



SISTEMA INTELIGENTE WEB ADAPTATIVO PARA LA RECOMENDACIÓN DE DESTINOS TURÍSTICOS EN EL CANTÓN MORONA.

Autores: ¹Flores Orozco Ángel Patricio.

aflores@epoch.edu.ec

² Hurtado Jerves Julio Cesar.

jhurtado@epoch.edu.ec

³ Zabala Bustamante Brayan Andrés.

bzabala@epoch.edu.ec

Docente ESPOCH Ext. Morona Santiago.

Para citar este artículo puede utilizar el siguiente formato:

Flores Orozco Ángel Patricio, Hurtado Jerves Julio Cesar y Zabala Bustamante Brayan Andrés (2018): "Sistema inteligente web adaptativo para la recomendación de destinos turísticos en el Cantón Morona.", Revista Caribeña de Ciencias Sociales (mayo 2018). En línea: [//www.eumed.net/rev/caribe/2018/05/destinos-turisticos-ecuador.html](http://www.eumed.net/rev/caribe/2018/05/destinos-turisticos-ecuador.html)

RESUMEN

En el presente artículo se explica el proceso de desarrollo de un sistema inteligente web para la recomendación de destinos turísticos en el Cantón Morona. El sistema se diseñó con la finalidad de automatizar la recomendación de destinos turísticos en el cantón. El desarrollo de la aplicación se realizó utilizando la metodología Microsoft Solution Framework, que permitió la división del proyecto en iteraciones que generan entregables funcionales. La aplicación se desarrolló utilizando el entorno de desarrollo integrado (IDE) Pycharm for Education, con el cual fue posible la utilización de entornos virtuales para el desarrollo de la aplicación en Python. Se utilizó el paradigma orientado a objetos y la división en módulos de la aplicación desarrollada. Para verificar la exactitud de las recomendaciones turísticas, se probó la aplicación realizando 30 ejecuciones de la encuesta de selección de destinos, los resultados de esta inferencia fueron posteriormente validados por el usuario, obteniendo una aceptación satisfactoria de las recomendaciones brindadas por el sistema. De esta forma se pudo concluir que la utilización de un sistema inteligente permite que la recomendación de destinos turísticos se realice sin la presencia de un experto humano en turismo. Para asegurar la fiabilidad de la inferencia, se recomienda al personal de la oficina de turismo la realización de auditorías periódicas a la información ingresada por los usuarios encargados de la gestión del conocimiento.

¹ Ingeniero en Sistemas Informáticos, Magister en Informática Educativa. Docente de la ESPOCH Ext. Morona Santiago

² Ingeniero en Sistemas Informáticos. Operador en la Dirección Distrital 14006 Limón Indanza – Santiago – Tiwinza - Educación.

³ Ingeniero en Sistemas Informáticos.

ABSTRACT & KEYWORDS

In the present article the process of development of an intelligent web system for the recommendation of tourist destinations in the Morona Canton is explained. The system was designed with the aim of automating the recommendation of tourist destinations in the canton. The development of the application was made using the Microsoft Solution Framework methodology, which allowed the division of the project into iterations that generate functional deliverables. The application was developed using the integrated development environment (IDE) Pycharm for Education, with which it was possible to use virtual environments for the development of the application in Python. The object-oriented paradigm and the division into modules of the developed application were used. To verify the accuracy of the tourist recommendations, the application was tested by performing 30 executions of the destination selection survey, the results of this inference were subsequently validated by the user, obtaining a satisfactory acceptance of the recommendations provided by the system. In this way it was possible to conclude that the use of an intelligent system allows the recommendation of tourist destinations to be made without the presence of a human expert in tourism. To ensure the reliability of the inference, the staff of the tourism office is recommended to carry out periodic audits of the information entered by the users in charge of knowledge management.

Palabras claves:

<TECNOLOGÍA Y CIENCIAS DE LA INGENIERÍA>, <INGENIERÍA DE SOFTWARE>, <TURISMO>, <PROGRAMACIÓN DE APLICACIONES WEB>, <MORONA (CANTÓN)>, <PYTHON (SOFTWARE)>, <SISTEMA EXPERTO>.

Key words:

<TECHNOLOGY AND ENGINEERING SIENCE>, <SOFTWARE ENGINEERING>, <TOURISM>, <WEB APPLICATIONS PROGRAMMIG>, <MORONA (CANTÓN)>, <PYTHON (SOFTWARE)>, <EXPERT SYSTEM>.

1. INTRODUCCIÓN:

El desarrollo tecnológico ha evolucionado de manera acelerada en los últimos años. Y junto a este desarrollo, han aparecido paradigmas que buscan cambiar los métodos en que se realizan los procesos en una empresa.

En la actualidad, la oficina de Turismo del Gobierno Autónomo Descentralizado del Cantón Morona tiene como principal objetivo el promover la actividad turística en el cantón.

La recomendación de destinos turísticos, así como de restaurantes y centros de diversión se realiza de forma mayoritaria por parte de la ciudadanía del cantón. Se identificó que los turistas acuden a cualquier persona para solicitar que les recomienden algún sitio turístico, y estas lo hacen sin tener en cuenta las preferencias del turista; esto causa que la satisfacción del turista disminuya significativamente.

En conjunto con la experta humana en turismo, se definieron varias consideraciones que tener en cuenta para realizar la recomendación de destinos turísticos. Las mismas que son: Conformación del grupo de turistas, actividades que desean realizar, forma de transporte que desea utilizar y la cantidad de dinero que se está dispuesto a gastar.

2. MARCO TEÓRICO:

TURISMO

La definición del turismo ha causado controversias debido a las diversas interpretaciones que se le han presentado a este sector. Las discrepancias surgen por algunos conceptos es que disponen distintas disciplinas que ven en el turismo un campo de estudio y otras que se basan en puntos de vista de algunos tratadistas que pretenden explicar este fenómeno de la ideología.

Estas definiciones han evolucionado conforme la misma sociedad lo ha hecho para considerar aspectos que antes no se consideraron. Miguel Ángel Acerenza ha estudiado las diferentes conceptualizaciones del turismo armando una definición bastante completa:

“El turismo para la sociedad es, en esencia, una actividad relacionada con el descanso, el placer, el desarrollo cultural y la recreación. [...] el turismo para la sociedad constituye una forma particular del uso del tiempo libre y una forma particular de recreación, pero que no cubre todas las formas de uso que las personas pueden hacer de su tiempo libre, ni todas las formas posibles de recreación.” (Acerenza, 2006: p.27).

Para Guerrero y Ramos el turismo lo definen como la actividad que conjuga la información, acciones y desempeño constante de la industria del viaje, gobierno y recursos turísticos como se muestra en la figura 1.

SECTOR TURÍSTICO EN ECUADOR

La población del Ecuador es de aproximadamente catorce millones de habitantes y cuenta con una gran cantidad de recursos culturales y sociales en el interior de sus fronteras. Está habitado por un 65% de mestizos, 25% de indígenas, 10% de caucásicos y 7% de africanos.

Ecuador cuenta con una frontera cultural bien marcada dividida entre las tres regiones continentales: Sierra, Costa y Amazonía con una gran densidad poblacional en la Costa y la Sierra.

Tiene un calendario de eventos folclóricos y festividades religiosas tan diversas como su misma población. Cuenta con muchas festividades anuales correspondientes a la fundación e independencia de las localidades y también en honor a la Virgen y a los santos. Estas festividades se caracterizan por la fusión entre las tradiciones indígenas y católicas.

Cuenta con una historia de cuatro décadas de acciones de conservación de su naturaleza desde que en 1959 las Islas Galápagos como área protegida. En la actualidad, el Sistema Nacional de Áreas Protegidas tiene a su cargo 45 áreas protegidas que representan el 18.5% del territorio nacional con una superficie de aproximadamente 5 hectáreas.

La principal actividad predominante es la de Alimentos y Bebidas representando el 60% seguida por alojamiento (23%) y agencias de viaje (9%); el 8% de las demás actividades se reparte entre recreación, transporte y empresas de promoción turística.

Según el anuario estadístico del Ministerio de Turismo, en los periodos comprendidos desde el 2011 al 2015, el sector hotelero y alimentario de la provincia de Morona Santiago representa unos ingresos que oscilan apenas entre el 0.2% y el 0.36% respectivamente del total nacional. Lo cual se ve reflejado en que, en el final del periodo, el sector turístico de la provincia generó apenas 779 plazas de trabajo e ingresos económicos de 291.047 dólares.

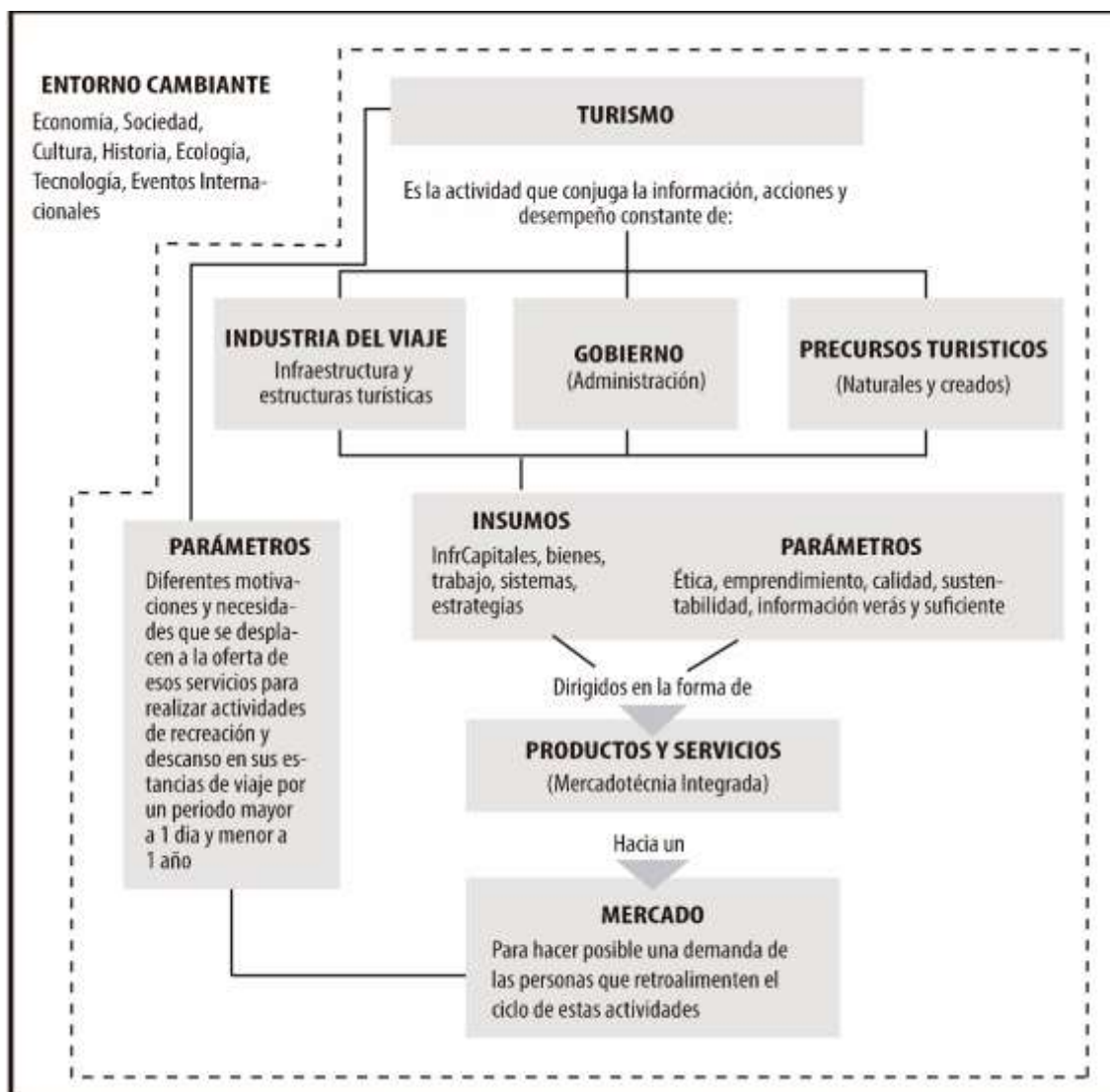


Figura 1. Gráfica de definición de turismo.

Fuente: Guerrero y Ramos, 2014: p.33

INTELIGENCIA ARTIFICIAL

En una primera aproximación podemos definir la Inteligencia Artificial como una ciencia que tiene como objetivo el diseño y construcción de máquinas capaces de imitar el comportamiento inteligente de las personas. Una rama especializada de la informática que investiga y produce razonamiento por medio de máquinas automáticas y que pretende fabricar artefactos dotados de la capacidad de pensar.

El nacimiento de la AI como disciplina de investigación se remonta a 1956, durante una conferencia de informática teórica que tuvo lugar en Dartmouth College (Estados Unidos). A dicha conferencia asistieron algunos de los científicos que se encargarían de desarrollar la disciplina en diferentes ámbitos y dotarla de una estructura teórica y computacional apropiada.

Según Turing (1950; citado en García, 2012: p.2) se debe considerar a una máquina como inteligente si esta es capaz de actuar como un humano. Para ello propone una prueba llamada Test de Turing que permite comprobar si una máquina es o no inteligente.

El Test de Turing aún es utilizado ya que exige lo que actualmente es conocido como inteligencia artificial: Reconocimiento del lenguaje natural, razonamiento, aprendizaje y representación del conocimiento.

SISTEMAS EXPERTOS

Un sistema experto(SE) se define como la aplicación que contiene el conocimiento de un especialista humano experimentado en un determinado campo de aplicación. Los SE pueden ser considerados como un subconjunto de la inteligencia artificial. Estos pueden funcionar mejor que cualquier experto humano tomando decisiones de forma individual en un área específica de pericia y están compuestos por cuatro componentes básicos: Subsistema de adquisición de conocimiento, base de conocimiento, base de hechos, el motor de inferencias y el subsistema de justificación, como se muestra en la figura 2.

Los SE tienen ventajas sobre los humanos. El conocimiento contenido en los SE es más fácil de documentar y de transferir que el de los expertos humanos. Dicho conocimiento es permanente en el tiempo.

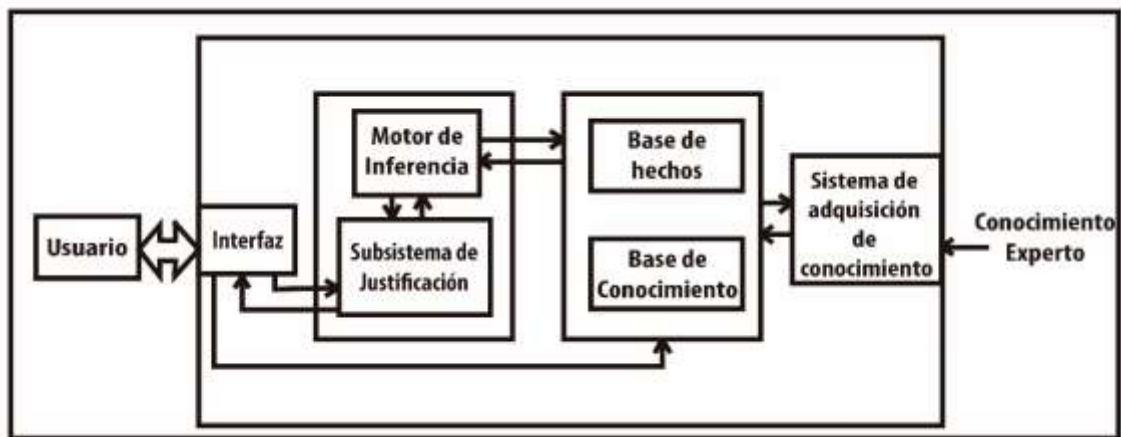


Figura 2. Estructura de un Sistema Experto

Fuente: Badaró et al, 2013: p.351

SISTEMAS EXPERTOS BASADOS EN LÓGICA DIFUSA

La lógica difusa (fuzzy logic) es considerada una generalización de la teoría general de conjuntos que permite que elementos de un universo tengan grados intermedios de pertenencia a conjunto por medio de una función característica.

Se debe hablar de la lógica difusa en plural, y son lógicas de valor múltiple que extiende de las lógicas clásicas. Mientras que en las lógicas clásicas se tiene valores de verdadero o falso, en las lógicas difusas se busca crear aproximaciones matemáticas que permitan resolver problemas. Lo difuso puede entenderse como la posibilidad de tener valores de verdad distintos a verdadero o falso.

SISTEMAS EXPERTOS BASADOS EN REDES NEURONALES

Las redes neuronales artificiales (RNA) son modelos informáticos que aparecieron como un intento de conseguir formalizaciones matemáticas acerca de la estructura del cerebro humano. Las RNA imitan la estructura física del sistema nervioso central, concentrándose en la estructura del cerebro, basado en el aprendizaje a través de la experiencia, obteniendo el conocimiento necesario a partir de estas.

La capacidad de aprendizaje adaptativo es una de las características más atractivas de redes neuronales. Esto es, aprenden a llevar a cabo ciertas tareas mediante un entrenamiento con ejemplos ilustrativos. Como las redes neuronales pueden aprender a diferenciar patrones mediante ejemplos y entrenamientos, no es necesario elaborar modelos a priori ni necesidad de especificar funciones de distribución de probabilidad. Las redes neuronales son sistemas dinámicos auto adaptativos. Son adaptables debido a la capacidad de autoajuste de los elementos procesales (neuronas) que componen el sistema. Son dinámicos, pues son capaces de estar constantemente cambiando para adaptarse a las nuevas condiciones.

En una RNA la unidad análoga a la neurona biológica es el elemento procesador que tiene varias entradas y las combina con una suma básica, como se muestra en la figura 3. Las entradas son modificadas por una función de transferencia y el valor de la salida al elemento procesador.

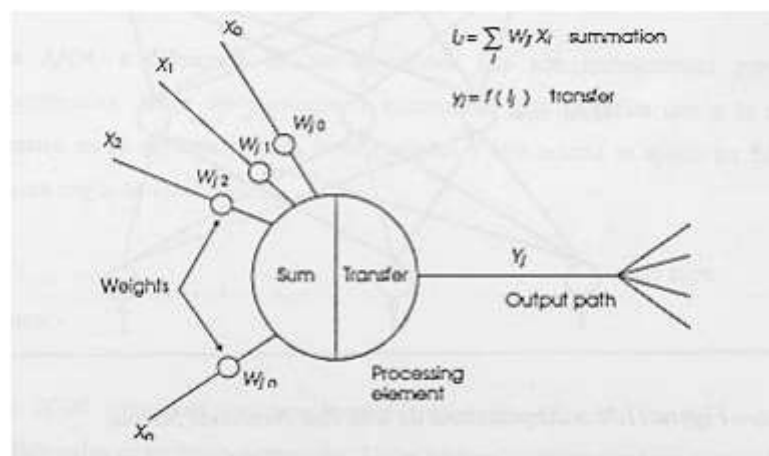
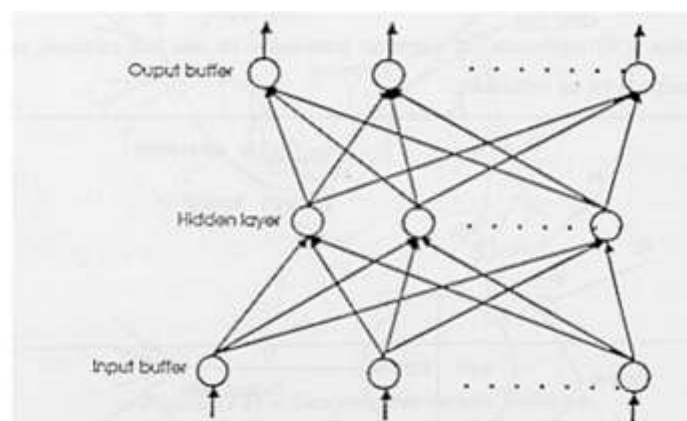


Figura 3. Diagrama de una Neurona Artificial (PE).
Fuente: Basogain, 2008: p.3

Una red neuronal consiste en un conjunto de unidades elementales conectadas de una forma concreta. Los elementos están organizados en grupos llamados niveles o capas. Una red típica consiste en una secuencia de capas con conexiones entre capas adyacentes consecutivas. Existen dos capas con conexiones con el mundo exterior. Una capa de entrada, buffer de entrada, donde se presentan los datos a la red, y una capa buffer de salida que mantiene la respuesta de la red a una entrada. El resto de las capas reciben el nombre de capas ocultas, como se muestra en la figura 4.



Figura

4. Arquitectura de una Red Neuronal Simple.
Fuente: Basogain, 2008: p.4

SISTEMAS EXPERTOS BASADOS EN REGLAS Y HECHOS

Los sistemas expertos basados en reglas y hechos tratan problemas deterministas, esto quiere decir que sacan sus conclusiones basándose en un conjunto de reglas utilizando un mecanismo de razonamiento lógico.

Los componentes principales de los sistemas expertos basados en reglas y hechos son los siguientes:

- *La Base de Conocimiento.*

Mientras el conocimiento es permanente, los datos no forman parte permanente de un sistema y son desechados luego de utilizarlos.

- *Subsistema de Adquisición de Conocimiento.*

Controla el flujo del nuevo conocimiento. El sistema determina qué nuevo conocimiento se necesita y, en caso necesario, incorpora estos conocimientos a la misma.

- *El Motor de Inferencia.*

Es el núcleo de un sistema experto. Su objetivo principal es sacar conclusiones aplicando el conocimiento a los datos.

En un sistema experto basado en reglas, el motor de inferencia se encarga de seleccionar de la base de conocimiento aquellas que son aplicables. Para ello, emplea la condición de las reglas en un conjunto de hechos (almacenados en la base de datos) y en caso de ser ciertos aplica las reglas obteniendo (infere) nuevos hechos que se incorporan a la base de datos. Repitiendo este proceso se produce un encadenamiento de conclusiones.

“El principal problema que se plantea es crear un conjunto de inferencias que nos permita llegar desde la definición inicial del problema a la solución. En este sentido podemos encontrar dos estrategias principales:

Avanzar desde el conjunto de datos o hechos hacia las conclusiones o razonamiento hacia delante. La regla de inferencia necesaria para realizar este tipo de razonamiento es el MODUS PONENS.

Seleccionar una posible conclusión e intentar demostrar su validez encontrando algunas evidencias que lo soporten o razonamiento hacia atrás. La regla de inferencia que se utiliza en este sentido es MODUS TOLLEN.”

Los valores de verdad de reglas y hechos son verdadero y falso, mientras que en sistemas que utilizan lógica difusa se hace uso valor de verdad que se encuentran entre el verdadero o falso. Mientras que las redes neuronales representan un modelo en el que las neuronas aprenden y se forman a sí mismas. Luego del aprendizaje, algunos sistemas representan una solución adecuada al problema mientras que otros no funcionan bien. Generalmente, el entrenamiento de una red neuronal necesita de grandes cantidades de datos y repeticiones y en la actualidad la oficina de turismo del GAD no cuenta con información actualizada ni precisa de todos los destinos turísticos del cantón.

Actualmente, el problema planteado por la unidad de turismo del GAD cantonal de Morona presentaba un modelo bastante parecido al de reglas hechos. Es decir, se planteaba la resolución de la recomendación utilizando preguntas enmarcadas en enunciados del tipo “Si... actividad, entonces... recomendar” y “Si... tipo de personas, entonces... recomendar”.

Por lo que, para el desarrollo de la aplicación, se eligió desarrollar el sistema experto usando un motor de inferencia basado en reglas y hechos con la regla MODUS PONENS ya que cumple con las especificaciones necesarias planteadas por la experta humana.

CALIDAD DE SOFTWARE

La calidad de uso se define como el grado en que un producto o sistema puede ser utilizado por usuarios específicos para satisfacer sus necesidades y así alcanzar objetivos específicos con eficacia, eficiencia, libertad de riesgo y satisfacción en contextos específicos de uso. Para cumplir con estas características utilizaremos el modelo de calidad de uso iso/iec 250101:2011

Estándar ISO/IEC 25010:2011

El estándar ISO/IEC 25010:2011 se define como:

- Un modelo de calidad de uso compuesto de 5 características (alguna de las cuales están subdivididas en respectivas subcaracterísticas) que se relacionan con el resultado de la interacción cuando un producto se utiliza en un contexto particular de uso. Este modelo de sistema es aplicable al sistema humano-ordenador completo, incluyendo tanto los sistemas informáticos en uso como los productos de software en uso.
- Un modelo de calidad del producto compuesto de ocho características (que se subdividen en subcaracterísticas) que se relacionan con las propiedades estáticas del software y las propiedades dinámicas del sistema informático. El modelo es aplicable tanto a sistemas informáticos como a productos de software.

Las características definidas por ambos modelos son relevantes para todos los productos de software y sistemas informáticos. Las características y subcaracterísticas proporcionan una terminología coherente para especificar, medir y evaluar la calidad del producto del sistema y del software. También proporcionan un conjunto de características de calidad contra las cuales se pueden comparar los requisitos de calidad establecidos para que sean completos. Aunque el alcance del modelo de calidad del producto está destinado a ser software y sistemas informáticos, muchas de las características también son relevantes para sistemas y servicios más amplios.

Dentro de las características y subcaracterísticas de calidad de uso tenemos:

- Efectividad
- Eficiencia
- Satisfacción
- Utilidad
- Confianza
- Complacencia
- Confortabilidad
- Libertad de riesgo
- Mitigación de riesgo económico
- Reducción de riesgo de seguridad
- Mitigación del riesgo ambiental
- Cobertura del contexto

- Completitud del contexto
- Flexibilidad

3. METODOLOGÍA:

El sistema fue desarrollado utilizando la metodología Microsoft Solution Framework ya que permite la gestión iterativa controlada por hitos. El ciclo de vida del software está orientado a la gestión temprana de riesgos utilizando pocos recursos humanos. Se definen las cinco etapas del ciclo de vida del software contempladas en la metodología.

3.1. Visión. Haciendo uso de técnicas como las encuestas, los cuestionarios y la lluvia de ideas con el personal técnico de la oficina de Turismo del GAD municipal de Morona, se plantean los requerimientos englobados en los siguientes módulos:

- **Módulo de gestión de usuarios.** La aplicación permite el registro público de cuentas de usuario, las mismas que únicamente podrán hacer uso de las funciones públicas de la aplicación (consulta de información y encuesta turística). De igual forma, permite al administrador validar usuarios que tienen la capacidad de crear un destino turístico y gestionar el conocimiento de ese destino.
- **Módulo de gestión de destinos turísticos.** Permite a un usuario validado (premium, desde ahora) crear y editar un destino turístico (nombre, ubicación, descripción) así como también los servicios vinculados al mismo (nombre, precio, descripción)
- **Módulo de conocimiento.** Es la parte medular del sistema, y permite crear y modificar el conocimiento de un destino turístico esto es el grupo de personas, los servicios disponibles, el tipo de comida, los medios de transporte y el dinero disponible para gastos. También permite que los usuarios realicen una encuesta con la finalidad de indicarle al sistema la conformación del grupo de turistas y otras preferencias para, en consecuencia, recibir una recomendación de un destino turístico.

3.2. Planeación. En esta etapa de ciclo de vida se traslada el diseño del sistema (y sus requerimientos) a un cronograma o detalle de actividades en dónde interviene todo el equipo del proyecto (desarrolladores y personal técnico turístico). También se definen los casos de uso críticos del sistema:

- **Gestión de usuarios**
- **Gestión de destinos turísticos**
- **Gestión del conocimiento**

3.3. Desarrollo. Posterior a la planeación, se definen las iteraciones del ciclo de vida en el que se definen los distintos entregables funcionales para cada iteración. En la primera iteración se definen y desarrollan los modelos correspondientes al MVT, así como también los primeros prototipos de los templates (Diseño de interfaces y Diseño estético). Se definen las pruebas funcionales automatizadas para los distintos requerimientos en la iteración correspondiente.

- **Estabilización.** En la etapa de estabilización se definen las pruebas unitarias y de integración del sistema. Al ejecutarse pruebas automatizadas de software, en esta etapa solamente se registra el acta de aceptación de la versión final de la aplicación.
- **Instalación o Entrega.** En la etapa de entrega del ciclo de vida se procede a entregar el software finalizado con el cumplimiento total de la funcionalidad del producto. Se

entregan también todos los documentos técnicos generados concernientes al software: Manual de Usuario, Manual Técnico, Manual de Instalación.

Se utilizó el lenguaje de programación python ya que este lenguaje se caracteriza por su velocidad de ejecución y flexibilidad. Como característica adicional, es un lenguaje multiplataforma.

Se utilizó el framework django que promueve el desarrollo de código limpio con buenas prácticas de desarrollo web, siguiendo el principio DRY (Don't Repeat Yourself).

Para la etapa de desarrollo de la aplicación web se tuvo en cuenta el patrón de diseño Modelo-Vista-Template otorgado por el framework django. Adicionalmente, el nombramiento y definición de variables, clases y métodos siguieron las recomendaciones planteadas por el lenguaje de programación Python.

Se detallan los componentes principales dentro del modelo-vista-template.

Modelos. Son nombrados como objetos que representan en singular. Comienza con mayúscula y las demás letras se escribirán en minúscula, figura 1.

```
class Center(models.Model):
    name = models.TextField(blank=False)
    address = map_fields.AddressField(max_length=200)
    additional_information = models.TextField(blank=True,
                                              verbose_name='Place Description')
    geolocation = map_fields.GeolocationField(max_length=100)
    slug = AutoSlugField(unique=True,
                        populate_from='name',
                        always_update=True)
    user = models.OneToOneField(User,
                                unique=True,
                                null=True,
                                related_name='center')
    free = models.BooleanField(default=False)
    service = models.ManyToManyField(Service,
                                    through='ServiceCenter')
```

Figura 1. Modelo Center
Realizado por: Autores. 2018

Vistas. Las vistas están todas basadas en clases para aprovechar las bondades del paradigma de la orientación a objetos. Los nombres representan la función principal de la vista, los nombres compuestos por más de dos palabras, estas se escriben juntas y comenzarán en mayúscula, figura 2.

```

class CenterCreateView(CreateView):
    model = Center
    template_name = 'center_create.html'
    form_class = CenterCreateForm
    context_object_name = 'center'

    @method_decorator(permission_required('usuario.add_center'))
    def dispatch(self, request, *args, **kwargs):
        if Center.objects.filter(user=request.user.profile).exists():
            return HttpResponseRedirect(
                reverse_lazy('Center:center_edit')
            )
        else:
            return super(CenterCreateView, self).dispatch(
                request, *args, **kwargs
            )

    def form_valid(self, form):
        form.instance.user = self.request.user.profile
        form.save()
        return super(CenterCreateView, self).form_valid(form)

```

Figura 2. Vista CenterCreateView
Realizado por: Autores. 2018

Templates: representa la capa de presentación del patrón de diseño, los nombres son referidos a la funcionalidad desempeñada por cada template dentro del sistema. Para realizar la automatización de la recomendación de destinos turísticos se desarrolló un método para inferir el conocimiento almacenado. El método desarrollado está aplicado a partir de los sistemas expertos basados en reglas y hechos.

Para crear el conocimiento correspondiente a cada centro se ejecuta la siguiente función, figura 3.

```

class CreateKnowledge(CreateView):
    model = Knowledge
    template_name = "knowledge_create.html"
    form_class = KnowledgeCreate

    def form_valid(self, form):
        user = Usuario.objects.get(user=self.request.user)
        center = Center.objects.get(user=user)
        form.instance.center = center
        form.save()
        return super(CreateKnowledge, self).form_valid(form)

    def get_success_url(self):
        return reverse('Center:center_detail', args=[self.request.user.profile.center.slug])

    @method_decorator(permission_required('usuario.add_center'))
    def dispatch(self, request, *args, **kwargs):
        if Knowledge.objects.filter(center=request.user.profile.center).exists():
            return HttpResponseRedirect(reverse_lazy('Center:knowledge_update'))
        else:
            return super(CreateKnowledge, self).dispatch(request, *args, **kwargs)

```

Figura 3 Código de la Función crear conocimiento
Realizado por: Autores. 2018

La Figura 3 muestra el código que se realiza en cada ejecución de la prueba. El método setUp configura la clase de la prueba, es el primer método que se ejecuta. El segundo y tercer método prueban, respectivamente, la funcionalidad de crear conocimiento cuando se ha creado un destino y cuando no. En el primer caso, se verifica con un estado 200 (Muestra correctamente la página esperada) y muestra el nombre del destino creado. El tercer método redirecciona a la página de creación de un destino (estado 302).

La Figura 4-4 muestra el código que se realizará en cada ejecución de la prueba. El método setUp configura la clase de la prueba, es el primer método que se ejecuta. El segundo método prueba que los usuarios autenticados puedan visualizar el formulario de la encuesta verificando que el estado de la petición sea correcto (código 200). El tercer método ejecuta una solicitud POST a la url, y verifica que obtiene una redirección (código 302) y posteriormente que se puede acceder a la url del resultado generado. El último método verifica que se obtiene una redirección a la página de login para los usuarios que no están autenticados.

Para inferir en el conocimiento se desarrolló el siguiente método basado en reglas y hechos.

```

@method_decorator(login_required, name='dispatch')
class PollForm(FormView):
    template_name = "poll_form_2.html"
    form_class = KnowledgePollForm
    success_url = reverse_lazy('Center:encuesta_resultado')

    def form_valid(self, form):
        form_groups = set(form.cleaned_data.get('group_type').values_list('name'))
        form_activities = set(form.cleaned_data.get('activity').values_list('name'))
        form_transport = set(form.cleaned_data.get('transport').values_list('name'))
        form_food = set(form.cleaned_data.get('food').values_list('name'))
        form_money_per_person = form.cleaned_data.get('money_per_people')

        centers = Knowledge.objects.all()
        centervalue = list()

        for c in centers:
            value = 0
            center = c.center
            groups = set(c.group_type.all().values_list('name'))
            activities = set(c.activities.all().values_list('name'))
            transport = set(c.transport.all().values_list('name'))
            food = set(c.food.all().values_list('name'))
            money_per_person = c.money_per_person
            temp_groups = groups & form_groups
            value += len(temp_groups)
            temp_activities = activities & form_activities
            value += len(temp_activities)
            temp_transport = transport & form_transport
            value += len(temp_transport)
            temp_food = food & form_food
            value += len(temp_food)
            if money_per_person <= form_money_per_person:
                value += 1
            centervalue.append((center, value))

        centervalue.sort(key=lambda x: x[1])
        selectedcenter = centervalue.pop()
        poll = Poll()
        poll.center = selectedcenter[0]
        poll.value = selectedcenter[1]

        poll.user = Usuario.objects.get(
            user=self.request.user
        )
        poll.save()
        return HttpResponseRedirect(
            reverse('Center:encuesta_resultado',
                args=[poll.id]
            )
        )

```

Figura 4-4. Código de la Función realizar encuesta.
Realizado por: Autores. 2018

4. RESULTADOS:

Para verificar la influencia del sistema en la exactitud de las inferencias, se realizaron treinta pruebas de recomendación de destinos turísticos sin utilizar el sistema, y posteriormente la misma cantidad de pruebas utilizando el sistema; en ambos casos, las recomendaciones fueron analizadas por la experta en turismo del GAD del Cantón Morona.

Resultados de exactitud utilizando el método de recomendación verbal.

Actualmente, la recomendación de destinos turísticos la realiza la ciudadanía teniendo en cuenta únicamente sus propias preferencias o experiencias; sin considerar en ningún momento los gustos o deseos que en ese momento tiene el turista.

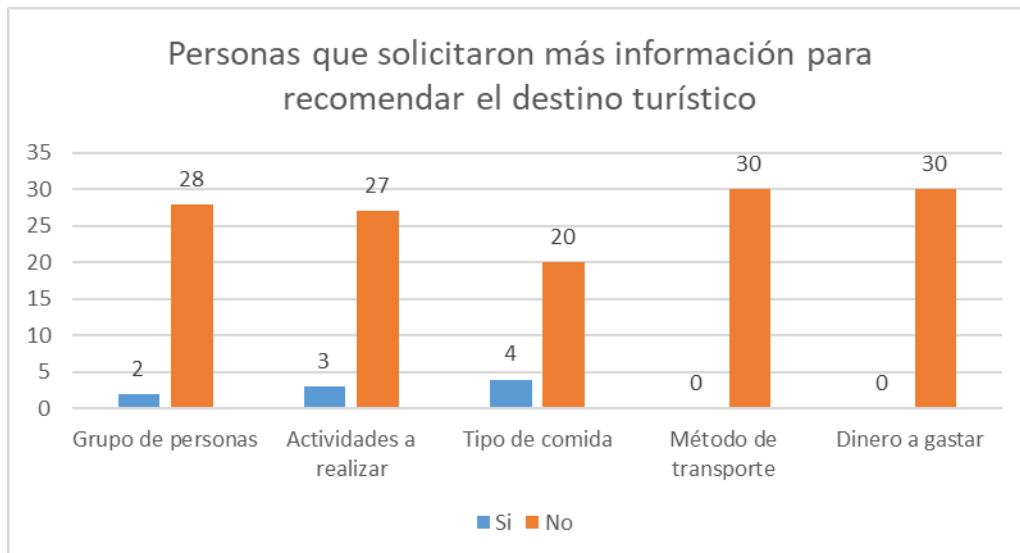


Gráfico 1. Personas que solicitaron información para recomendar el destino turístico
Realizado por: Autores. 2018

Para realizar estas pruebas, se solicitó a treinta personas distintas en las calles céntricas de la ciudad de Macas que recomienden un lugar turístico; en la mayoría de los casos, las personas no solicitaron información detallada respecto a lo que se buscaba del destino turístico, sino que se limitaron a realizar una recomendación subjetiva de acuerdo con sus propias experiencias y a la poca información solicitada. Como muestra el Gráfico 1, solamente 2 personas preguntaron acerca del tipo de personas que visitarían el destino turístico, 3 sobre las actividades que deseaban realizar, 4 sobre el tipo de comida y ninguna acerca del método de transporte y el dinero que está dispuesto a gastar.

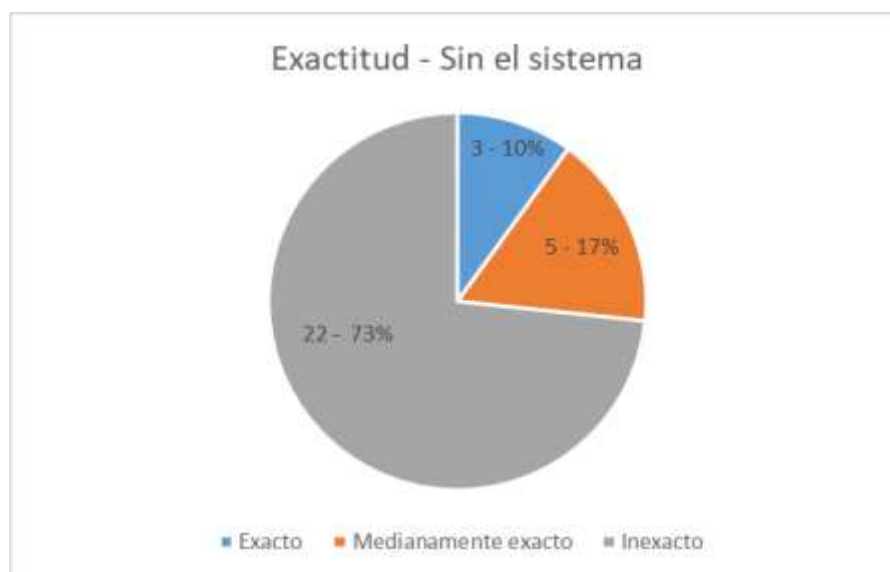


Gráfico 2. Aciertos en las recomendaciones por parte de los ciudadanos.

Realizado por: Autores. 2018

Posteriormente, el experto humano en turismo analiza las recomendaciones teniendo en cuenta las preferencias del turista. En el Gráfico 2 se puede observar que solamente en 3 (10%) casos la recomendación es la más adecuada, en 5 (17%) casos la recomendación no es la más adecuada, sin embargo, es aceptable. Y en 22 (73%) casos no se recomendó la opción más adecuada. Lo que significa que únicamente en el 27% de los casos se recomendó la opción correcta al turista.

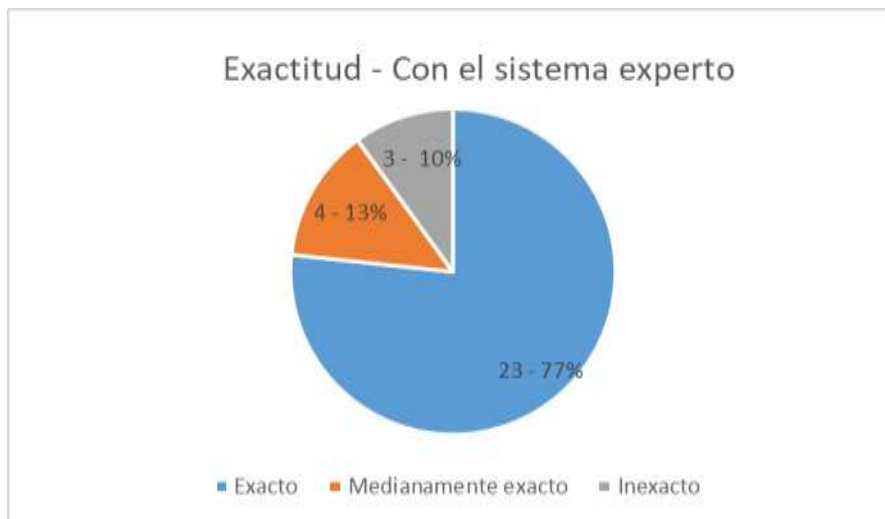


Gráfico 3. Aciertos en las recomendaciones utilizando el sistema.
Realizado por: Autores. 2018

El nivel de exactitud del sistema es realizado de igual forma por el experto humano (Anexo 1), los resultados se detallan en el Gráfico 3 obteniendo como resultado que en el 77% de los casos el sistema brindó la recomendación exacta, que coincide con lo que hubiese recomendado el experto humano, en un 13% la recomendación es aceptable y cumple con las expectativas del turista, sin embargo, el experto habría recomendado otro lugar más adecuado. Y apenas en el 10% de los casos el sistema recomendó un lugar inadecuado teniendo en cuenta todas las preferencias del turista. Por lo que, en el 90% de los casos, el sistema brinda una recomendación adecuada; una mejora del 63% de precisión.

Métricas de calidad del software

Las métricas tratan de servir de medio para entender, monitorizar, controlar, predecir y probar el desarrollo de software y de los proyectos de mantenimiento (García, 2010. pp. 7-8). La evaluación de la calidad de la aplicación será realizada según las métricas y sub-métricas de Usabilidad y Funcionalidad del estándar ISO/IEC 9126-1 utilizando los siguientes indicadores.

- **Excede los requisitos.** Las funciones del sistema cumplen los requisitos del usuario y exceden sus expectativas positivamente.
- **Cumple los requisitos.** Cuando las funcionalidades del sistema cumplen los requisitos del sistema definidos por el cliente.
- **Mínimamente aceptable.** Cuando las funcionalidades del sistema cumplen los requisitos, pero no las necesidades reales del sistema.

- **No aceptable.** Cuando las funcionalidades del sistema no cumplen con todos los requisitos establecidos.

Tabla 1. Métrica de Usabilidad

MÉTRICA DE USABILIDAD				
SUB-MÉTRICAS	INDICADORES			
	Excede los requisitos	Cumple los requisitos	Mínimamente aceptables	No aceptable
Capacidad para ser entendido	2	11	2	0
Capacidad para ser aprendido	4	9	2	0
Capacidad para ser operado	6	9	0	0
Capacidad de atracción	7	6	2	0
Cumplimiento de usabilidad	8	7	0	0
TOTAL	27	42	6	0
PORCENTAJES	36%	56%	8%	0%

Realizado por: Autores. 2018

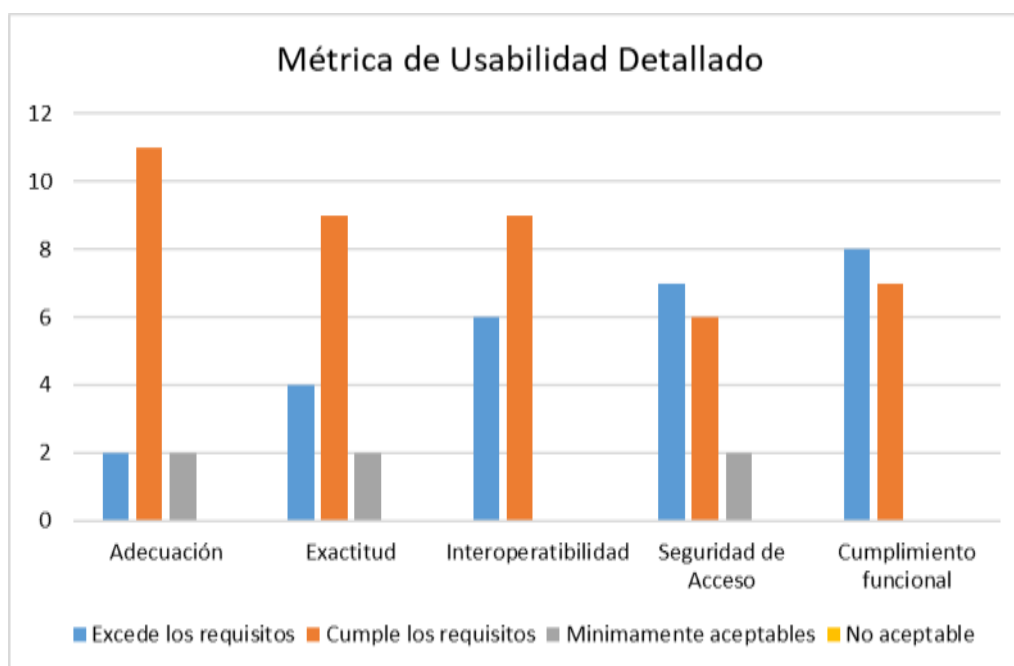


Gráfico 4. Tabulación de Resultados: Métrica de Usabilidad Detallado

Realizado por: Autores. 2018

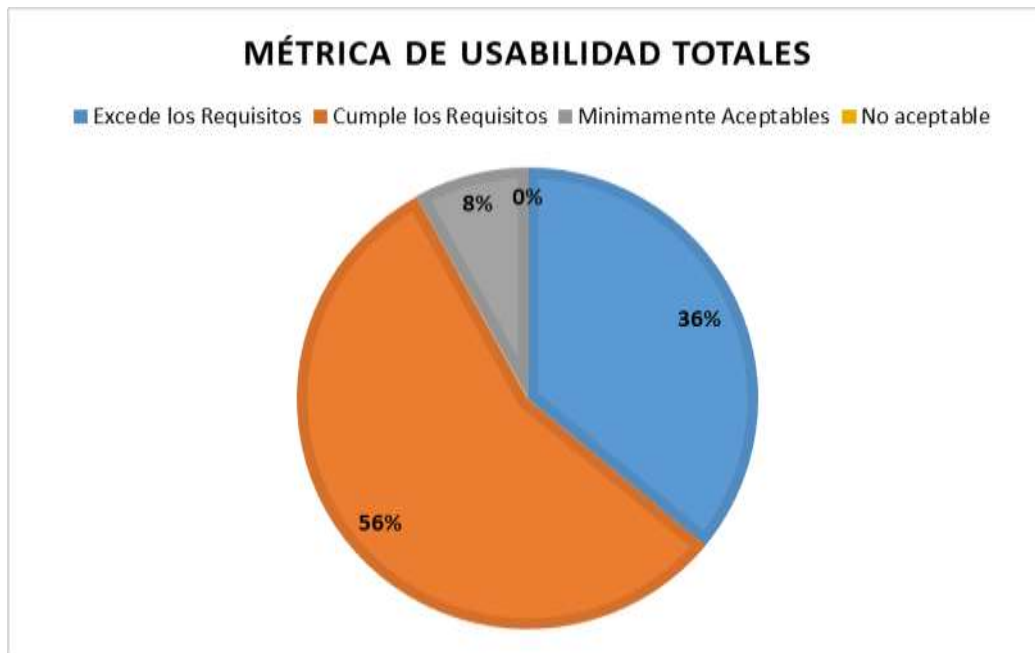


Gráfico 5. Tabulación de Resultados: Métrica de Usabilidad Detallado
Realizado por: Autores. 2018

El gráfico 5 muestra la tabulación de los datos de la aplicación, en la que se puede observar que la aplicación “cumple con los requisitos” en un 56%, “excede los requisitos” en un 36%, y cumple de manera “mínimamente aceptable” en un 8% respecto a la métrica de usabilidad de acuerdo con las encuestas realizadas.

5. CONCLUSIONES

Tras realizar las pruebas del sistema, se concluye que el desarrollo del sistema experto influyó positivamente en la exactitud de la recomendación de destinos turísticos en el cantón Morona en un 63%.

En la actualidad la recomendación de destinos turísticos se realiza utilizando la experiencia o preferencias de la persona que realiza dicha recomendación, es decir, los habitantes del cantón a quienes el turista solicita la información acerca de los destinos turísticos; al no tener en cuenta las preferencias del turista, la exactitud de las recomendaciones es de apenas un 27%.

Una vez realizada la investigación acerca de sistemas expertos se concluye que, un sistema experto tiene la capacidad de simular el comportamiento humano en un área de trabajo determinada, que se lo realiza, utilizando el conocimiento del experto en combinación de tecnologías de desarrollo de sistemas inteligentes.

Luego de desarrollar el sistema experto se concluyó que, es posible utilizar un sistema inteligente basado en reglas y hechos que simule el conocimiento de un experto humano para realizar la recomendación de destinos turísticos.

Después de realizadas las pruebas de inferencia, y tras la validación de los resultados por parte de la experta humana, se concluye que al utilizar el sistema se obtiene una precisión del 90% en las recomendaciones turísticas.

6. REFERENCIAS

ACERENZA, Miguel. *Conceptualización, Origen y Evolución del Turismo*. México-México: Trillas, 2006, p 27.

ÁLVAREZ, Luis. *Fundamentos de inteligencia artificial*. Murcia-España: Universidad de Murcia. Secretariado de Publicaciones, 1994, p 19.

BASOGAIN, Xavier. *Redes neuronales artificiales y sus aplicaciones*. Bilbao-España, Escuela Superior de Ingeniería de Bilbao, 2008, pp. 3-4.

BENITEZ, Raúl; et al. *INTELIGENCIA ARTIFICIAL AVANZADA*. Barcelona-España: Editorial UOC, 2013, p 6.

BUCHANAN, Bruce; y SHORTLIFFE, Edward. *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. MA-Estados Unidos: Addison, Wesley, 1984, p 3.

CASTILLO, Enrique; et al. *Sistemas Expertos y Modelos de redes probabilísticas*. Santander-España: Academia de ingeniería, Universidad de Cantabria, 1996, p.11, p.12.

CONDORI, José. "Phython – Django. Framework de desarrollo web para perfeccionistas. Basado en el Modelo MTV" [en línea]. *Revista de Información, Tecnología y Sociedad*, n°7 (2012), (Bolivia) pp. 36-37. Disponible en: http://www.revistasbolivianas.org.bo/scielo.php?script=sci_arttext&pid=S1997-40442012000200016&lng=es&nrm=iso

DABARÓ, Sebastián; et al. "Sistemas Expertos: Fundamentos, Metodologías y Aplicaciones" *Revista Ciencia y Tecnología de la Universidad de Palermo* [en línea]. 2013, (Argentina) vol.13, pp.349-364. [Consulta: 10 marzo 2018]. ISSN 1850-0870. Disponible en: http://www.palermo.edu/ingenieria/pdf2014/13/CyT_13_24.pdf

DRAKE, Fred. *El tutorial de Python* [en línea]. Argentina, Python Software Foundation: 2017, pp.1-3-8-9-11-12-13-66. [Consulta: 10 enero 2017] Disponible en: <http://docs.python.org.ar/tutorial/pdfs/TutorialPython3.pdf>

FLÓREZ, Raquel y FERNÁNDEZ, José. *Las redes neuronales artificiales fundamentos teóricos y aplicaciones prácticas*. La Coruña-España: Netbiblo S.L., 2008, p.16

FUENTES, Aníbal. *Teoría y Desarrollo del Turismo en el ECUADOR*. Quito-Ecuador: Efecto Gráfico, 2013, pp.121-122-124

GÁMEZ, José y PUERTA, José. *Sistemas Expertos probabilísticas*. Castilla-España: Ediciones de la universidad de Castilla la Mancha, 1998, p.6, p.7.

GARCÍA, Alberto. *INTELIGENCIA ARTIFICIAL: Fundamentos, práctica y aplicaciones*. Madrid-España: Grupo RC, 2012, pp.2-6

GUERRERO, Perla; & RAMOS, José. *Introducción al Turismo* [en línea]. México-México: Patria, 2014, p.33. [Consulta: 30 noviembre 2016]. Disponible en: <http://www.editorialpatria.com.mx/pdf/files/9786074384130.pdf>

INFANTE, Sergio. *Maestros del Web* [en línea]. 2012. Curso Django: Entendiendo como Funciona Django. [Consulta: 14 febrero 2017] Disponible en: <http://www.maestrosdelweb.com/curso-django-entendiendo-como-trabaja-django>

MATICH, Jorge. *Redes neuronales: conceptos básicos y aplicaciones*. Rosario-Argentina: Universidad Tecnológica Nacional, 2001, p.9.

MORALES-LUNA, Guillermo. *Introducción a la lógica difusa*. México-México: Centro de Investigación y Estudios Avanzados del IPN, 2002, p.1

PÉREZ, Iván. *Lógica difusa para principiantes*. Caracas-Venezuela: Universidad Católica Andrés Bello, 2007, p.12.

PERUREN, Lilliam y MORAGUE, Mercedes. "Usability of Web sites, methods and evaluation techniques" *Rev. cuba. inf. cienc. salud* [en línea]. 2013, (Cuba) vol.24, n.2, p.188. [Consulta: 18

marzo 2018]. ISSN 2307-2113. Disponible en: http://scielo.sld.cu/scielo.php?pid=S2307-21132013000200007&script=sci_arttext&tlng=en

PINO, Raúl; et al. *Introducción a la inteligencia artificial: Sistemas Expertos, Redes neuronales artificiales y computación evolutiva*. Madrid-España: Servicio de publicaciones de la Universidad de Oviedo, 2001, p.1.