeniería de Sistemas:

```
ingenieria_de_sistemas:-vocacion_ingenieria_de_sistemas,
consultar('¿Eres hábil resolviendo problemas?'),
consultar('¿Te gustan las matemáticas?'),
consultar('¿Resuelves cálculos con facilidad?'),
consultar('¿Te gusta resolver problemas utilizando herramientas tecnológicas?'),
consultar('¿Te gusta crear cosas?'),
consultar('¿Te emociona diseñar cosas?').
```

### Para Enfermería:

```
enfermeria:-vocacion_enfermeria,
consultar('¿Te gusta servir a los demás?'),
consultar('¿Eres paciente con las otras personas?'),
consultar('¿Te preocupa el estado actual del sistema de salud del país?'),
consultar('¿Si estuviese dentro de tus posibilidades, te gustaría descubrir la cura de alguna enfermedad?'),
consultar('¿Ayudarías a una persona que se encuentre herida?'),
consultar('¿Eres sensible al dolor ajeno?'),
consultar('tiene un cambio en los habitos intestinales').
```

### Para Economía:

```
economia:-vocacion_economia,
consultar('¿Te gustaría trabajar en un Banco?'),
consultar('¿Te gusta contar dinero?'),
consultar('¿Te interesan los temas relacionados a los mercados financieros?'),
consultar('¿Te gusta manejar finanzas?'),
consultar('¿Te interesa el contexto político actual en cuánto a los precios y alzas de valores monetarios?'),
consultar('¿Te apasionan los negocios?').
```

## Para Administración:

```
administracion:-vocacion_administracion,  
  consultar('¿Tienes liderazgo al trabajar en equipo?'),  
  consultar('¿Sientes que estás capacitado para contribuir a un mejor rendimiento de una empresa?'),  
  consultar('¿Tienes habilidades para distribuir tareas grupales?'),  
  consultar('¿Te gustan los cursos de economía, sociología y contabilidad?'),  
  consultar('¿Te interesan estudios, producción y la exploración de una empresa?').
```

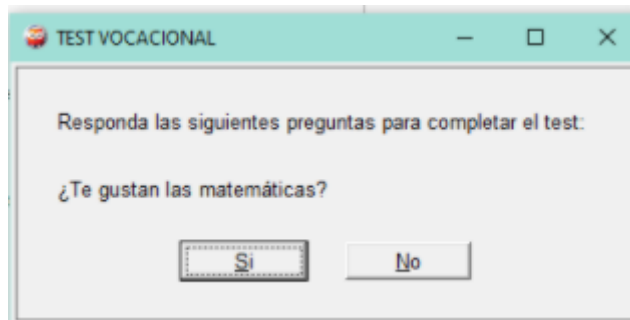
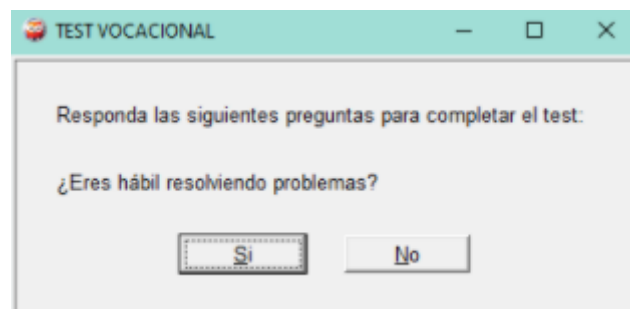
### 3.4. Reglas para tomar la ruta

```
vocacion_ingenieria_de_sistemas:-consultar('¿Eres hábil resolviendo problemas?'),!.  
vocacion_enfermeria:-consultar('¿Te gusta servir a los demás?'),!.  
vocacion_economia:-consultar('¿Te gustaría trabajar en un Banco?'),!.  
vocacion_administracion:-consultar('¿Tienes liderazgo al trabajar en equipo?'),!.
```

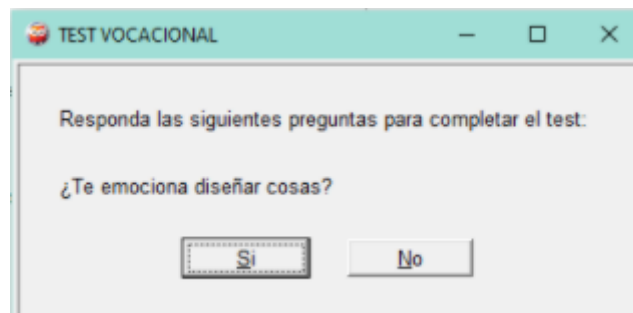
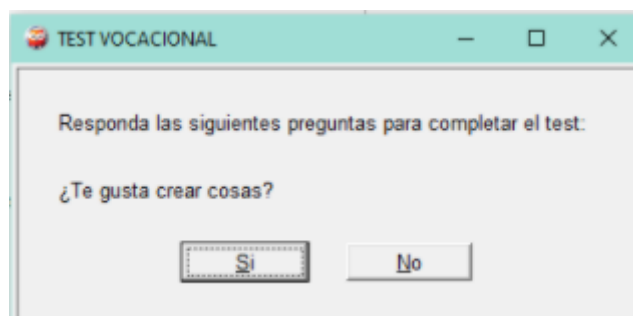
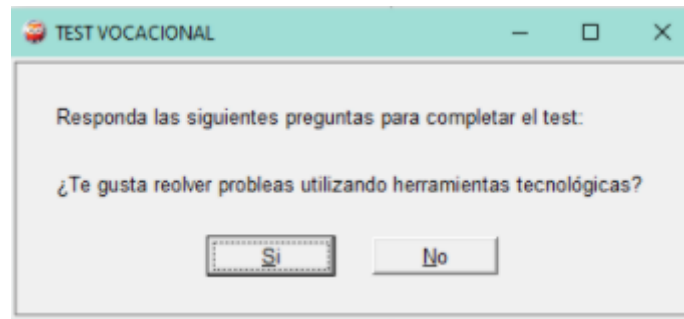
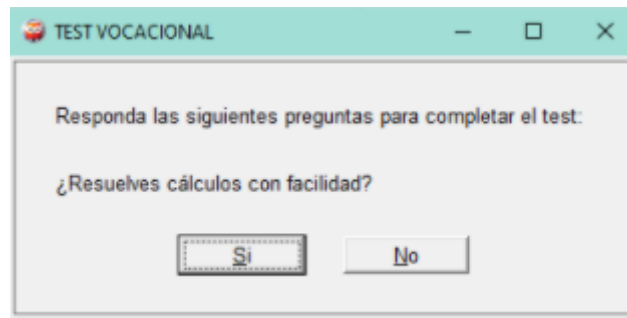
### 3.5. Creación del diálogo de preguntas

```
:-dynamic si/1,no/1.  
  
preguntar(Problema):-new(Di,dialog('TEST VOCACIONAL')),  
  new(L9,label(texto,'Responda las siguientes preguntas para completar el test:')),  
  new(L10,label(text,Problema)),
```

Nos mostrará las siguientes ventanas:







### 3.6. Botones de verificación

```
new (B1, button (si, and (message (Di, return, si)))),
new (B2, button (no, and (message (Di, return, no)))),
send (Di, gap, size (25, 25)),
```

### 3.7. Agregar Label y botones de cuadro de diálogos

```
send(Di, append(L9)),
send(Di, append(L10)),
send(Di, append(B1)),
send(Di, append(B2)),

send(Di, default_button, si),
send(Di, open_centered), get(Di, confirm, Answer),
write(Answer), send(Di, destroy),
```

### 3.8. Llamada de sentencias para determinar respuestas

```
( (Answer==si) -> assert(si(Problema));
assert(no(Problema)), fail).

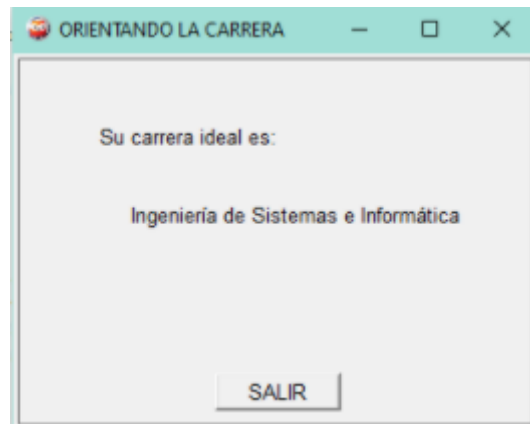
consultar(S):- (si(S)->true; (no(S)->fail; preguntar(S))).
deshacer:- retract(si(_)), fail.
deshacer:- retract(no(_)), fail.
deshacer.

botones:- borrado,
    send(@boton, free),
    send(@btncarrera, free),
```

### 3.9. Vocación respecto a respuestas dadas

```
hipotesis(Vocacion),
    send(@texto, selection('Su carrera ideal es:')),
    send(@respl, selection(Vocacion)),
    new(@boton, button('Iniciar su evaluacion', message(@prolog, botones))),
    send(Menu, display, @boton, point(40, 600)),
    send(Menu, display, @btncarrera, point(20, 50)),
    send(Menu, append, new(Ayuda, popup(Ayuda))),
    deshacer.

borrado:- send(@respl, selection('')).
```



Resultado final siguiendo la Ruta de Ingeniería de Sistemas e Informática.

## Bibliografía

Kolovós Ortega, Erik, and Karla Guerrero Rayas. "Sistema experto para orientación vocacional de educación media-superior." *o, Instituto Tecnológico de Celaya.*, Diciembre 2015, file:///C:/Users/hp/Downloads/306-613-2-PB%20(1).pdf. Accessed 13 mayo 2021.

Sancho, Fernando. "Una Introducción a Prolog." *Introducción al Aprendizaje*, Fernando Sancho Caparrini, 7 enero 2014, <http://www.cs.us.es/~fsancho/?e=73>. Accessed 13 mayo 2021.

Tapia Castillo, Jakeline. "SISTEMA EXPERTO PARA EL APOYO DEL PROCESO DE ORIENTACIÓN VOCACIONAL PARA LAS CARRERAS DE INGENIERÍA EN LA PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ." *Repositorio PUCP*, licencia Creative Commons, abril 2009, [http://tesis.pucp.edu.pe/repositorio/bitstream/handle/20.500.12404/366/TAPIA\\_JACKELINE\\_SISTEMA\\_EXPERTO\\_PARA\\_EL\\_APOYO\\_DEL\\_PROCESO\\_DE\\_ORIENTACION\\_VOCACIONAL\\_PARA\\_LAS\\_CARRERAS\\_DE\\_INGENIERIA\\_EN\\_LA\\_PONTIFICIA\\_UNIVERSIDAD\\_CATOLICA\\_DEL\\_PERU](http://tesis.pucp.edu.pe/repositorio/bitstream/handle/20.500.12404/366/TAPIA_JACKELINE_SISTEMA_EXPERTO_PARA_EL_APOYO_DEL_PROCESO_DE_ORIENTACION_VOCACIONAL_PARA_LAS_CARRERAS_DE_INGENIERIA_EN_LA_PONTIFICIA_UNIVERSIDAD_CATOLICA_DEL_PERU).pd. Accessed Jueves mayo 2021.