



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA INGENIERÍA EN SISTEMAS INFORMÁTICOS

DESARROLLO DE UN SISTEMA DE RECOMENDACIÓN DE
CONTENIDOS EDUCATIVOS BASADO EN ESTILOS DE
APRENDIZAJE APLICADO A LA ESCUELA DE INGENIERÍA EN
SISTEMAS

Trabajo De Integración Curricular

Tipo: Proyecto Técnico

Presentado para optar al grado académico de:

INGENIERO EN SISTEMAS INFORMÁTICOS

AUTORES: DAYANA ELIZABETH ALMACHE LOPEZ

KEVIN PAÚL JÁCOME NIAMA

DIRECTOR: ING. DANILO MAURICIO PASTOR RAMIREZ

Riobamba - Ecuador

2022

©2022, Dayana Elizabeth Almache Lopez, Kevin Paúl Jácome Niama

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

Nosotros, **Dayana Elizabeth Almache Lopez** y **Kevin Paúl Jácome Niama**, declaramos que el presente trabajo de integración curricular es de nuestra autoría y los resultados del mismo son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autores asumimos la responsabilidad legal y académica de los contenidos de este trabajo **de titulación**; el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 21 de enero de 2022

Dayana Elizabeth Almache Lopez
1804039251

Kevin Paúl Jácome Niama
0604213637

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA INGENIERÍA EN SISTEMAS INFORMÁTICOS

El Tribunal de Integración Curricular certifica que: El trabajo de integración curricular; tipo: Proyecto Técnico, **“DESARROLLO DE UN SISTEMA DE RECOMENDACIÓN DE CONTENIDOS EDUCATIVOS BASADO EN ESTILOS DE APRENDIZAJE APLICADO A LA ESCUELA DE INGENIERÍA EN SISTEMAS”**, realizado por los señores: **DAYANA ELIZABETH ALMACHE LOPEZ** y **KEVIN PAÚL JÁCOME NIAMA**, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Integración Curricular, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal autoriza su presentación.

FIRMAS

FECHA

Dra. Gloria de Lourdes Arcos
PRESIDENTE DEL TRIBUNAL

Ing. Danilo Mauricio Pastor
**DIRECTOR DEL TRABAJO DE
INTEGRACIÓN CURRICULAR**

Ing. Diego Fernando Ávila
MIEMBRO DEL TRIBUNAL

DEDICATORIA

A Dios por haberme bendecido en cada paso que di, por brindarme la salud y fuerza para cumplir esta meta. A mi madre por ser mi motivación constante el pilar fundamental para culminar esta etapa, por enseñarme valores, principios y apoyarme siempre, este logro es de las dos. A mi padre en el cielo por ser mi ángel guardián. A mis hermanos por la ayuda que me brindaron. A mi compañero y amigo Kevin Jácome por los años compartidos, por el esfuerzo y dedicación.

Dayana

Dedico este trabajo de titulación a mis amados padres Paulina Niama y Efraín Jácome, por su apoyo incondicional durante todo el trayecto de mi vida estudiantil. A mis hermanas Ariana y Valentina por su compañía y ánimos a seguir adelante. A mis amigos y compañeros tanto dentro de la institución como fuera de ella, con quienes pase momentos inolvidables.

Kevin

AGRADECIMIENTO

Un agradecimiento eterno a nuestra alma mater por permitir, formarnos en sus aulas, cumplir nuestras metas y sueños académicos. Un cordial reconocimiento y agradecimiento a todos nuestros docentes de la Escuela de Ingeniería en Sistemas, por haberme brindado sus conocimientos y experiencia para formarnos como buenos profesionales. A nuestro director y miembro de tesis, Ing. Danilo Pastor e Ing. Diego Ávila por su contribución, orientación y apoyo en la realización del presente trabajo de titulación.

Dayana & Kevin

TABLA DE CONTENIDO

ÍNDICE DE TABLAS.....	x
ÍNDICE DE FIGURAS.....	xii
ÍNDICE DE GRÁFICOS.....	xiii
ÍNDICE DE ABREVIATURAS.....	xiv
ÍNDICE DE ANEXOS	xv
RESUMEN.....	xvi
ABSTRACT	xvii
INTRODUCCIÓN	xviii

CAPITULO I

1. DIAGNÓSTICO DEL PROBLEMA	1
1.1. Antecedentes	1
1.2. Formulación del problema	2
1.3. Sistematización del problema.....	2
1.4. Justificación	3
1.4.1. <i>Justificación teórica</i>	3
1.4.2. <i>Justificación aplicada</i>	3
1.5. Objetivos	5
1.5.1. <i>Objetivo General</i>	5
1.5.2. <i>Objetivos Específicos</i>	5

CAPITULO II

2. REVISIÓN DE LA LITERATURA O FUNDAMENTOS TEÓRICOS	6
2.1. Trabajos Relacionados.....	6
2.2. Aprendizaje.....	9
2.2.1. <i>Aprendizaje experiencial</i>	9
2.3. Estilos de aprendizaje	9
2.4. Modelos de aprendizaje	10
2.5. Clasificación de los modelos de aprendizaje por autor.....	10
2.5.1. <i>Comparativa de modelos de aprendizaje</i>	12
2.5.2. <i>Análisis de estilos de aprendizaje del modelo experimental de David Kolb</i>	13

2.6.	Rendimiento académico.....	15
2.6.1.	<i>Como medir el rendimiento académico.....</i>	15
2.7.	Metodología de desarrollo	15
2.7.1.	<i>Metodologías ágiles</i>	16
2.7.2.	<i>Metodología XP</i>	17
2.8.	Arquitectura de software.....	18
2.8.1.	<i>Patrón modelo-vista-controlador</i>	19
2.9.	Lenguajes de programación	19
2.9.1.	<i>Comparativa de lenguajes de programación</i>	19
2.10.	Herramientas de desarrollo.....	20
2.10.1.	<i>Java</i>	20
2.10.2.	<i>JavaScript.....</i>	21
2.10.3.	<i>JQuery.....</i>	22
2.10.4.	<i>HTML.....</i>	22
2.10.5.	<i>PostgreSQL</i>	23
2.10.6.	<i>CSS.....</i>	23
2.10.7.	<i>NetBeans</i>	24
2.10.8.	<i>CLisp</i>	24
2.10.9.	<i>Servicio web</i>	24
2.10.10.	<i>Frameworks front-end.....</i>	26
2.11.	Sistema recomendador.....	30
2.11.1.	<i>Características de los sistemas recomendadores</i>	30
2.11.2.	<i>Beneficio de los sistemas recomendadores</i>	30
2.11.3.	<i>Funcionamiento de los sistemas recomendadores.....</i>	31
2.11.4.	<i>Clasificación de los sistemas recomendadores</i>	31
2.11.5.	<i>Arquitectura de un Sistema Recomendador</i>	33

CAPITULO III

3.	MARCO METODOLÓGICO	34
3.1.	Tipo de estudio	34
3.2.	Métodos y técnicas de investigación	34
3.3.	Etapas del desarrollo del trabajo.....	35
3.3.1.	<i>Etapa 1: Determinar la preferencia de contenidos.....</i>	36
3.3.2.	<i>Etapa 2: Desarrollo del sistema recomendador.....</i>	38
3.3.3.	<i>Proceso para evaluar el rendimiento académico.....</i>	77

CAPITULO IV

4.	RESULTADOS	80
4.1.	Resultados de la etapa 1: Determinar la preferencia de contenidos.....	80
<i>4.1.1.</i>	<i>Análisis de estilos de aprendizaje.....</i>	<i>80</i>
<i>4.1.2.</i>	<i>Análisis de preferencia de contenido</i>	<i>81</i>
4.2.	Resultados del estudio de evaluación del rendimiento académico.....	85
<i>4.2.1.</i>	<i>Análisis descriptivo</i>	<i>85</i>
<i>4.2.2.</i>	<i>Análisis inferencial.....</i>	<i>87</i>
	CONCLUSIONES.....	90
	RECOMENDACIONES.....	92
	GLOSARIO	
	BIBLIOGRAFÍA	
	ANEXOS	

ÍNDICE DE TABLAS

Tabla 1-2: Sistemas de personalización e-learning con su parámetro de personalización.....	6
Tabla 2-2: Modelos de estilos de aprendizaje.....	10
Tabla 3-2: Tabla comparativa de modelos de aprendizaje.....	12
Tabla 4-2: Estilos de aprendizaje de Kolb	14
Tabla 5-2: Relación entre estilo de aprendizaje con formato de contenido educativo.....	15
Tabla 6-2: Comparativa metodologías de desarrollo tradicional-ágil.....	16
Tabla 7-2: Comparativa de metodologías ágiles XP-SCRUM	16
Tabla 8-2: Descripción de los roles establecidos en la metodología XP	17
Tabla 9-2: Comparativa de lenguajes de programación Java, Python y PHP.	20
Tabla 10-2: Componentes de Java	21
Tabla 11-2: Ventajas y desventajas del servicio web RESTful	25
Tabla 12-2: Comparativa de frameworks front-end.....	26
Tabla 13-3: Métodos y técnicas	34
Tabla 14-3: Análisis demográfico de los datos para estudio de estilo de aprendizaje.	37
Tabla 15-3: Roles del proyecto	39
Tabla 16-3: Tipos de usuarios.....	43
Tabla 17-3: Metáforas del sistema.....	45
Tabla 18-3: Historias del sistema.....	46
Tabla 19-3: Método t-shirt.....	47
Tabla 20-3: Priorización y estimación de esfuerzo de MS e HU del sistema	48
Tabla 21-3: Parejas asignadas.....	49
Tabla 22-3: Planificación de actividades del proyecto	50
Tabla 23-3: Formato de ficha historia de usuario	52
Tabla 24-3: Documentación de caso de uso “autenticación del administrador”.....	54
Tabla 25-3: Metáfora “Analizar y diseñar la arquitectura del sistema”.....	56
Tabla 26-3: Tarea de ingeniería de la MS_06.....	56
Tabla 27-3: Metáfora “Investigar y analizar el estándar de codificación”.....	58
Tabla 28-3: Diccionario de datos de la tabla usuario.....	59
Tabla 29-3: Metáfora “Análisis y diseño de la base de datos”.....	60
Tabla 30-3: Descripción de estilo de las secciones de la interfaz	61
Tabla 31-3: Metáfora “Diseño de la interfaz de usuario”	63
Tabla 32-3: Identificación y categorización de riesgos.....	64
Tabla 33-3: Actividades de la iteración 1	66

Tabla 34-3: Metáfora desarrollada en la iteración 1	66
Tabla 35-3: Actividades de la iteración 2	67
Tabla 36-3: Actividades de la iteración 3	67
Tabla 37-3: Actividades de la iteración 4	68
Tabla 38-3: Actividades de la iteración 5	68
Tabla 39-3: Actividades de la iteración 6	69
Tabla 40-3: Actividades de la iteración 7	69
Tabla 41-3: Actividades de la iteración 8	69
Tabla 42-3: Actividades de la iteración 9	70
Tabla 43-3: Actividades de la iteración 10	70
Tabla 44-3: Actividades de la iteración 11	71
Tabla 45-3: Actividades de la iteración 12	72
Tabla 46-3: Formato de tarjeta de prueba de aceptación	72
Tabla 47-3: Numero de pruebas de aceptación por HU/MS	73
Tabla 48-3: Metáfora “Documentar el manual de usuario”	75
Tabla 49-4: Análisis de los resultados de estilo de aprendizaje.....	80
Tabla 50-4: Resultados de preferencia de formato de contenido por cada estilo de aprendizaje.....	81
Tabla 51-4: Resultados de preferencia a sección de preguntas adicional al contenido.....	83
Tabla 52-4: Resultados de preferencia entre imágenes estáticas o animaciones	84
Tabla 53-4: Relación entre estilo de aprendizaje con formato de contenido educativo.....	85
Tabla 54-4: Resultados del rendimiento académico	85
Tabla 55-4: Análisis de los resultados para evaluar el rendimiento académico.....	86
Tabla 56-4: Prueba de normalidad de Shapiro-Wilk.....	88
Tabla 57-4: Estadísticas de grupo de los datos analizados	88
Tabla 58-4: Análisis estadístico t-student para muestras independientes	89

ÍNDICE DE FIGURAS

Figura 1-1:	Diagrama de fases de la metodología XP	18
Figura 2-2:	Implementación de JavaServer Faces de MVC.....	28
Figura 3-2:	Ciclo de vida de JSF	29
Figura 4-2:	Esquema de funcionamiento de un sistema recomendador.....	31
Figura 5-2:	Sistema recomendador basado en contenido	32
Figura 6-2:	Sistema recomendador colaborativo	32
Figura 7-2:	Sistema recomendador colaborativo	33
Figura 8-2:	Arquitectura del SR.....	33
Figura 9-3:	Herramienta online para estudio preliminar.....	38
Figura 10-3:	Diagrama de secuencia “Obtener estilo de aprendizaje del estudiante”	40
Figura 11-3:	Diagrama de secuencia “Creación de contenido”	41
Figura 12-3:	Diagrama de secuencia “Recomendar contenido”	42
Figura 13-3:	Mapa conceptual de funcionalidades del sistema	45
Figura 14-3:	Diagrama de casos de uso	54
Figura 15-3:	Diagrama de arquitectura del sistema	55
Figura 16-3:	Esquema de la base de datos del sistema	59
Figura 17-3:	Diseño de la interfaz de usuario	61
Figura 18-3:	Bosquejo de la interfaz de usuario	62
Figura 19-3:	Diagrama de despliegue.....	75
Figura 20-3:	Sistema “Custom Learning”	78
Figura 21-3:	Evaluación de rendimiento académico online.....	79

ÍNDICE DE GRÁFICOS

Gráfico 1-3: Velocidad del proyecto.....	76
Gráfico 2-3: Gráfica de análisis de los resultados de estilo de aprendizaje	81
Gráfico 3-4: Gráfica de preferencia de formato de contenido por cada estilo de aprendizaje ...	82
Gráfico 4-4: Gráfica de preferencia a sección de preguntas	83
Gráfico 5-4: Gráfica de preferencia entre imágenes estáticas o animaciones.....	84
Gráfico 6-4: Análisis descriptivo de los datos	87

ÍNDICE DE ABREVIATURAS

IDE	Development Environment (Entorno de Desarrollo Integrado)
HTML	HyperText Markup Language (Lenguaje de Marcas de Hipertexto)
TICs	Tecnologías de la Información y la Comunicación
XP	Extreme Programming (Programación extrema)
MVC	Modelo Vista Controlador
JVM	Máquina virtual de java
CSS	Cascading Style Sheets (Hojas de estilo)
IA	Inteligencia Artificial
XML	Extensible Markup Language (Lenguaje de Mercado Extensible)
URI	Identificador uniforme de recursos
JSF	JavaServerFaces
SR	Sistema Recomendador
UML	Lenguaje unificado de modelado
UAN	Unidad de Admisión y Nivelación
SRC	Sistema Recomendador de Contenido
MS	Metáforas del Sistema
HU	Historias de Usuario
PA	Prueba de Aceptación
Iter.	Iteración
Esf.	Esfuerzo
PLCP	Planificador de Cursos de Aprendizaje Personalizado
OA	Objeto de Aprendizaje

ÍNDICE DE ANEXOS

- ANEXO A:** Gestión de riesgos
- ANEXO B:** Casos de uso
- ANEXO C:** Historias de usuario
- ANEXO D:** Modelo de la base de datos
- ANEXO E:** Diccionario de datos
- ANEXO F:** Test de estilos de aprendizaje propuesto por David Kolb
- ANEXO G:** Test de preferencia de contenidos
- ANEXO H:** Evaluación para determinar el rendimiento académico de los estudiantes
- ANEXO I:** Manual de usuario

RESUMEN

El objetivo del presente trabajo fue desarrollar un sistema recomendador de contenidos educativos basado en estilos de aprendizaje que permita gestionar y recomendar contenidos a estudiantes con la finalidad de mejorar su rendimiento académico. Su desarrollo contempló dos estudios, un estudio preliminar que estableció las preferencias de contenido con respecto a estilos de aprendizaje, y otro estudio para evaluar el rendimiento académico de los estudiantes que emplearon el sistema. Como técnicas de recolección de datos se utilizaron: revisión de documentación, test de estilos de aprendizaje de David Kolb, test para determinar preferencia de contenidos y una evaluación estructurada con preguntas de selección múltiple. La población para los estudios fueron estudiantes de ingeniería en software de la Escuela Superior Politécnica de Chimborazo, siendo la muestra para el primer estudio 128 estudiantes, y para el segundo se tomó a dos grupos de estudiantes determinados como grupo de control y grupo experimental respectivamente, para aplicar y comparar sus resultados al interactuar con el sistema y su contenido. Se realizó el desarrollo del sistema recomendador de contenidos con el lenguaje de programación java, siguiendo la metodología XP, haciendo uso de herramientas y tecnologías como: IDE NetBeans, javascript, Ajax, PrimeFaces y PostgreSQL. Los resultados obtenidos sobre preferencia de formato de contenido en relación al estilo de aprendizaje mostraron que los grupos convergente/asimilador prefieren el formato video, mientras que los grupos divergente/acomodador el formato simulación. La evaluación del rendimiento académico encontró que el grupo experimental obtuvo un promedio de 15.60/20, mientras que el grupo de control 12.74/20. Aplicando la prueba t-student se determinó que existe una diferencia significativa entre las medias de los datos. Se concluye que el uso del sistema recomendador mejoró el rendimiento de los estudiantes en un 14.3%. Se recomienda cargar contenido en diversos formatos para tener más opciones al realizar la recomendación.

Palabras clave: <INGENIERÍA DE SOFTWARE>, <SISTEMA RECOMENDADOR>, <ESTILOS DE APRENDIZAJE>, <DAVID KOLB>, <METODOLOGÍA DE DESARROLLO XP>, <LENGUAJE DE PROGRAMACIÓN JAVA>, <RENDIMIENTO ACADÉMICO>



Firmado electrónicamente por:
ELIZABETH
FERNANDA AREVALO
MEDINA



0276-DBRA-UPT-2022

ABSTRACT

The objective of this work was to develop an educational content recommender system based on learning styles that allows managing and recommending content to students in order to improve their academic performance. Its development contemplated two studies, a preliminary study that established the preferences of content regarding learning styles, and another study to evaluate the academic performance of the students who utilized the system. As data collection techniques, we utilized the following: documentation review, learning style test of David Kolb, test to determine content preference and a structured evaluation with multiple choice questions. The population for the studies were software engineering students from the Escuela Superior Politécnica de Chimborazo, the sample for the first study were 128 students, and for the second study there were two groups of students named control group and experimental group, respectively to apply and compare their results when interacting with the system and its content. We developed the content recommender system with java language programming following the XP methodology and using tools and technologies such as: NetBeans IDE, javascript, Ajax, PrimeFaces and PostgreSQL. The results obtained related to content format preference in relation to learning style demonstrated that the convergent/assimilator groups prefer the video format, while the divergent/accommodator groups prefer the simulation format. The evaluation of academic performance determined that the experimental group obtained an average of 15.60/20, while the control group obtained 12.74/20. We applied the t-student test and we determined that there is a significant difference between the means of the data. We concluded that the use of the recommender system improved student performance by 14.3%. We suggest uploading content in various formats to have more options when making the recommendation.

Key words: <SOFTWARE ENGINEERING>, <RECOMMENDER>, <LEARNING STYLES>, <DAVID KOLB>, <XP SOFTWARE DEVELOPMENT>, <JAVA LANGUAGE PROGRAMMING>, <ACADEMIC PERFORMANCE>.

DIANA
CAROLINA
CAMPAÑA
DIAS

Firmado digitalmente
por DIANA CAROLINA
CAMPAÑA DIAS
Fecha: 2022.02.18
13:30:47 -05'00'

INTRODUCCIÓN

En la actualidad, el constante desarrollo tecnológico ha dado paso a una propagación de las aplicaciones web hacia cualquier área, permitiendo de esta manera la automatización de diferentes procesos realizados por el ser humano con la finalidad de optimizar tiempos, recursos y gastos económicos. Es así como, el área de educación se ha visto beneficiada por la tecnología, recibiendo diversas innovaciones que permiten mejorar el proceso enseñanza-aprendizaje a través de softwares y ambientes e-learning.

Gran parte de los cursos que se imparten de forma online, a través de ambientes e-learning, desarrollan el proceso de enseñanza-aprendizaje de manera tradicional, es decir, sin tomar en cuenta características de los estudiantes para poder personalizar los contenidos y ofrecer un ambiente que favorezca en mayor medida su aprendizaje. De esta manera, se ha visto en la necesidad de implementar un sistema que permita gestionar y recomendar contenidos educativos basados en estilos de aprendizaje, mismo que servirá para potenciar el aprendizaje de los estudiantes y por ende mejorar su rendimiento académico.

La estructuración del presente documento se divide en cuatro capítulos principales, mismos que se detallan a continuación:

CAPÍTULO I: Detalla los antecedentes, la problematización con su respectiva justificación teórica y aplicativa, y se detallan los objetivos generales y específicos del presente trabajo.

CAPÍTULO II: Contiene una revisión sobre trabajos relacionados, la conceptualización teórica de los temas de aprendizaje, así como estilos de aprendizaje, metodología a utilizar, arquitectura del sistema, lenguajes de programación y herramientas de desarrollo.

CAPÍTULO III: Aborda el tipo de estudio, métodos y técnicas empleadas para cada uno de los objetivos, se describiendo el desarrollo del estudio preliminar, procesos aplicados para desarrollar el sistema basado en la metodología XP y desarrollo del estudio de rendimiento académico.

CAPÍTULO IV: Detalla los resultados obtenidos a través de los estudios realizados en el presente trabajo. Finalmente, se describen las conclusiones y recomendaciones del proyecto realizado.

CAPITULO I

1. DIAGNÓSTICO DEL PROBLEMA

1.1. Antecedentes

El proceso de aprendizaje se ha ido adaptando y acoplado a los avances tecnológicos y a las nuevas exigencias de las tecnologías de información y comunicación (TIC's), permitiendo generar nuevos ambientes para dinamizar las relaciones en el proceso de enseñanza-aprendizaje, mediados por modelos como el e-learning y aplicaciones web, a lo cual se denomina el aprendizaje virtual o en red.

En la actualidad, se están definiendo algunos de los factores que podrían incidir sobre este objeto de conocimiento, por ejemplo, la formación universitaria, el rendimiento académico, los tipos de tareas, los dominios disciplinares, los rasgos personales, la edad, el sexo, la culturalidad, entre otros. En este sentido, la relevancia de estos estudios radica básicamente en dos razones: la posibilidad de indagar la naturaleza procedimental de aprendizaje y, simultáneamente, de lograr un análisis centrado en las habilidades y destrezas humanas adquiridas en la interacción social.

Desde niños se adopta diversos métodos de aprendizaje, según (Delgadillo y Guillén, 2007, p.3) indica que el proceso de aprendizaje se puede dar de manera auditiva, kinestésico, lectura/escritura, multimodal y visual, pero en estos tiempos, se ha optado por enseñar en un solo perfil de aprendizaje. Este caso se lo puede ver reflejado en la parte de la comunidad de estudiantes de la ESPOCH, en donde, el proceso de enseñanza-aprendizaje está limitado a presentar el contenido de una sola forma, haciendo que, para aquellos estudiantes que se identifican por otro estilo de aprendizaje puedan presentar falencias que afecten la apropiación del conocimiento.

La sociedad actual ha presentado grandes cambios derivados de la aparición de las TIC's, por ejemplo, (Gallego y Martínez, 2003, p.1-10) en su investigación "Estilos de Aprendizaje y E-learning. Hacia un mayor rendimiento académico", concluye que se tiene un mejor rendimiento y adaptación de los alumnos en sus cursos adaptados a los estilos de aprendizaje.

(Duran y Costaguta, 2008, p.19-28) presenta y discute una experiencia de enseñanza adaptada al estilo de aprendizaje de los estudiantes en un curso de simulación de la carrera de Licenciatura en Sistemas de Información de la Universidad Nacional de Santiago del Estero. Los resultados permiten concluir que es ventajoso adecuar las estrategias de enseñanza al estilo de aprendizaje de los estudiantes, como medio para mejorar la calidad educativa, pero esto por lo general no es tomado en cuenta en sistemas de enseñanza en la comunidad politécnica, lo cual ha despertado interés para realizar la siguiente investigación. Según el trabajo de (Vegas y Petrow, 2008, p.5-37) los estudiantes presentan un desempeño deficiente en las evaluaciones, resultados que se atribuyen a la manera en cómo los estudiantes aprenden y los recursos empleados para ello.

En Ecuador, según (MINTEL, 2013) se plantea la Estrategia Ecuador Digital 2.0 que nace a partir de un exhaustivo diagnóstico del sector TIC, cuya problemática será resuelta por un “conjunto de Políticas Sectoriales que el MINTEL impulsa para que todos los ciudadanos accedan y generen información y conocimiento”, pero no se hace mención a combinar estas herramientas con alguna estrategia para captar el conocimiento de forma viable y eficaz. El proceso de detección de estilos y estrategias de aprendizaje predominantes en estudiantes lo podría hacer un docente que tenga noción en estos temas pedagógicos, pero este proceso no sería eficiente sin un sistema que le brinde soporte.

De acuerdo a los estudios citados anteriormente, se vislumbra que la adaptación de contenidos no se la aplica adecuadamente a los procesos de enseñanza-aprendizaje, por lo que se propone la personalización de contenidos con respecto a perfiles de aprendizaje. Como caso práctico se tomará la materia de “Fundamentos de programación” del primer semestre de la carrera de Ingeniería de Software de la ESPOCH, en la que participará el curso A como grupo de control al cual se les presentará el contenido de formato tradicional, y el curso B como grupo experimental al cual se aplicará el contenido de manera personalizada en base a estilos de aprendizaje. Para cumplir este objetivo se ve necesario la creación del sistema recomendador que sirva para el manejo de los estilos de aprendizaje clasificados por Kolb (Guerrero, 2012, p.27) y estrategias de aprendizaje (Castells, 2001, p.1). El fin de desarrollar este sistema es brindar soporte a los estudiantes para asimilar los contenidos de mejor manera, al presentarlos de forma personalizada en base a su estilo de aprendizaje.

1.2. Formulación del problema

El presente trabajo está dirigido a resolver la siguiente interrogante ¿Cómo se puede mejorar el aprendizaje de los estudiantes de la ESPOCH, usando un sistema recomendador basado en estilos de aprendizaje?

1.3. Sistematización del problema

- ¿Cómo se puede aplicar los sistemas recomendador para el diseño de contenido educativo, basado en los estilos de aprendizaje?
- ¿Cómo se puede personalizar el contenido en base a los perfiles de aprendizaje de cada estudiante?
- ¿De qué forma se puede obtener una aplicación que permita personalizar el contenido educativo?
- ¿Cómo se puede verificar la mejora del rendimiento aplicando el sistema recomendador propuesto?

1.4. Justificación

1.4.1. Justificación teórica

El presente trabajo aporta al proceso de enseñanza-aprendizaje basado en los estilos de aprendizaje, cuyos resultados podrán sistematizarse en una propuesta, para ser incorporado como conocimiento a las ciencias de la educación, ya que se estaría demostrando que el uso de los estilos de aprendizaje mejora el nivel de desempeño de los estudiantes y consigo el rendimiento académico.

La modalidad enseñanza-aprendizaje basados en sistemas recomendadores ofrece herramientas que permiten la implementación de teorías pedagógicas como estilos y estrategias de aprendizaje, los cuales son difícilmente aplicables en un aula tradicional, para conseguir un aprendizaje más efectivo.

La innovación y la tecnología se han convertido en una herramienta de gran valor para el proceso enseñanza-aprendizaje, permitiendo la implementación de teorías pedagógicas como estilos y estrategias de aprendizaje, los cuales son difícilmente aplicables en un aula tradicional, para conseguir un aprendizaje más efectivo. Este es el caso de los sistemas recomendadores, los cuales clasifican a los estudiantes en base a parámetros preestablecidos para presentar ítems acordes a sus características, con la finalidad de fomentar su aprendizaje y mejorar su rendimiento académico. Para este propósito se empleará el test establecido por David Kolb propuesto en 1981 y modificado en 1985, compuesto por un cuestionario de 12 preguntas que permitirá establecer el estilo de aprendizaje de los estudiantes en una de sus cuatro formas (acomodador, convergente, asimilador o divergente).

La metodología ágil utilizada es XP (Programación Extrema), la cual brinda flexibilidad, adaptabilidad y promueve el trabajo en equipo a través de la programación en parejas, con el fin de mejorar el desempeño de desarrollo y minimizar errores que puedan surgir. Entre las herramientas para realizar el sistema se emplearán el IDE NetBeans para llevar a cabo la codificación y como base de datos utilizaremos PostgreSQL, herramientas que son fáciles de utilizar. Algunas ventajas que presenta NetBeans son el soporte para trabajar con diferentes lenguajes de programación, permitir la integración y reutilización de módulos, alta compatibilidad con diferentes programas, funciones para facilitar el desarrollo, entre otras. (ORACLE., 2019).

1.4.2. Justificación aplicada

Los sistemas recomendadores pueden ofrecer a los alumnos una interactividad diferenciada y personalizada tomando en cuenta sus preferencias y necesidades a la hora de presentarles información y actividades durante su proceso de aprendizaje. (Charnelli., 2019, p. 12-65)

El aplicar un sistema recomendador permite mejorar de manera significativa el aprendizaje, basándose en uno o varios parámetros para establecer la recomendación de contenido hacia los estudiantes. Se propone el desarrollo de un sistema recomendador, mismo que permita determinar el estilo de aprendizaje de los estudiantes, siguiendo el modelo de aprendizaje experiencial propuesto por David A. Kolb, con la finalidad de mostrar contenidos personalizados, de tal manera que permita a los estudiantes mejorar su rendimiento académico.

El sistema a desarrollar consta de varios módulos como son: módulo manejador de perfil de estudiante, encargado de construir y mantener el perfil de cada usuario, obteniendo la información de la base de datos; el módulo recomendador de recursos educativos encargado de aplicar un test para determinar el perfil de aprendizaje del estudiante y en base a ello establecer el contenido que satisface las características del usuario y mostrarlo en el módulo interfaz de usuario. Además, se contará con un módulo tutor, que permitirá al docente gestionar el curso y sus contenidos.

Esta investigación se encontrará acorde al Plan Nacional de Desarrollo 2017 - 2021 de la república del Ecuador en su eje número 2 correspondiente a “Economía al servicio de la sociedad”, en su objetivo 5 “Impulsar la productividad y competitividad para el crecimiento económico sostenible de manera redistributiva y solidaria” y su política 10 “Fortalecer e incrementar la eficiencia de las empresas públicas para la provisión de bienes y servicios de calidad, el aprovechamiento responsable de los recursos naturales, la dinamización de la economía, y la intervención estratégica en mercados, maximizando su rentabilidad económica y social”. También se encontrará acorde a las líneas y programas de investigación de la ESPOCH, en su eje de TIC’s, en la línea de investigación de “Tecnologías de la Información y Comunicación” en el programa de “Ingeniería de Software” y acorde a la línea de investigación de la Escuela de Ingeniería en Sistemas (EIS) de “Tecnologías de la Información y Comunicación”, en los programa de “Ingeniería de Software” dentro el ámbito de “Análisis y Diseño de Software” y el programa de “Tecnología Educativa” en el ámbito de “Diseño de materiales educativos multimedia y entornos virtuales de aprendizaje”. En la EIS, los docentes tienen diferentes alternativas para presentar el contenido educativo a los estudiantes, la más usada últimamente es el contenido virtual subida en la plataforma de educación ESPOCH campus virtual. Para la presentación de contenidos virtuales a los estudiantes, los docentes realizan su contenido y lo suben a la plataforma de acuerdo a cada materia que corresponda este proceso, los contenidos se muestran a todos los estudiantes por igual sin tomar en consideración el estilo de aprendizaje de cada alumno. Por lo tanto, los contenidos que se plasman actualmente no son personalizados de acuerdo al modo de adquirir conocimientos de cada individuo.

1.5. Objetivos

1.5.1. Objetivo General

Desarrollar un sistema de recomendación de contenidos educativos basado en estilos de aprendizaje como soporte a la personalización de recursos educativos de una materia de la EIS.

1.5.2. Objetivos Específicos

- Caracterizar los sistemas recomendadores para aplicarlos en el diseño de contenidos educativos mediante revisión de literatura.
- Analizar los estilos de aprendizaje que permitan definir los lineamientos para el desarrollo de contenidos educativos.
- Desarrollar los módulos del sistema recomendador usando el framework PrimeFaces aplicado a la metodología XP.
- Evaluar el rendimiento académico de los estudiantes, que han aplicado los contenidos educativos personalizados del sistema recomendador.

CAPITULO II

2. REVISIÓN DE LA LITERATURA O FUNDAMENTOS TEÓRICOS

2.1. Trabajos Relacionados

En base a la revisión literaria sobre sistemas enfocados en la adaptación de contenidos educativos se concluye que el incorporar estrategias de adaptabilidad dentro del ambiente de aprendizaje, se logra potenciar la capacidad de comprensión y apropiamiento de los contenidos revisados por parte de los estudiantes.

La adaptabilidad en los ambientes e-learning se puede llevar a cabo a través de varias formas como pueden ser adaptabilidad de recursos de aprendizaje/contenido, adaptabilidad del seguimiento, interfaz, entre otros. Pero la que mayormente se ocupa dentro de las plataformas e-learning corresponde a la adaptabilidad del contenido. (Humam, 2021, p.384)

Para el desarrollo de sistemas de personalización de contenidos educativos cada autor emplea diversos parámetros, construyendo de esta manera su propia estrategia de personalización a aplicar en escenarios de aprendizaje. En base a (Essalmi, 2010, p.583), y revisión bibliográfica de otros autores, en la **Tabla 1-2**. se muestra un listado de sistemas de personalización e-learning, que se han sido desarrollados a lo largo de la historia, cada uno aplicando sus propios parámetros de personalización.

Tabla 1-2: Sistemas de personalización e-learning con su parámetro de personalización

Sistema de personalización e-learning	Parámetro de personalización
POP (Höök et al., 1996; citado en Essalmi, 2010, p.583)	Información, búsqueda de tareas
Interbook (Brusilovsky et al., 1996; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante
The Intelligent Helpdesk (Greer et al., 1998; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante, metas de aprendizaje
ECSAI (Grandbastien, 1999; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante, metas de aprendizaje
KBS-hyperbook (Henze y Nejd, 1999; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante, metas de aprendizaje
VC prolog tutor (Peylo, Thelen, Rollinger, y Gust, 2000; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante, metas de aprendizaje
NetCoach (Weber et al., 2001; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante, metas de aprendizaje, preferencia de medios
German Tutor (Heift y Nicholson, 2001; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante

Sistema de personalización e-learning	Parámetro de personalización
ActiveMath (Melis et al., 2001; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante, metas de aprendizaje, preferencia de medios
ELM-ART (Weber y Brusilovsky, 2001; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante, preferencia de medios, preferencia del lenguaje (inglés o alemán)
KOD (Sampson, Karagiannidis, y Cardinali, 2002; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante, preferencia del lenguaje, metas de aprendizaje
SIMBAD (Bouzeghoub et al., 2003; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante, metas de aprendizaje, preferencia de medios
MetaLinks (Murray, 2003; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante, metas de aprendizaje, preferencia de medios
INSPIRE (Papanikolaou et al., 2003; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante, metas de aprendizaje, estilo de aprendizaje (Honey y Munford)
MLTutor (Smith y Blandford, 2003; citado en Essalmi, 2010, p.583)	Metas de aprendizaje (basadas en el historial de búsqueda del usuario)
COLER (Constantino-González et al., 2003; citado en Essalmi, 2010, p.583)	Saldo de participación, progreso en tareas, espera por retroalimentación
SQL-Tutor (Mitrovic, 2003; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante
EPSILON (Soller, 2004; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante
SIETTE (Conejo et al. 2004; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante
PERSO (Chorfi y Jemni, 2004; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante, preferencia de medios
ELENA (Dolog, Henze, Nejd, y Sintek, 2005; citado en Essalmi, 2010, p.583)	Nivel de conocimiento del estudiante, preferencia del lenguaje, metas de aprendizaje
AHA! (Stash et al., 2006; citado en Essalmi, 2010, p.583)	Estilos de aprendizaje (Felder-Silverman), preferencia de medios, preferencia de navegación
Modelo semántico para recomendar recursos educativos (Morocho y Carranza, 2017, p. 21-69)	Preferencia de lenguaje, preferencia de medios, preferencia de navegación, estilos de aprendizaje (Felder-Silverman)
Recomendador de usuarios en base al perfil y reputación (Aciar y Aciar, 2013, p.1-11)	Perfil de usuario y reputación.
Recomendador de contenidos accesibles basados en perfiles de usuarios para ambientes virtuales y objetos de aprendizaje a partir de metadatos de accesibilidad haciendo uso de ontologías (Vera y Ulloa, 2018, p.46-90)	Contenidos y nivel de conocimientos del estudiante.

Sistema de personalización e-learning	Parámetro de personalización
PLCP (Jeong et al., 2012, p.2567-2577)	Perfil de usuario, nivel de conocimiento del estudiante, preferencias del estudiante.
CleverUniversity (Humam y Ufuoma, 2021, 381-393)	Estilos de aprendizaje (Felder-Silverman)

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Tal como se puede observar en la **Tabla 1-2.**, existen diversos sistemas de personalización e-learning, todos únicos a su propia manera, ya que aplican diferentes estilos de personalización mediante la agrupación de uno o más parámetros con el fin de brindar una personalización distinta, pero todas encaminadas a potenciar el aprendizaje de los estudiantes.

Una de las estrategias más empleadas en la personalización de escenarios de aprendizaje es la personalización en base al estilo de aprendizaje, ya que contribuye a elevar los niveles de rendimiento educativo de los estudiantes. (Gutiérrez, 2018, p.94)

(Aciar, 2013, p.3), describe el desarrollo de un sistema recomendador de usuarios basado en los parámetros de perfil y reputación, donde este último es el resultado de interacciones pasadas del usuario dentro de la plataforma colaborativa. Como resultado de aplicar este sistema recomendador se mostró una mejora en la satisfacción y rendimiento de trabajo por parte de los usuarios, sin embargo, este sistema recomendador se centra únicamente en recomendar usuarios para trabajar de manera colaborativa, mientras que el sistema recomendador propuesto en el presente trabajo pretende recomendar contenido educativo a los usuarios.

(Vera y Ulloa, 2018, p.46-90), aborda el diseño y desarrollo de un sistema recomendador de contenidos accesibles basado en perfiles de usuarios donde emplea objetos de aprendizaje haciendo uso de ontologías. Este sistema recomendador busca proveer de contenido accesible a aquellas personas que presentan las capacidades especiales. Por otro lado, el sistema recomendador propuesto en el presente trabajo se centra en recomendar contenido a todos los usuarios basándose en su estilo de aprendizaje.

(Jeong et al., 2012, p.2567-2577), propone el desarrollo de un Planificador de cursos de aprendizaje personalizado (PLCP), el cual permita a los estudiantes elegir que contenido de aprendizaje desean visualizar en base a sus características y nivel de educación, en base a ello recomendar una estructura de curso que facilite su aprendizaje, mientras que el sistema recomendador propuesto en el presente trabajo recomendará el contenido en un formato acorde al estilo de aprendizaje del estudiante.

(Humam y Ufuoma, 2021, 381-393) describe el diseño e implementación de un sistema hipermedia dinámico adaptativo llamado “CleverUniversity”, el cual emplea a los estilos de aprendizaje de Felder-Silverman como uno de los más importantes factores para recomendar y personalizar el

contenido, en contraste el sistema propuesto se basa en los estilos de aprendizaje de Kolb para gestionar la personalización de contenidos.

2.2. Aprendizaje

El aprendizaje es un “proceso de cambio relativamente permanente en el comportamiento de una persona generado por la experiencia” (García, 2018, p.218-228). Mientras que para (Zea Silva et al., 2016, p.32) “es un proceso holístico en el cual el sujeto construye su conocimiento de manera integral.” En base a las dos concepciones mencionadas se establece que el aprendizaje es un proceso en el cual una persona adquiere conocimiento a través de la experiencia.

2.2.1. Aprendizaje experiencial

Como su nombre lo indica, este es un tipo aprendizaje activo el cual se basa en el proceso de aprender mediante la experiencia y reflexión. Para John Dewey, la experiencia es una herramienta fundamental para una auténtica educación, un aprendizaje experiencial es activo ya que una persona que aprende no solo se queda en su interior si no que se refleja en su entorno, para cambiar entornos físicos y sociales. (Dewey, 2014; citados en León Grijalva, 2014)

2.3. Estilos de aprendizaje

En el transcurso del tiempo algunos psicólogos de la educación han propuesto algunas definiciones, clasificaciones e instrumentos de diagnóstico de los estilos de aprendizaje, como por ejemplo para autores como Hunt, D, describe las condiciones bajo las que un alumno está en la mejor situación para aprender, o qué estructura necesita para aprender mejor (Hunt, 1979; citado en Cabrera y Fariñas, 2005). Además, para Reinert que asume el Estilo de Aprendizaje como la aptitud o el talento, es para él la manera en que cada individuo se encuentra “programado”, es decir, el cómo recibir, comprender, memorizar y ser capaz de utilizar la nueva información (Reinert, 1976; citado en Silva, 2018). Para Ocaña los estilos de aprendizaje “se refiere al hecho que cuando queremos aprender algo, cada uno de nosotros utiliza su propio método o conjunto de estrategias”. (Ocaña, 2010; citado en Trujillo y Vega, 2015)

David Kolb propone una definición de los estilos de aprendizaje de acuerdo a su modelo de aprendizaje en base a la experiencia y lo describe como:

“algunas capacidades de aprender que se destacan por encima de otras como resultado del aparato hereditario, de las experiencias vitales propias, y de las exigencias del medio actual. Llegamos a resolver de manera característica, los conflictos entre el ser activo y reflexivo y entre el ser inmediato y analítico. Algunas personas desarrollan mentes que sobresalen en la conversión de hechos dispares en teorías coherentes, y, sin embargo, estas mismas personas son

incapaces de deducir hipótesis a partir de su teoría, o no se interesan por hacerlo; otras personas son genios lógicos, pero encuentran imposible sumergirse en una experiencia y entregarse a ella” (Kolb, 1984; citado en Cabrera y Fariñas, 2005).

Además, Eiszler en 1983 menciona que los estilos de aprendizaje tienen una relación muy significativa con el proceso de aprendizaje, el cual también recibe una influencia de la cultura, la experiencia y el desarrollo lo que da a concluir que el ambiente en el que se desarrolla y desenvuelve una persona forma parte de este proceso. (Eiszler, 1984; citado en Trujillo y Vega, 2015).

2.4. Modelos de aprendizaje

Es un conjunto de rasgos, estrategias y pautas a seguir con el fin de orientar y potenciar el proceso de aprendizaje. Algunas investigaciones demuestran que las personas tienen diferentes formas de aprender, esto depende como el individuo se relaciona con el mundo y que es lo que desea aprender, de aquí nace los estilos de aprendizaje, siendo rasgos cognitivos, afectivos y fisiológicos. (Rodríguez Cepeda, 2017, p.51-64)

2.5. Clasificación de los modelos de aprendizaje por autor

Existen diferentes teorías y modelos que estudian características de los individuos, a continuación, según (Pantoja Ospina, 2013, p.82-86) se muestran algunos modelos de aprendizaje propuestos por varios autores.

Tabla 2-2: Modelos de estilos de aprendizaje

Autor	Característica	Estilos de aprendizaje
Dewey (1897)	Centrado en la teoría práctica, la vinculación de la experiencia con el proceso de aprendizaje	1. Experimentalismo
Bloom (1956)	Consta de seis etapas: recordar, comprender, aplicar, analizar, sintetizar y evaluar. A través de estas se da el proceso de aprendizaje.	No aplica.
Argyris y Schön (1978)	Se centran en la forma como el individuo aprende dentro de la organización.	1. Aprendizaje de circuito simple 2. Aprendizaje de circuito doble 3. Aprendizaje secundario
Dunn et ál. (1979)	Determinan 21 factores que influyen en el aprendizaje, clasificándolos en cinco canales: ambiental, fisiológico, psicológico, sociológico y emocional.	1. Visual 2. Auditivo 3. Táctil o kinestésico
David Kolb (1984)	Basado en la experiencia.	1. Divergente 2. Asimilador

Autor	Característica	Estilos de aprendizaje
	Propone un test compuesto de 12 preguntas para establecer el tipo de estilo de aprendizaje que tiene cada persona.	3. Convergente 4. Acomodador
Honey y Mumford	Replantean las teorías de Kolb, definiendo las cuatro etapas como: experiencia, reflexión, elaboración de hipótesis y aplicación. Cada fase se asocia a los estilos de aprendizaje	1. Activo 2. Reflexivo 3. Teórico 4. Pragmático
Felder y Silverman (1988)	Proponen que el estilo de aprendizaje está compuesto por algún rasgo bipolar en cuatro dimensiones: activo/reflexivo, sensitivo/intuitivo, visual/verbal y secuencial/global.	1. Activo/reflexivo 2. Sensitivo/intuitivo 3. Visual/verbal 4. secuencial/global
VerLee Williams (1988)	Basa sus estudios en el contraste existente entre los dos hemisferios del cerebro.	1. Senistrohemisferios 2. Dextrohemisfericos
Gardner (1997)	Propone la existencia de ocho tipos de inteligencia: lingüística, matemática, corporal–kinética, espacial, musical, interpersonal, intrapersonal y naturalista. Influencia	1. Lingüística 2. Matemática 3. Corporal Kinxetica 4. Espacial 5. Musical 6. Inter personalidad 7. Intrapersonal 8. Naturalista
Entwistle (1998)	Relaciona características de personalidad, motivación y enfoque de aprendizaje para plantear tres estilos: superficial, profundo y estratégico.	1. Superficial 2. Profundo 3. Estratégico
Charles Owen (1997 y 1998)	Abstracción del conocimiento a través de dos fases: analítica y sintética, que a su vez operan en dos mundos: analítico y práctico.	1. Analítico y sintético 2. Analítico y practico
Clark (2002)	Parte de la adaptación de Anderson y Krathworhl para asignarle actividades a cada fase. Para recordar: preguntar, escuchar, localizar, observar, identificar y descubrir. Para aplicar: manipular, enseñar, experimentar y entrevistar. Para analizar: clasificar, categorizar, comparar, contrastar y encuestar. Para evaluar: juzgar, decidir, elegir, debatir, recomendar. Para crear: combinar, componer, inventar, inferir, imaginar y producir.	1. Actuar 2. Reflexionar 3. Experimentar 4. Teorizar

Autor	Característica	Estilos de aprendizaje
Cacheiro et ál. (2008)	Proponen el modelo EAAP, que parte de los principios de Honey y Mumford y adaptados por Alonso. De la combinación de factores se generan trece filtros en cuatro tipos: monofásicos, bifásicos, trifásicos y eclécticos.	<ol style="list-style-type: none"> 1. Monofásicos 2. Bifásicos 3. Trifásicos 4. Eclécticos

Fuente: (Pantoja Ospina, 2013, p.82-86)

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En la **Tabla 2-2** se indica varios modelos de los cuales se va a tomar los propuestos por David Kolb, Honey y Mumford, VerLee Williams y Felder y Silverman para poder realizar un análisis comparativo con el cual se selecciona el modelo a trabajar.

2.5.1. Comparativa de modelos de aprendizaje

Tabla 3-2: Tabla comparativa de modelos de aprendizaje

Modelo	Categoría	Test	Proceso de aprendizaje	Identificar estilos
Kolb	Basados en la experiencia	Dispone de test, cuyo objetivo es identificar características personales de la forma en como procesa la información.	<p>Considera que el aprendizaje es un proceso cíclico en el cual se crea el conocimiento a través de la experiencia que es base de la observación y reflexión, para dar paso al conocimiento.</p> <p>El proceso de aprendizaje se lo divide en dos dimensiones, la perceptiva relacionada a los sentidos y la de procesamiento en la cual intervienen características intrínsecas de los individuos.</p>	Logra identificar al estudiante dentro de un perfil completo, en el que se pueden observar las principales fortalezas, así como las debilidades.
Honey y Mumford	Basados en la experiencia	Si dispone de test, cuyo objetivo es identificar estilo preferido de aprendizaje.	Consideran al aprendizaje como un proceso circular en donde todo es parte de la experiencia y se adquiere conocimiento mediante ella. El proceso de aprendizaje lo agrupa en procesos actitudinales y de comportamiento.	Define los estilos de aprendizaje dentro de un perfil, pero influye mucho la actitud y el comportamiento del estudiante.

Modelo	Categoría	Test	Proceso de aprendizaje	Identificar estilos
VerLee Williams	Bilateralidad cerebral	Dispone de un cuestionario cuyo objetivo es determinar con que hemisferio del cerebro aprende.	Considera que el conocimiento se puede adquirir con cualquiera de los dos hemisferios (derecho o izquierdo), pero lo óptimo es usar todo el cerebro.	No define como tal, estilos de aprendizaje.
Felder y Silverman	Basados en canales de percepción de Información.	Dispone de un cuestionario cuyo objetivo es saber de las preferencias en cuatro dimensiones.	Considera que el aprendizaje se lo adquiere a través de los canales de percepción y que para definir estilos de aprendizaje se involucra las preferencias del estudiante y educador.	Identifican aspectos concretos del aprendizaje y dos posibles opciones para cada uno de ellos, pero no en un perfil completo.

Fuente: (Pantoja Ospina, 2013, p.82-86)

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En base a la comparativa de los modelos presentados en la **Tabla 3-2**, se visualiza que el modelo de aprendizaje propuesto por David Kolb presenta mayores beneficios para ser aplicado en el presente trabajo debido a que a más de establecer que el aprendizaje como un proceso experiencial haciendo énfasis en los sentidos, posee una herramienta para la clasificación de los estudiantes de acuerdo a su estilo de aprendizaje en uno de los cuatro estilos que se establece en dicho modelo, mismos que, de acuerdo a sus características, permiten establecer el tipo de contenido apropiado en el que se presentaran al estudiante buscando una potenciación del aprendizaje. Por estos motivos se escogió el modelo de estilos de aprendizaje propuesto por David Kolb como el modelo de estilos de aprendizaje a utilizar en el desarrollo del presente trabajo.

2.5.2. Análisis de estilos de aprendizaje del modelo experimental de David Kolb

David A. Kolb (1939) es un psicólogo y teórico de la educación que estudio en la universidad de Harvard donde consiguió su título de masterado y doctorado en psicología social. Es conocido principalmente por sus aportes en el campo de aprendizaje experiencial, la cual se enfoca en la relevancia que tiene la experiencia en el proceso de aprendizaje del individuo, dando como fruto de sus investigaciones la identificación de las diferentes maneras en las que los individuos perciben y procesan la información, es decir, los estilos de aprendizaje. (Trujillo y Vega, 2015, p.7)

El modelo propuesto por David Kolb distingue cuatro tipologías de aprendizaje en base a la forma en la que las personas prefieren tratar con la información de su entorno. Dichos estilos de aprendizaje se describen en la **Tabla 4-2**.

Tabla 4-2: Estilos de aprendizaje de Kolb

Estilo	Descripción	Características
Divergente	Combinan experiencia concreta con observación reflexiva. Son individuos con habilidad imaginativa que ven situaciones desde diferentes perspectivas.	Aprenden con el movimiento. Son experimentales, flexibles con las actividades, tienen creatividad, no les gustan las normas, conceptos, tienen buenas ideas.
Asimiladores	Combinan conceptualización abstracta y observación reflexiva. Son individuos con habilidad para crear modelos teóricos, con razonamiento inductivo, preocupados más por los conceptos que por el uso práctico de las teorías.	Aprenden analizando, son estudiosos, utilizan el razonamiento deductivo, son individualistas, son pensadores abstractos, tienen capacidad de síntesis, generadores de modelos y teorías, son poco empáticos, herméticos, poco sensibles.
Convergente	Combinan conceptualización abstracta y experimentación activa. Son personas interesadas en la aplicación práctica de las ideas, buenas en situaciones donde hay más de una respuesta, no son emotivos.	Aprenden solucionando problemas, se les facilita transferir lo aprendido, les gusta las nuevas experiencias, son eficientes en la aplicación de teorías, son buenos con actividades técnicas, son herméticos, imaginativos, deductivos, líderes y poco sensibles.
Acomodadores	Combinan la experiencia concreta y la experimentación activa, son personas con habilidad para llevar a cabo planes orientados a la acción, son arriesgados, les gustan las nuevas experiencias, se adaptan a las circunstancias inmediatas, son intuitivos y aprenden por prueba y error.	Aprenden observando, tienen capacidad de relacionar diversos contenidos, son imaginativos, extrapolan contenidos, le dan importancia al entorno de trabajo, son sociables y espontáneos, impulsivos, les gusta depender de los demás y tienen poca capacidad analítica.

Fuente: (Guerrero Vergel, 2019, p.27)

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Los cuatro estilos de aprendizajes definidos por Kolb abarcan una variedad de características que las personas tienen al momento de adquirir el conocimiento dando una ventaja al aplicarlo en los estudiantes, esto hace que el modelo de David Kolb sobresalga entre los otros modelos.

➤ Preferencia de formato de contenido educativo basado en estilos de aprendizaje

Sahabudin Noor y Ali Mohamad (Sahabudin y Bilal, 2013, p.714), en su estudio establecían una relación entre estilos de aprendizaje y formatos de contenidos educativos, donde estos últimos se encontraban divididos en cuatro tipos como son: texto, gráficos, video y XML, tal como se observa en la **Tabla 5-2**.

Tabla 5-2: Relación entre estilo de aprendizaje con formato de contenido educativo

Estilo de aprendizaje	Formato de contenido educativo
Convergente	Texto
Divergente	Texto
Asimilador	Video
Acomodador	Video

Fuente: (Sahabudin y Bilal, 2013, p.715)

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Los resultados la investigación realizada por (Sahabudin y Bilal, 2013, p.715) se tomaron como base para el desarrollo de la presente investigación, pero, con respecto a los contenidos educativos se tomó en cuenta 8 tipos de contenido que son: texto, imagen, audio, video, objeto de aprendizaje (OA), animación, simulación y juego.

2.6. Rendimiento académico

El rendimiento académico se lo concibe como el resultado de una serie factores y actividades provenientes del proceso educativo. De acuerdo a Tejedor, citado en (Gómez et al., 2011, p.91) concibe el rendimiento académico como “nota o calificación media obtenida durante el periodo universitario que cada alumno haya cursado”. Esta calificación reflejara el nivel de aprendizaje del estudiante sobre un tema específico durante un periodo determinado.

2.6.1. Como medir el rendimiento académico

Según (Hernández Herrera, 2015) el rendimiento académico se puede medir mediante la observación de los resultados representados en calificaciones que se pueden obtener de exámenes, mismas que son notas definidas en una escala.

El rendimiento académico se lo expresa mediante una calificación cualitativa y cuantitativa que indicará el aprendizaje y cumplimiento de logros establecidos por el estudiante (Gómez et al., 2011, p.91), por lo que, el método más frecuentemente empleado para medirlo es una evaluación, que consta de una serie de preguntas relacionadas a un tema específico, con una valoración cuantitativa.

2.7. Metodología de desarrollo

Las metodologías de desarrollo de software comprenden un conjunto de procesos, técnicas y documentación para planificar, desarrollar y controlar el desarrollo de un producto software de calidad (Molina Ríos et al., 2018, p.4). Estas metodologías que surgieron con el fin de brindar soporte durante el proceso de desarrollo de software han ido evolucionando con el paso del tiempo,

dividiéndose de esta manera en metodologías tradicionales y ágiles, cuyas características se detallan en la **Tabla 6-2**.

Tabla 6-2: Comparativa metodologías de desarrollo tradicional-ágil

Metodología tradicional	Metodología ágil
Predictivos.	Adaptativos.
Está orientado a procesos.	Está orientado a personas.
Proceso rígido.	Proceso flexible.
Se concibe un gran proyecto solamente.	Subdivide el proyecto en otros de menor dimensión.
Escasa interacción con el cliente.	Constante interacción con el cliente.
Una sola entrega del software al finalizar el proyecto.	Entregas periódicas de software durante el proyecto.
Extensa documentación.	Escasa documentación.

Fuente: (Navarro Cadavid et al., 2013, p.31)

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En base a la **Tabla 6-2**, en la que se presenta la comparativa entre metodologías tradicionales y ágiles para el desarrollo de software, se vislumbra que las metodologías ágiles, como su nombre lo indican buscan un desarrollo ágil enfocado en la flexibilidad para adaptarse a cambios que puedan surgir durante el proceso, interacción con el cliente y entregas periódicas para mantener un control del desarrollo, y software sobre documentación, por lo que, dichas metodologías ágiles, son preferidas en la actualidad.

2.7.1. Metodologías ágiles

Las metodologías ágiles proveen un paradigma de desarrollo iterativo e incremental que hace énfasis en el desarrollo, interacción, y adaptabilidad (Faiza et al., 2017, p.1). De entre las metodologías ágiles más empleadas en la actualidad, se destacan dos que corresponden a la metodología SCRUM y la metodología XP, cuya comparación se muestra en la **Tabla 7-2**.

Tabla 7-2: Comparativa de metodologías ágiles XP-SCRUM

Característica	XP	SCRUM
Desarrollo	Iterativo e incremental	Iterativo e incremental
Tamaño de proyecto	Pequeño	Todo tipo de tamaño
Tamaño de equipo	2 a 10 personas	Múltiples equipos de al menos 10 miembros cada uno.
Actividades grupales	<ul style="list-style-type: none"> • Juego de planificación • Programación en parejas • Propiedad de código colectiva, etc. 	No
Duración de iteración/sprint	1 a 3 semanas	4 semanas
Intervención de interesados	Durante todo el proceso	No definido
Obtención de requerimientos	Historias de usuario	No definido

Característica	XP	SCRUM
Documentación	Poca	Poca
Cambios durante iteración	Permitidos	No permitidos

Fuente: (Faiza et al., 2017, p. 6)

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En base a la **Tabla 7-2** en la que se presenta la comparativa entre metodologías ágiles XP y SCRUM, se vislumbra que la metodología que mejor se adapta para el desarrollo del presente trabajo es la metodología XP, por ser una metodología ágil y flexible que promueve el trabajo en equipo, aplica la programación en parejas y se adapta a los cambios que puedan surgir durante el proceso de desarrollo.

2.7.2. Metodología XP

Programación Extreme (XP) es una metodología ágil, que tiene como objetivos el desarrollo y gestión de proyectos con eficiencia, control y flexibilidad, centrándose en potenciar relaciones interpersonales como una estrategia para un desarrollo de software exitoso. Gracias a esta estrategia se promueve un trabajo en equipo, aprendizaje continuo, generando un buen ambiente de trabajo.

XP se basa en la retroalimentación continua, la comunicación fluida entre el cliente y el equipo de desarrollo, la simplicidad en las soluciones implementadas y fuerza para aceptar los cambios (Letelier y Penádes, 2006, p.1-17). Es por esto que XP se adapta perfectamente para proyectos que tienen requisitos muy variantes e imprecisos.

➤ Roles de XP

Según Beck citado en (Letelier y Penádes, 2006, p.1-17), menciona que en la metodología XP se involucran varios roles, mismos que se encuentran detallados en la **Tabla 8-2**.

Tabla 8-2: Descripción de los roles establecidos en la metodología XP

Rol	Descripción
Cliente	Persona que define las historias de usuario y las pruebas funcionales para la validación de la implementación.
Programadores	Personas que escriben las pruebas y desarrollan el código del sistema, además estiman tiempo de desarrollo del mismo.
Tester	Persona encargada de hacer pruebas, ayudando al cliente a escribir pruebas funcionales.
Tracker	Encargado de seguimiento proporciona realimentación al equipo en el proceso XP. Su objetivo es verificar el grado de acierto entre las estimaciones realizadas y el tiempo real dedicado.

Rol	Descripción
Coach	Persona responsable de guiar y orientar al equipo, es necesario que conozca a fondo el proceso XP.
Consultor	Es el miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto. Guía al equipo para resolver un problema específico.
Big Boss	Es el gestor del proyecto, vínculo entre clientes y programadores, ayuda a que el equipo trabaje efectivamente creando las condiciones adecuadas.

Fuente: (Letelier y Penádes, 2006, p.1-17)

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ *Proceso XP*

Un proyecto desarrollado con la metodología XP consigue el éxito cuando el cliente selecciona el valor de negocio a implementar basado en la habilidad del equipo para medir la funcionalidad que se puede entregar a través del tiempo. Para el desarrollo correcto de la metodología XP se deben seguir 6 fases que conforman su ciclo de vida, el cual se encuentra detallado en la **Figura 1-1**.

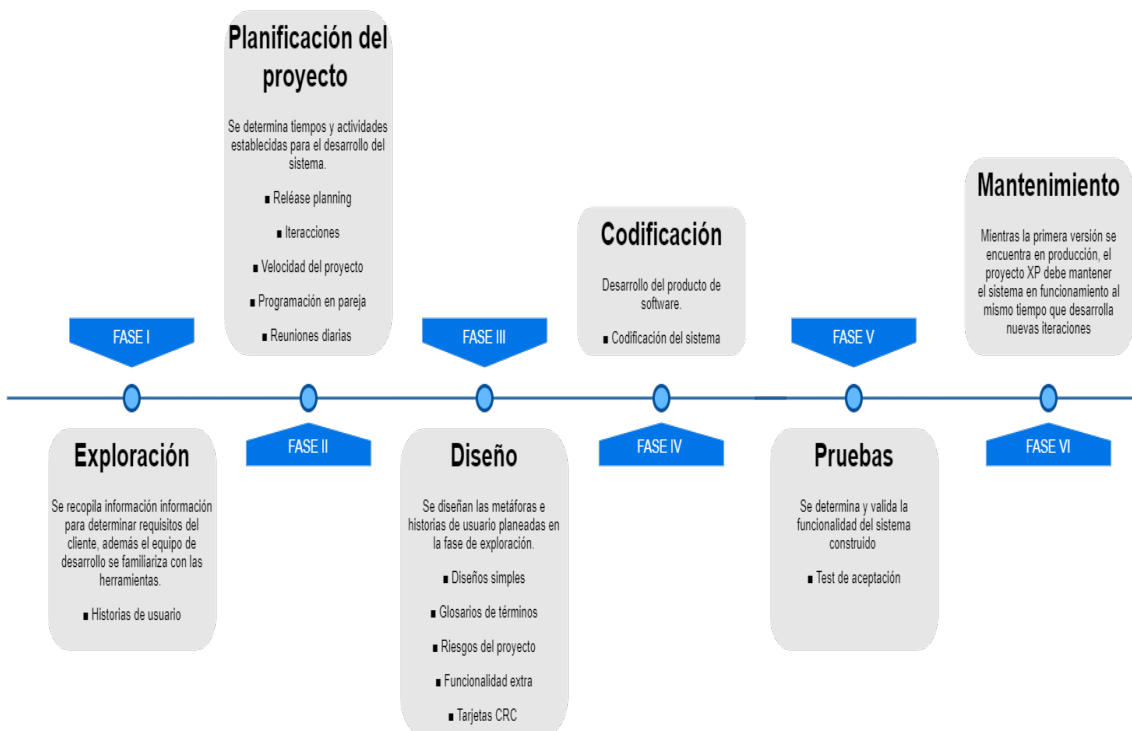


Figura 1-1: Diagrama de fases de la metodología XP

Fuente: (Letelier y Penádes, 2006, p.1-17)

2.8. Arquitectura de software

La arquitectura de software es la parte principal para el desarrollo de sistemas, ya que es donde consta los componentes de software, las propiedades visibles externamente y las relaciones entre ellas.

2.8.1. Patrón modelo-vista-controlador

El patrón de diseño modelo-vista-controlador (MVC) comprende un patrón de arquitectura de software que detalla la manera en la que se deberá estructurar la aplicación, para lo cual la divide en tres niveles (modelo, vista y controlador), separando los datos de la lógica y la interfaz de usuario con el fin de mejorar la flexibilidad y reutilización para facilitar el desarrollo y mantenimiento del sistema. (Ríos Luna, 2014, p.13-53)

➤ Niveles MVC

- **Modelo:** modulo encargado de gestionar las reglas del negocio (funcionalidades) del sistema y desempeña el rol de intermediario para el almacenamiento de datos. (Gloria, et al., 2018, p.244)
- **Vista:** modulo encargado de presentar de manera visual el modelo, generando de esta manera la interfaz gráfica del sistema, la misma que sirve de intermediaria entre la aplicación y el usuario. (Gloria, et al., 2018, p.244)
- **Controlador:** modulo que desempeña la función de intermediario entre el modelo y la vista, siendo así que recepta las órdenes del usuario para enviárselas al modelo el cual realiza el proceso y finalmente las muestra en la vista correspondiente. (Gloria, et al., 2018, p.244)

2.9. Lenguajes de programación

Los lenguajes de programación constituyen lenguajes formales que permiten el desarrollo de programas, que son un conjunto de instrucciones, para ser ejecutado en un ordenador controlando su comportamiento físico y lógico. Dentro de los lenguajes de programación existen tres tipos: lenguaje de máquina, lenguaje ensamblador y lenguaje de alto nivel, donde este último posee una sintaxis similar al lenguaje humano, haciendo más sencillo su utilización. (Salvador Sánchez, 2014, p.1). Dentro de los lenguajes de alto nivel más utilizados actualmente se tienen a: C, Java, Python, PHP, entre otros. (TIOBE, 2020, p.1)

2.9.1. Comparativa de lenguajes de programación

Cada lenguaje de programación tiene sus propias fortalezas como falencias, las mismas que determinan las características de cada uno y establecen los criterios para que los desarrolladores opten por uno de ellos. En la **Tabla 9-2** se muestra un análisis comparativo entre 3 de los principales lenguajes de programación que son: Java, Python y PHP.

Tabla 9-2: Comparativa de lenguajes de programación Java, Python y PHP.

Característica	Java	Python	PHP
Costo de servidor	Gratis	Gratis	Gratis
Orientación a objetos	Si	Si	Parcialmente
Sistema operativo	Linux o Windows	Linux o Windows	Linux o Windows
Servidor	Apache, Tomcat y Glassfish	Apache	Apache
Base de datos	Oracle, MySQL	MySQL y PostgreSQL	MySQL
Énfasis	Desarrollo dinámico de aplicaciones web	Productividad y facilitar la lectura de código	Desarrollo dinámico de aplicaciones web
Documentación de apoyo	Mucha - descentralizada	Mucha – descentralizada	Mucha – descentralizada
Principales ambientes de desarrollo	NetBeans, Eclipse y otras herramientas Open Source	NetBeans, Eclipse y otras herramientas Open Source	NetBeans, Eclipse y otras herramientas Open Source

Fuente: (Valarezo Pardo et al., 2018, p.42)

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En base a la **Tabla 9-2**, se eligió al lenguaje de programación Java para el desarrollo de la aplicación en el presente trabajo, debido a que es orientado a objetos, facilitando el desarrollo estructurado y reutilización de código, además, de contar con basta documentación de apoyo para guiar el desarrollo, y principalmente por su alta compatibilidad con el IDE NetBeans y otras herramientas que se emplearan en el presente proyecto.

2.10. Herramientas de desarrollo

2.10.1. Java

Desarrollado por Sun microsystems es un lenguaje de programación orientado a objetos parecido al C++, usado para la elaboración de pequeñas aplicaciones exportables a la red (applets) capaces de operar sobre cualquier plataforma a través de navegadores WWW, permitiendo dar dinamismo a las páginas web. (Gómez Fajardo y Lara Silva, 2014, p. 46-65)

Características

- Posee licencia GNU GPL, por lo que es un software libre y gratuito. (Valarezo Pardo et al., 2018, p.42)
- Es un lenguaje de programación de propósito general orientado a objetos. (Gosling et al., 2020)
- Está relacionado con C y C++, pero organizado de manera diferente. (Gosling et al., 2020)
- Es un lenguaje de programación multiplataforma. (Valarezo Pardo et al., 2018, p.42)
- Es un lenguaje de programación robusto de alto nivel. (Gosling et al., 2020)
- Posee una basta documentación de apoyo. (Valarezo Pardo et al., 2018, p.42)

- Es eficaz, ya que posee un proceso de optimización en la interpretación de su código. (Pazmiño Haro y Torres Arroba, 2018, p.16)
- Permite el desarrollo de aplicaciones web distribuidas mediante la implementación de protocolos de red estándar. (Pazmiño Haro y Torres Arroba, 2018, p.16)

Componentes de Java

Dentro de java se emplean diversos componentes, tal como se detalla en la **Tabla 10-2**.

Tabla 10-2: Componentes de Java

Componente Java	Descripción
Máquina virtual de Java (JVM)	Permite ejecutar los programas desarrollados con Java, volviéndolo multiplataforma. Se encarga de la carga de clases, gestión de la memoria y seguridad.
Interfaz de programación de aplicaciones Java (API Java)	Están conformadas por un conjunto de elementos software prefabricados que incluyen clases e interfaces con varias funcionalidades.
Herramienta de despliegue de aplicaciones	Ayudan en el despliegue de aplicaciones a través de Java Web Start y Java Plug-in.
Herramienta de ayuda al desarrollo	Son un conjunto de herramientas que brindan apoyo al desarrollador en tareas como: compilar, generar automáticamente la documentación del código fuente, lanzar la ejecución y visualizar la ejecución del applet Java.

Fuente: (Pazmiño Haro y Torres Arroba, 2018, p.9-36)

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

2.10.2. JavaScript

Es un lenguaje de programación WEB que permite crear contenido dinámico para mejorar la interacción del usuario con la página web. Este dinamismo se lo consigue por medio de eventos que interpreta JavaScript, lenguaje que no requiere de un compilador, haciendo que la interpretación del mismo se desarrolle durante la ejecución. Este lenguaje se ejecuta en el navegador del cliente, por lo que la respuesta es mucho más rápida, permitiendo desarrollar aplicaciones RIA (Rich Internet Application.) con mayor flexibilidad y dinamismo. (Rúales Pintado y Suarez Quispe, 2015, p.30)

Características

A continuación, se presentan algunas características propuestas por (Pazmiño Haro y Torres Arroba, 2018, p.19) de JavaScript:

- Es un lenguaje de cliente web, por lo que va desde el servidor web hacia el navegador, ejecutándose rápidamente en este último.
- Es un lenguaje interpretado (justo a tiempo), dado que el navegador recibe el programa a manera de código fuente y traduce a código binario sus instrucciones para ejecutarlas.
- Es orientado a eventos, lo que permite sustituir partes de una página debido a que se define eventos que se relacionan a las acciones que realiza el usuario.

2.10.3. JQuery

Es una librería JavaScript rápida, pequeña y rica en funciones que permiten recorrer y manipular documentos HTML, manejar de eventos, animaciones, y Ajax, de manera mucho más simple y fácil brindando así una combinación de versatilidad y extensibilidad durante el desarrollo. (js.foundation, 2015)

Una de las principales funcionalidades que brinda JQuery es permitir una mayor flexibilidad a las páginas web, pudiendo cambiar su contenido sin requerir actualizar la página por completo. (Villarreal Fuentes, 2013)

Características

- Permite realizar efectos y animaciones personalizadas. (Villarreal Fuentes, 2013)
- Facilidad de comprender y estudiar el código. (Pazmiño Haro y Torres Arroba, 2018, p.21)
- Permite el escalamiento y la mantenibilidad de la aplicación. (Pazmiño Haro y Torres Arroba, 2018, p.21)
- Posee varias utilidades a incorporar dentro de la página web, entre las cuales se encuentran: obtener información del navegador, operar con objetos y vectores. (Villarreal Fuentes, 2013)
- Soporta extensiones. (Villarreal Fuentes, 2013)

2.10.4. HTML

HTML por sus siglas en inglés que en español significa “Lenguaje de Marcas de Hipertexto” (“Hiper Text Markup Language”), es un lenguaje de marcado de hipertexto para codificar aplicaciones web estáticas, donde los navegadores se encargan de leer e interpretan las etiquetas dentro del archivo de hipertexto para determinar cómo desplegar la página web. (Sánchez, 2012, citado en Rúales Pintado y Suarez Quispe, 2015)

Características

De acuerdo a (Valarezo Pardo et al., 2018, p.36), se establecen algunas características sobre HTML como son:

- No es un lenguaje de programación.
- Genera únicamente aplicaciones web estáticas.

- Se lo puede combinar con otros lenguajes de programación para añadir dinamismo a las aplicaciones web.
- La sintaxis de HTML5 está compuesta por HTML y XHTML (“Lenguaje de Marcas de Hipertexto Extendido”).

2.10.5. PostgreSQL

Es un sistema de base de datos relacionales orientado a objetos que tiene su código disponible a cualquier persona (Open Source). Emplea el lenguaje de programación SQL combinado, lo que le permite almacenar y escalar de manera segura una gran cantidad de datos. (PostgreSQL, 2020)

Según (PostgreSQL, 2020), este sistema de base de datos posee las siguientes características:

- Consiste en un gestor de bases de datos relacionales, soporta diferentes tipos de datos.
- Emplea un lenguaje SQL basado en el estándar ISO/IEC, permitiéndole importar consultas y código de scripts de otras bases de datos.
- Cumple con el modelo ACID, es decir, los datos almacenados cuentan con las características de atomicidad, consistencia, aislamiento y durabilidad.
- Ofrece opciones que otras bases de datos no tienen, como la posibilidad de crear tablas heredadas, esquemas o triggers, entre otros.
- No solo permite crear procedimientos en lenguaje SQL, sino que también permite utilizar otros lenguajes de programación como Python o Pearl.
- Ofrece una gran escalabilidad vertical.

Principales ventajas

- Instalación y uso gratuito.
- Compatibilidad con multiplataformas, ya que es adaptable a diferentes sistemas operativos y sobre diversos servidores web.
- Fácil y rápida configuración.
- Gran flexibilidad para desarrollar proyectos.
- Funciona con el estándar SQL.
- Tiene una alta fiabilidad y robustez.
- Posee una función de control de concurrencias multiversión o MVCC, permitiendo añadir a las transacciones una imagen del estado de la base de datos.

2.10.6. CSS

CSS, por sus siglas en inglés que en español significa “Hojas de Estilo en Cascada” (“Cascading Style Sheets”), comprende un lenguaje de hojas de estilo para controlar la manera en la que se presentan las páginas web en el navegador, es decir, da estilo a las páginas web, separando el

contenido de la presentación para mejorar el aspecto de las mismas y brindar una mayor reusabilidad de estilos, permitiendo gestionar estilos y formatos de múltiples páginas al mismo tiempo. (Eguiluz Pérez, 2009, p.5)

Características:

CSS presenta varias características como son:

- Es simple de aprender y aplicar.
- Controla documentos electrónicos HTML y XHTML.
- Permite un control centralizado del estilo de múltiples páginas web.
- Mejora la accesibilidad de la página web.
- Reduce la complejidad del mantenimiento.

2.10.7. NetBeans

Es un entorno integrado de desarrollo o IDE (Integrated Development Environment) en el que podemos realizar todas las tareas asociadas a la programación como: editar el código, compilarlo, ejecutarlo, depurarlo, entre otros, permitiendo el desarrollo de aplicaciones a partir de un conjunto de componentes de software llamados módulos. (Gimeno y Gonzales, 2011)

2.10.8. CLisp

CLisp es un lenguaje de programación de alto nivel dinámico y funcional orientado a objetos bajo la categoría de software libre, que implementa el lenguaje escrito en ANSI Common Lisp con varias extensiones. (CLISP, 2010)

El lenguaje CLisp es una variante maximalista del lenguaje Lisp el cual se maneja mediante listas, por lo cual es ampliamente empleado dentro de la inteligencia artificial (IA) para la creación de reglas de inferencia que permitan emular razonamiento humano.

2.10.9. Servicio web

Son sistemas software que permite la interacción e intercambio de información entre aplicaciones a través de la red, empleando protocolos de transporte para el intercambio de mensajes tales como HTTP o HTTPS. Esta comunicación entre aplicaciones está basada en los principios SOAP y RESTFUL, en donde las comunicaciones RESTFUL con el paso del tiempo han ido reemplazando a los servicios SOAP, debido a que brindan mejoras en términos de tráfico de red, latencia y procesamiento. (Mumbaikar y Padiya, 2013, p.1-4)

➤ *Servicio web RESTFUL*

Los servicios web REST (Transferencia de estado representacional) es un nuevo enfoque más ligero que SOAP, el cual se basa en recursos autodefinidos que emplean el protocolo HTTP para transmitir datos que están formados por formatos XML, JSON, entre otros. Simplifica el acceso

a servicios web empleando estándares existentes y bien conocidos. Un servicio se proporciona como un recurso que puede identificarse mediante URI (Identificador uniforme de recursos) (Tihomirovs y Grabis, 2016, p.92-97).

Características

(Malik y Kim, 2017, p.754) menciona algunas características de los servicios web REST como:

- Las operaciones principales que emplea son: crear, leer actualizar y borrar.
- Emplea URI y HTTP para exponer la lógica del negocio.
- Es una arquitectura orientada a recursos.
- Menor requisitos de ancho de banda y recursos que SOAP.
- Proporciona más flexibilidad y menos gastos generales.
- No requiere un formato de mensaje.
- Todas las transacciones son independientes.

Principios de diseño

Debido a que los servicios REST no poseen un estándar oficial, este provee unos conceptos claves conocidos como “principios de diseño” para el desarrollo de servicios basados en la web (Paik et al., 2017, 67-91). Los principales principios de diseño son:

- **Identificación de recursos:** todos los recursos se identifican de manera única mediante el uso de un URI.
- **Interfaz de recursos unificada:** todos los recursos son accesibles a través de un conjunto de operaciones HTTP.
- **Links e hipermedia:** los recursos se pueden enlazar a través de varios tipos de enlaces de relación.

Ventajas y desventajas

Tabla 11-2: Ventajas y desventajas del servicio web RESTful

Ventajas	Desventajas
<ul style="list-style-type: none"> • Consumo bajo de recursos. • Las instancias del proceso son creadas de manera explícita. • El cliente no requiere información de enrutamiento a partir de la URI inicial. • Los clientes pueden tener una interfaz genérica para las notificaciones. • Por lo general, es fácil de construir y adoptar. 	<ul style="list-style-type: none"> • Ocupa un gran número de objetos. • El manejo de los espacios de nombres (URIs) puede ser complicado. • Emplea una descripción sintáctica / semántica muy informal. • Presenta pocas herramientas de desarrollo.

Fuente: (Meneses, 2017)

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

2.10.10. Frameworks front-end

Dentro del entorno de desarrollo de software existen una gran variedad de frameworks que brindan soporte en distintas áreas del proceso de desarrollo, entre los cuales se encuentran aquellos que brindan soporte al diseño de interfaces ricas relacionadas a JavaServerFaces (JSF), denominados frameworks front-end, en los cuales destacan: Apache Tobago, PrimeFaces y RichFaces.

➤ Comparativa de frameworks front-end

Existen varios frameworks front-end que pueden brindar soporte dentro del desarrollo de un producto software. En este caso, se presenta una comparativa entre los frameworks JSF Apache Tobago, PrimeFaces y RichFaces en la **Tabla 12-2**.

Tabla 12-2: Comparativa de frameworks front-end

Parámetro	Framework		
	Apache Tobago	PrimeFaces	RichFaces
Componentes	31	140	71
Documentación y soporte	Escasa	Abundante	Abundante
Frecuencia de actualización	Pocas correcciones y dudas. Disponible compatibilidad con JSF 2.0 desde la versión 1.5.x	Gran comunidad de desarrolladores y desde sus primeras versiones compatible con JSF 2.0	Gran contribución de los desarrolladores. Desde la versión 4.0 es compatible con JSF 2.0
Configuraciones para el uso	<ul style="list-style-type: none">➤ 19 bibliotecas➤ Tamaño de bibliotecas 3.2 MB➤ Requiere configuración de web.xml	<ul style="list-style-type: none">➤ 1 biblioteca➤ Tamaño de bibliotecas 1.7 MB➤ No requiere configuración de web.xml	<ul style="list-style-type: none">➤ 8 bibliotecas➤ Tamaño de bibliotecas 8.2 MB➤ Requiere configuración de web.xml
Requerimientos de funcionamiento	Pocos servidores compatibles	Se ejecuta en cualquier servidor que admita JSF 1.2 y 2.0	Gran cantidad de servidores y navegadores compatibles

Fuente: (Carmisini y Vahldick, 2012, p.10-18)

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En la **Tabla 12-2**, se presenta un análisis comparativo entre los frameworks JSF realizado por (Carmisini y Vahldick, 2012, p.10-18) donde se analizaron los parámetros: componentes, documentación y

soporte, frecuencia de actualización, configuración para el uso y requerimientos de funcionamiento; los cuales dan a conocer que, Apache Tobago posee pocos componentes en comparación a los otros dos frameworks, escasa documentación, soporte y servidores compatibles que, añadido a su alta cantidad de bibliotecas dificultan su uso y configuración. RichFaces posee más componentes que Tobago, pero también cuenta con varias bibliotecas además de documentación y soporte abundante. PrimeFaces posee la mayor cantidad de componentes, una comunidad amplia y activa con documentación extensa, altamente compatible y práctico al tener una sola biblioteca que facilita su manejo. Por las razones antes mencionadas, se eligió a PrimeFaces como la alternativa más óptima a emplear en el desarrollo del presente trabajo.

➤ *JavaServerFaces*

La tecnología JavaServerFaces (JSF) es un marco de trabajo de interfaces de usuario formado por una amplia lista de componentes ejecutadas en el lado del servidor, cuyo objetivo principal es facilitar y simplificar el diseño de la interfaz.

JSF tiene una clara separación entre el diseño de interfaces con archivos XHTML y CSS de las clases que se encargan del resto de funcionalidades del sistema. Por esto, JSF está basado en el paradigma de programación MVC, ya que permite la ejecución de los procedimientos java mediante la conexión con los Beans. (Hinojosa, 2013, p.1-21)

Según (Cuevas, 2009, p.1) las características principales son:

- Clara separación entre la vista y el modelo.
- Desarrollo basado en componentes y no es peticiones.
- Las acciones de usuario se ligan muy fácilmente al código del servidor.
- Creación de familias de componentes visuales para acelerar el desarrollo.

➤ *Arquitectura JSF*

JSF sigue un modelo MVC que permite separar la capa de lógica de control, lógica de negocio, y la lógica de presentación, consiguiendo un diseño más refinado que permitirá facilitar el uso y mantenimiento. En la **Figura 2-2** se puede visualizar gráficamente la arquitectura que utiliza JSF.

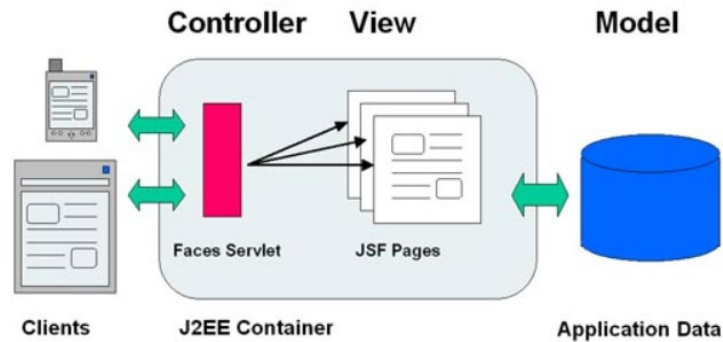


Figura 2-2: Implementación de JavaServer Faces de MVC

Fuente: (JavaServer Faces Technology, 2006)

En la **Figura 2-2** se visualiza las 3 capas de la arquitectura que son:

- **Modelo:** Se encarga de representar, gestionar y controlar los datos obtenidos de una base de datos, siendo totalmente independiente de la vista y el controlador, el modelo se representa con los Beans, mismos que deben estar definidos en el archivo faces-config.xml.
- **Vista:** Es la forma como presenta los datos obtenidos en el modelo, interrelacionado a este mediante referencias al mismo. Las páginas que se puede ver aquí son: .jsp, .jsf, .xhtml.
- **Controlador:** Gestiona la interacción entre usuario y la aplicación dando así significado a las peticiones del usuario desde la vista, gestionando las acciones que se va a realizar sobre el modelo. Una aplicación JSF requiere un servlet, conocido como Faces Servlet el cual actúa como controlador encontrado en el archivo web.xml.

➤ *Ciclo de vida de JSF*

Son las fases que pasa la petición JSF desde que se recibe en el servidor hasta que se genere la página HTML. Las fases según (Depto. CCIA, 2010, p.4) son las siguientes:

1. **Restaurar la vista (restore view).** Obtiene el árbol de componentes correspondiente a la vista JSF de la petición, verificando si se ha generado, en cuyo caso se recupera, caso contrario, si es la primera vez que el usuario visita la página, partiendo de la descripción JSF se lo genera.
2. **Aplicar los valores de la petición (apply request values).** Una vez se ha generado el árbol de componentes, se procede a procesar todos los valores asociados al mismo, convirtiendo a tipos de datos Java todos los datos de la petición y para aquellos que tienen la propiedad immediate a cierta, se validan, adelantándose a la siguiente fase.
3. **Procesar las validaciones (process validations).** Valida todos los datos, y en caso de existir algún error, se termina el ciclo de vida con un encolamiento de mensaje de error, dando paso a la renderización de la respuesta.

4. **Actualizar los valores del modelo (update model values).** Para esta fase, ya se han procesado y validado todos los valores involucrados.
5. **Invocar a la aplicación (invoke application).** En esta fase ya se encuentran actualizadas todas las propiedades de los beans asociados a componentes de entrada (input), procediendo a llamar a la acción indicada por el usuario.
6. **Renderizar la respuesta (render response).**

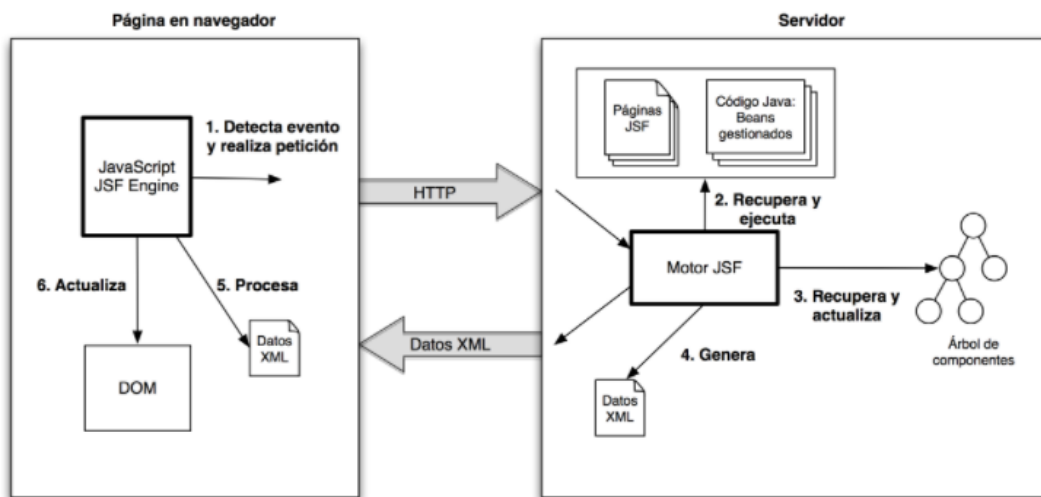


Figura 3-2: Ciclo de vida de JSF

Fuente: (Depto. CCIA, 2010, p.4)

➤ *Framework PrimeFaces*

Desarrollado por Prime Technology, comprende un conjunto de componentes JSF (Java Server Faces) de código abierto que incluye diversas extensiones. (Primefaces, 2020)

La principal diferencia de PrimeFaces frente a otras bibliotecas es que, permite ver el proyecto desde el punto de vista de los desarrolladores de tal manera que las correcciones y ajustes a la biblioteca se realizan de manera más oportuna. (Primefaces, 2020)

Características

Las principales características de PrimeFaces son:

- Posee un potente conjunto de componentes (HTML Editor, Dialog, AutoComplete, Gráficos y muchos más).
- Incorpora Ajax basado en API Ajax de JSF estándar.
- Es muy ligero, y no requiere realizar complejas configuraciones ni agregar dependencias.
- Soporte a través de Atmosphere Framework.
- Tiene Skinning Framework con temas variados incluidos y soporte para la herramienta de diseño visual.
- Integra temas y diseños para aplicaciones de tipo premium.

- Cuenta con una amplia documentación para los desarrolladores.

2.11. Sistema recomendador

Los sistemas recomendadores (SR) son herramientas que determinan criterios, preferencias, necesidades e indicaciones dadas por los usuarios, proporcionando sugerencias sobre ítems más apropiados (Vera y Ulloa, 2018, p.46-47). Por lo general, en los SR se maneja un elemento específico a ser recomendado (denominado Ítem), por ejemplo, e-commerce de viajes o de compras, un repositorio de material educativos, entre otros (Charnelli, 2018, p.47).

Los SR funcionan en dos etapas: la primera es el aprendizaje de lo que le gusta al usuario, en donde se hace una recopilación de información del usuario para así tener un perfil personalizado; y en la segunda etapa se hace la recomendación en sí. El aprendizaje en un SR se puede obtener de dos formas: la retroalimentación explícita, donde el usuario es quien da información sobre sus intereses en los ítems, y la retroalimentación implícita, en donde la recopilación de la información lo efectúan los SR mediante la observación del comportamiento de los usuarios. (Vicuña Lojano, 2018, p.2-7).

Existe una variedad de SR dependiendo del ítem que recomienden y según a los usuarios que vaya dirigidos, pero todos deben cumplir con las siguientes características según lo mencionado en (Vicuña Lojano, 2018, p.2-7):

- Existe un conjunto de usuarios a los que realizar las recomendaciones.
- Se dispone de un conjunto de ítems para recomendar.

2.11.1. Características de los sistemas recomendadores

Según (Matos, 2018), los sistemas recomendadores se caracterizan por:

- Tener autonomía de tomar decisiones en base a la información disponible.
- Requerir un método para la filtración de información sobre los ítems.
- Ser aplicados en algún dominio específico.
- Utilizar motor de recomendación (también conocido como “técnica”).

2.11.2. Beneficio de los sistemas recomendadores

- Mejor satisfacción en los usuarios
- El usuario lograra tener una experiencia más interesante
- Ayudan a obtener datos para análisis estadísticos

2.11.3. Funcionamiento de los sistemas recomendadores

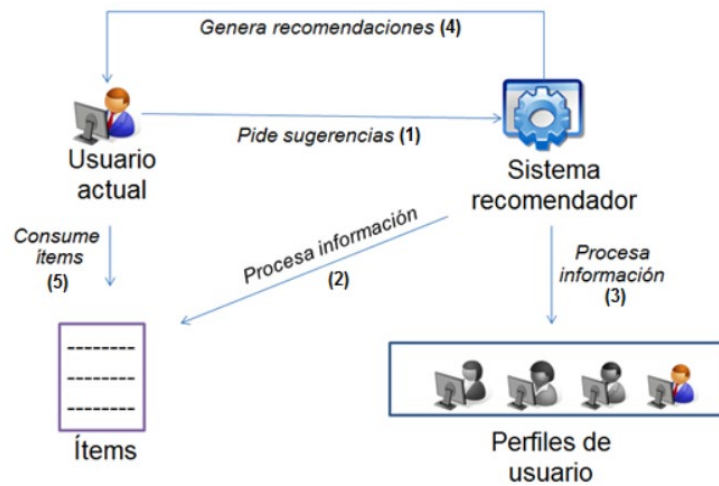


Figura 4-2: Esquema de funcionamiento de un sistema recomendador

Fuente: (Almaguer et al., 2021, p.56)

La **Figura 4-2** detalla cómo funciona de manera general un sistema recomendador en donde para iniciar el un usuario tiene que elegir un ítem adecuado para él, para lo cual solicita sugerencias al sistema, mismo que procesa la información asociando a las características de los ítems y perfiles de usuario, para después recomendar ítems más apropiados, el usuario podrá elegir basándose en estas recomendaciones.

2.11.4. Clasificación de los sistemas recomendadores

Se puede clasificar a los sistemas recomendadores en cinco categorías basándose en la técnica de recomendación empleada para identificar de manera eficaz la información, a la cual desea el usuario acceder en base a sus datos ingresados. A continuación, se detalla según (Sari Villa, 2017, p.7-9):

➤ **Recomendadores basados en el contenido (RBC)**

En este sistema las sugerencias de los ítems se realizan en base a las deducciones sobre las necesidades de los usuarios y sus preferencias, entendiéndose que, la información que se tiene sobre un ítem responde a la necesidad particular del usuario para así poder dar la recomendación.

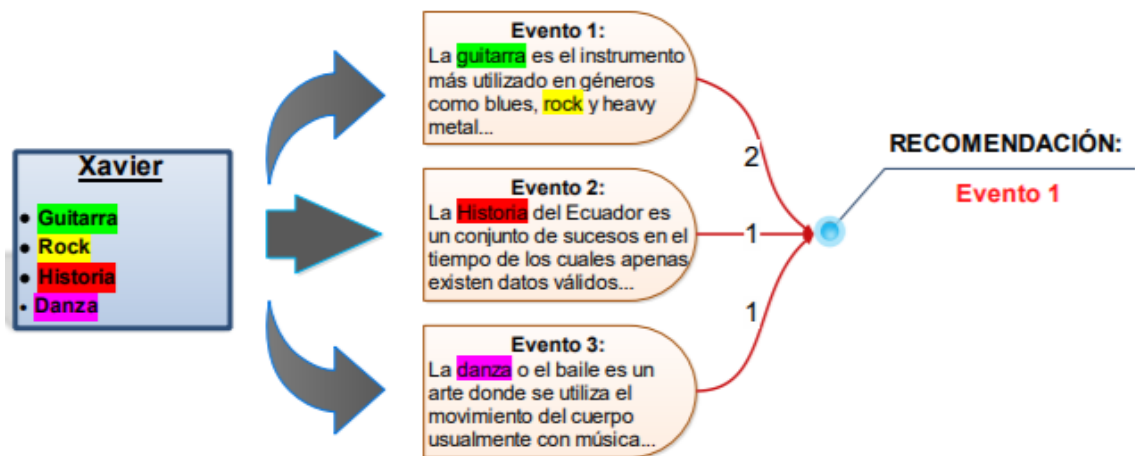


Figura 5-2: Sistema recomendador basado en contenido

Fuente: (Sari Villa, 2017, p.7-9)

La fuente principal de información de este sistema recomendador es el perfil del usuario, en donde se utiliza los datos del usuario para relacionarlo con un ítem, que cumpla todas las características del perfil.

➤ **Recomendadores basados en filtrado colaborativo (BFC)**

Este sistema usa la información que se obtiene de una comunidad de usuarios para determinar si varios usuarios tienen un mismo interés en eventos pasados, pueda hacer una recomendación futura.

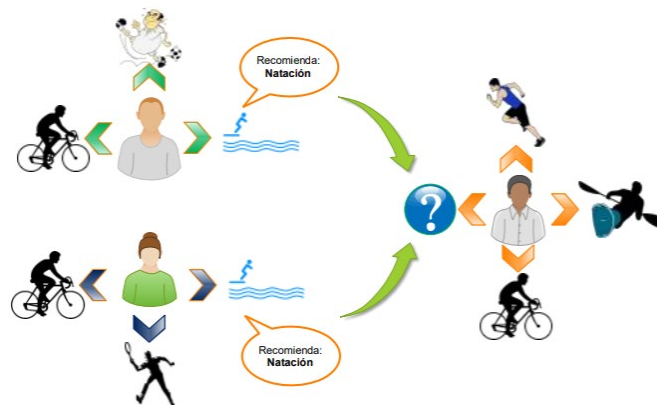


Figura 6-2: Sistema recomendador colaborativo

Fuente: (Sari Villa, 2017, p.7-9)

➤ **Recomendadores con filtro demográfico (RFD)**

Este sistema realiza la recomendación de una forma similar al sistema basado en el contenido, con la diferencia que, en vez de usar la información de los ítems lo hace con información demográfica del usuario.



Figura 7-2: Sistema recomendador colaborativo

Fuente: (Sari Villa, 2017, p.7-9)

➤ **Recomendadores con filtrado híbrido (RFH)**

Este sistema es el resultado de una combinación de diferentes SR para poder realizar recomendaciones a los usuarios, por lo que es más complejo de construir, pero es muy eficiente al mostrar los resultados.

2.11.5. Arquitectura de un Sistema Recomendador

La mayoría de los sistemas recomendadores según (Matos, 2018), tienen una arquitectura como se muestra en la **Figura 8-2**, la cual es una de las más generales.

Arquitectura del SR

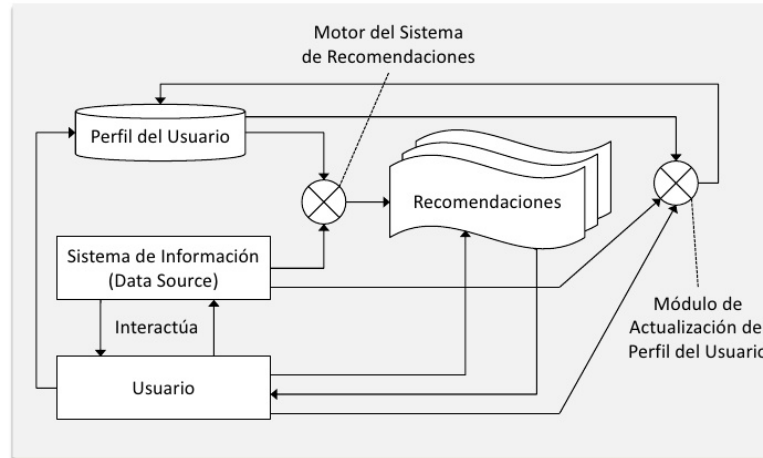


Figura 8-2: Arquitectura del SR

Fuente: (Matos, 2018, p.1)

En estos sistemas un usuario interactúa con un conjunto de datos (sistema de información), para definir un perfil de usuario y con estas dos combinaciones poder buscar coincidencias que permitan hacer recomendaciones mediante el motor de recomendaciones.

CAPITULO III

3. MARCO METODOLÓGICO

El presente capítulo contiene una descripción de los métodos y técnicas utilizados para el desarrollo y cumplimiento de los objetivos del presente trabajo denominado “Desarrollo de un sistema recomendador de contenidos educativos basado en estilos de aprendizaje aplicado a la Escuela de Ingeniería en Sistemas”, el cual tiene como propósito brindar soporte en el proceso enseñanza – aprendizaje con el fin de mejorar el rendimiento académico de los estudiantes.

3.1. Tipo de estudio

El presente trabajo es un estudio de tipo aplicativo donde haciendo uso de diversos conocimientos, herramientas y técnicas se desarrolla el sistema recomendador “Custom Learning”, el cual está dirigido a brindar soporte a los estudiantes en su proceso de aprendizaje basándose en los estilos de aprendizaje como criterio de recomendación.

3.2. Métodos y técnicas de investigación

Los métodos y técnicas empleados se detallan en base a los objetivos propuestos en el presente trabajo, tal como se indica en la **Tabla 13-3**.

Tabla 13-3: Métodos y técnicas

Objetivos	Método	Descripción	Técnicas	Fuentes
Caracterizar los sistemas recomendadores para aplicarlos en el diseño de contenidos educativos mediante revisión de literatura.	Analítico	Con el fin de poder caracterizar los sistemas recomendadores se emplea el método analítico a través de la técnica de lectura y síntesis, para obtener la información necesaria que permita el desarrollo del presente objetivo.	<ul style="list-style-type: none">• Revisión de documentación a través de la web.• Síntesis.	<ul style="list-style-type: none">• Libros• Artículos científicos
Analizar los estilos de aprendizaje que permitan definir los lineamientos para el desarrollo de contenidos educativos	Analítico, estadístico.	Con el fin de poder analizar los estilos de aprendizaje se emplea un método analítico y el método estadístico, para llevar a cabo un estudio preliminar que permita establecer la información necesaria sobre la preferencia de formato de	<ul style="list-style-type: none">• Revisión de documentación a través de la web.• Síntesis• Encuesta y test online (https://testkolb.000webhostapp.com/TestKolb/)• Anexo F	<ul style="list-style-type: none">• Libros• Artículos científicos• Estudiantes

Objetivos	Método	Descripción	Técnicas	Fuentes
		contenido con el estilo de aprendizaje, para el posterior desarrollo del sistema/, para lo cual se manejó un análisis e interpretación de datos cuantitativos y cualitativos.	<ul style="list-style-type: none"> • Anexo G 	
Desarrollar los módulos del sistema recomendador usando el framework PrimeFaces aplicado a la metodología XP.	Metodología XP	Es una metodología ágil que permite el desarrollo y gestión del proyecto con eficiencia, facilidad y flexibilidad, siendo muy adaptable al cambio y promoviendo el trabajo en equipo.	<ul style="list-style-type: none"> • Diagrama de caso de uso • Diagrama de componentes • Diagrama de procesos • Historias de usuario • Metáforas del sistema • Tarjetas CRC • Prueba de aceptación • Diagrama de despliegue 	<ul style="list-style-type: none"> • Docente • Artículos científicos
Evaluar el rendimiento académico de los estudiantes, que han aplicado los contenidos educativos personalizados del sistema recomendador.	Estadístico	Para el presente objetivo se empleó como técnica de recolección de datos una evaluación online compuesta por 20 preguntas de selección multiple valorada en 20 puntos (1 punto por cada pregunta) para posteriormente poder analizar los resultados de manera descriptiva e inferencial.	<ul style="list-style-type: none"> • Prueba semiestructurada online (https://testkolb.000webhostapp.com/) Anexo H. • Prueba de normalidad de Shapiro Wilk • Prueba t-student para muestras independientes. 	<ul style="list-style-type: none"> • Estudiantes

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

3.3. Etapas del desarrollo del trabajo

El desarrollo del presente trabajo se dividió en dos etapas, donde la primera corresponde a un estudio preliminar enfocado en determinar la preferencia de contenidos en base a los estilos de aprendizaje de los estudiantes, información que servirá de base para el desarrollo del sistema recomendador “Custom Learning” y los contenidos a aplicar en el mismo, y la segunda etapa se refiere al análisis del rendimiento académico de los estudiantes que han aplicado los contenidos educativos personalizados del sistema recomendador “Custom Learning”.

3.3.1. Etapa 1: Determinar la preferencia de contenidos

Esta etapa comprende un estudio preliminar desarrollado en el presente trabajo con la finalidad de obtener la información necesaria que servirá de base para efectuar la recomendación de contenido por parte sistema recomendador.

a. Determinación de los formatos de contenido

Para la aplicación del sistema hacia los estudiantes es necesario contar con contenido en diversos formatos dentro del sistema, por lo cual se requiere determinar que formatos de contenido manejará el sistema y cuál de ellos se presentará respectivamente a los estudiantes según su estilo de aprendizaje, para lo cual se emplea como herramienta de recolección de datos el test de estilos de aprendizaje propuesto por David Kolb y un test que permita determinar las preferencias de formatos de contenidos educativos.

Como base de desarrollo de la presente investigación se tomaron los resultados la investigación realizada por (Sahabudin y Bilal, 2013, p.715), pero, con la diferencia de que se tomó en cuenta 8 tipos de contenido que son: texto, imagen, audio, video, objeto de aprendizaje (OA), animación, simulación y juego,

Una vez definidos los formatos de contenidos a tomar en cuenta se procedió a redactar las preguntas de la encuesta que permitirá determinar la preferencia del estudiante con respecto a dichos formatos de contenidos. Entre las preguntas que se redactaron para la encuesta esta:

“¿Entre novelas y comics, cual prefiere?”

- a) novelas
- b) comics

Esta pregunta fue realizada con el fin de determinar si los estudiantes prefieren texto o imágenes. Otra pregunta dentro de la encuesta fue:

“Usted ha comprado un nuevo aparato electrónico, pero no sabe cómo manejarlo ¿Cómo preferiría aprender a manejar dicho aparato, viendo imágenes descriptivas sobre el funcionamiento del mismo en el manual de usuario o buscando un tutorial en YouTube sobre dicho funcionamiento?”

- a) Viendo imágenes descriptivas del funcionamiento
- b) Ver el tutorial de YouTube sobre el funcionamiento

De igual manera esta pregunta tenía como finalidad descubrir si los estudiantes prefieren contenido a manera de imágenes o videos.

De esta forma se llevaron a cabo las preguntas dentro de la encuesta, teniendo un total de 18 preguntas, donde la última pregunta comprendía una pregunta de control en donde se le indicaba que seleccione de entre los 8 tipos de contenido aquel al que tenga mayor preferencia., para de esta manera corroborar los datos obtenidos.

b. *Determinación de la población y muestra*

Una vez establecidas las herramientas para la recolección de datos, se procede a establecer el ambiente donde se las aplicara, para lo cual se parte por determinar la población del estudio, conformada por un aproximado de 550 estudiantes de la carrera de ingeniería de software de la ESPOCH, valor del cual partimos para el cálculo de la muestra aplicando la siguiente formula:

$$\text{Tamaño de la muestra} = \frac{\frac{z^2 * p(1 - p)}{e^2}}{1 + \left(\frac{z^2 * p(1 - p)}{e^2 N}\right)}$$

N = tamaño de la población

e = margen de error (10% = 0.1)

z = nivel de confianza (99% = 2.58)

$$\text{Tamaño de la muestra} = \frac{\frac{2.58^2 * 0.5(1 - 0.5)}{0.1^2}}{1 + \left(\frac{2.58^2 * 0.5(1 - 0.5)}{0.1^2 550}\right)}$$

Tamaño de la muestra = 127.75

Tamaño de la muestra = 128

Al aplicar la fórmula de muestreo con un margen de error del 10%, se obtuvo un valor para la muestra de 128 personas.

Se aplicó el test y la encuesta a los 128 estudiantes para obtener los datos, pero, se trabajó únicamente con 113, al descartar a aquellos que no cumplían con el parámetro de rango de tiempo establecido (5 minutos – 1 hora) para la resolución de estas herramientas de recolección de información, con el fin de tener mayor veracidad de los datos recolectados.

En la **Tabla 14-3** se muestra la información con respecto al semestre al que pertenecen los estudiantes que participaron de la presente recolección de datos.

Tabla 14-3: Análisis demográfico de los datos para estudio de estilo de aprendizaje.

Semestre	Frecuencia	Porcentaje (%)
Primero (1ro)	39	30.47
Segundo (2do)	0	0
Tercero (3ro)	23	17.97
Cuarto (4to)	12	9.38
Quinto (5to)	6	4.69
Sexto (6to)	37	28.90
Séptimo (7mo)	0	0
Octavo (8vo)	6	4.69
Noveno (9no)	3	2.34

Semestre	Frecuencia	Porcentaje (%)
Otro	2	1.56
Total	128	100

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Como se observa en la **Tabla 14-3**, la mayoría de los estudiantes pertenecen al primer semestre con un total de 39 estudiantes (30.47%) de la carrera de ingeniería en software de la ESPOCH, seguido por el sexto semestre con un total de 37 estudiantes (28.9%), mientras que de segundo y séptimo semestre no se contó con estudiantes para este análisis.

c. Determinación de estilos de aprendizaje y preferencia de contenido

Para determinar los estilos de aprendizaje se procedió a emplear como herramienta de recolección de datos el test propuesto por David Kolb. Para la preferencia de contenido de dichos estilos se utilizó una encuesta compuesta por 18 preguntas de selección múltiple que se determinó en el paso anterior. Dichas herramientas se las aplicó de manera online (<https://testkolb.000webhostapp.com/TestKolb>), tal como se muestra en la **Figura 9-3**.

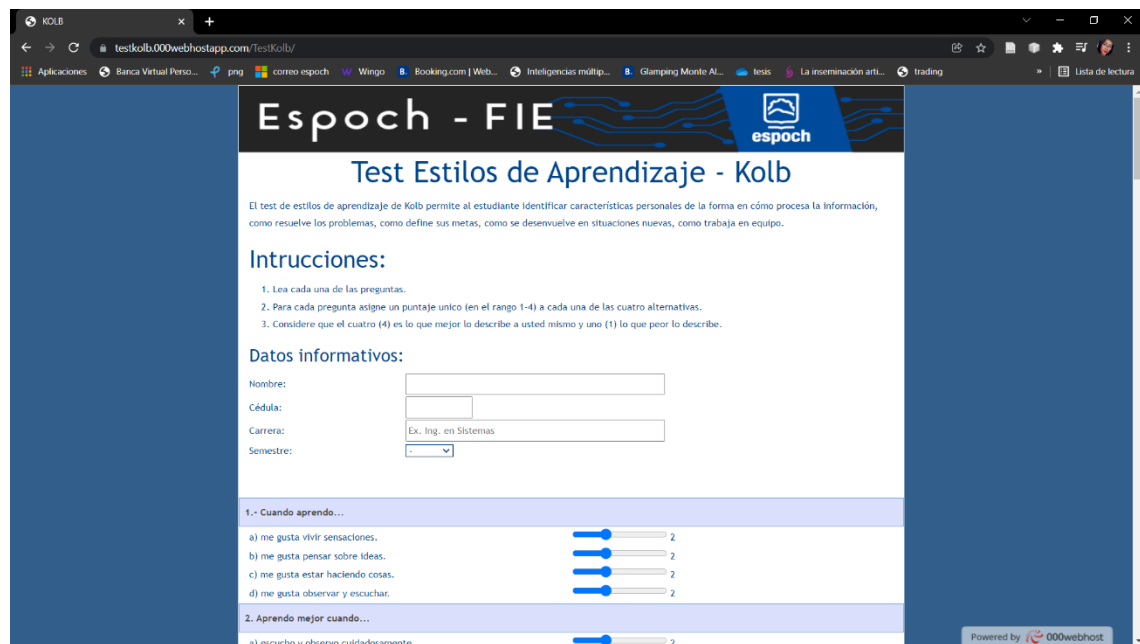


Figura 9-3: Herramienta online para estudio preliminar

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

3.3.2. Etapa 2: Desarrollo del sistema recomendador

Para el desarrollo del sistema recomendador propuesto en el presente trabajo de titulación se seguirán las diversas fases propuestas en la metodología XP, tal como se detalla a continuación.

a. *Fase de exploración*

Se desarrolla con el fin de tener una idea preliminar sobre el desarrollo del proyecto, recopilando información sobre roles, requerimientos, procesos y especificaciones del sistema.

➤ Roles del proyecto

De acuerdo a la metodología XP se definen diversos roles encargados de desarrollar tareas específicas con el fin de llevar a cabo el proyecto de manera adecuada y eficiente. En la **Tabla 15-3** se describen los roles para el desarrollo del sistema.

Tabla 15-3: Roles del proyecto

Rol	Encargado
Equipo de desarrollo	Kevin Paúl Jácome Niama Dayana Elizabeth Almache López
Cliente	Danilo Pastor
Consultor	Diego Ávila

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Procesos del Sistema Recomendador de Contenido

Los procesos a desarrollar dentro del sistema recomendador se detallan a través de diagramas de secuencia, tal como se muestra a continuación.

- Proceso de obtener el estilo de aprendizaje de los estudiantes

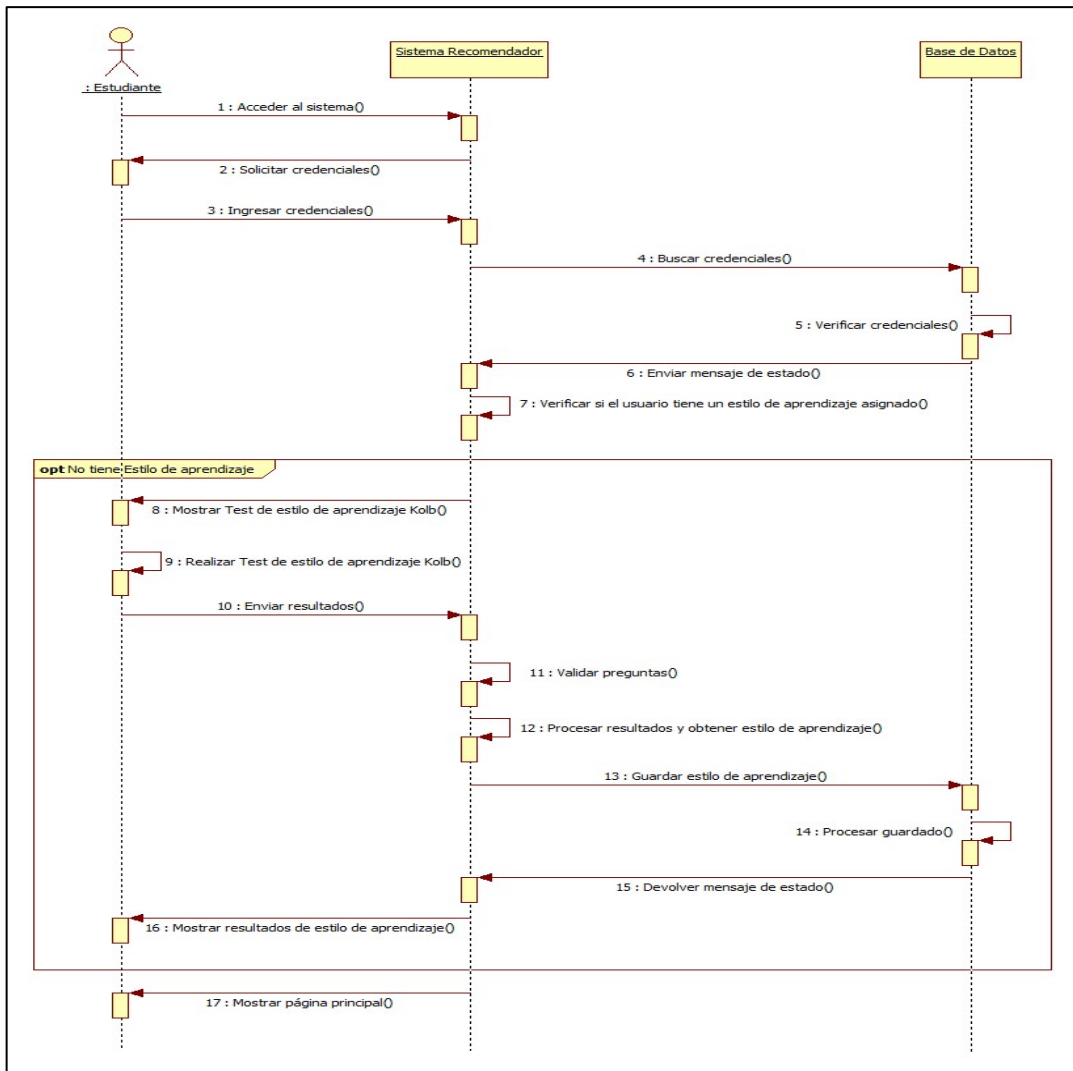


Figura 10-3: Diagrama de secuencia “Obtener estilo de aprendizaje del estudiante”

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En la **Figura 10-3** se muestra el flujo de la secuencia de obtención de estilos de aprendizaje de los estudiantes, el cual inicia con el ingreso al sistema, donde se solicitará que se autentifique con las credenciales respectivas para proceder a verificarlas con el fin de permitir o negar el acceso. Una vez ingresado al sistema, se verifica si el estudiante no tiene un estilo de aprendizaje asignado, en cuyo caso se mostrará la interfaz del test de estilo de aprendizaje de Kolb, donde el estudiante lo realizará y enviará, entonces el sistema validará y procesará los resultados para obtener el estilo de aprendizaje respectivo, mismo que se guardará en la base de datos devolviendo y mostrando un mensaje de estado.

- Proceso de crear contenido.

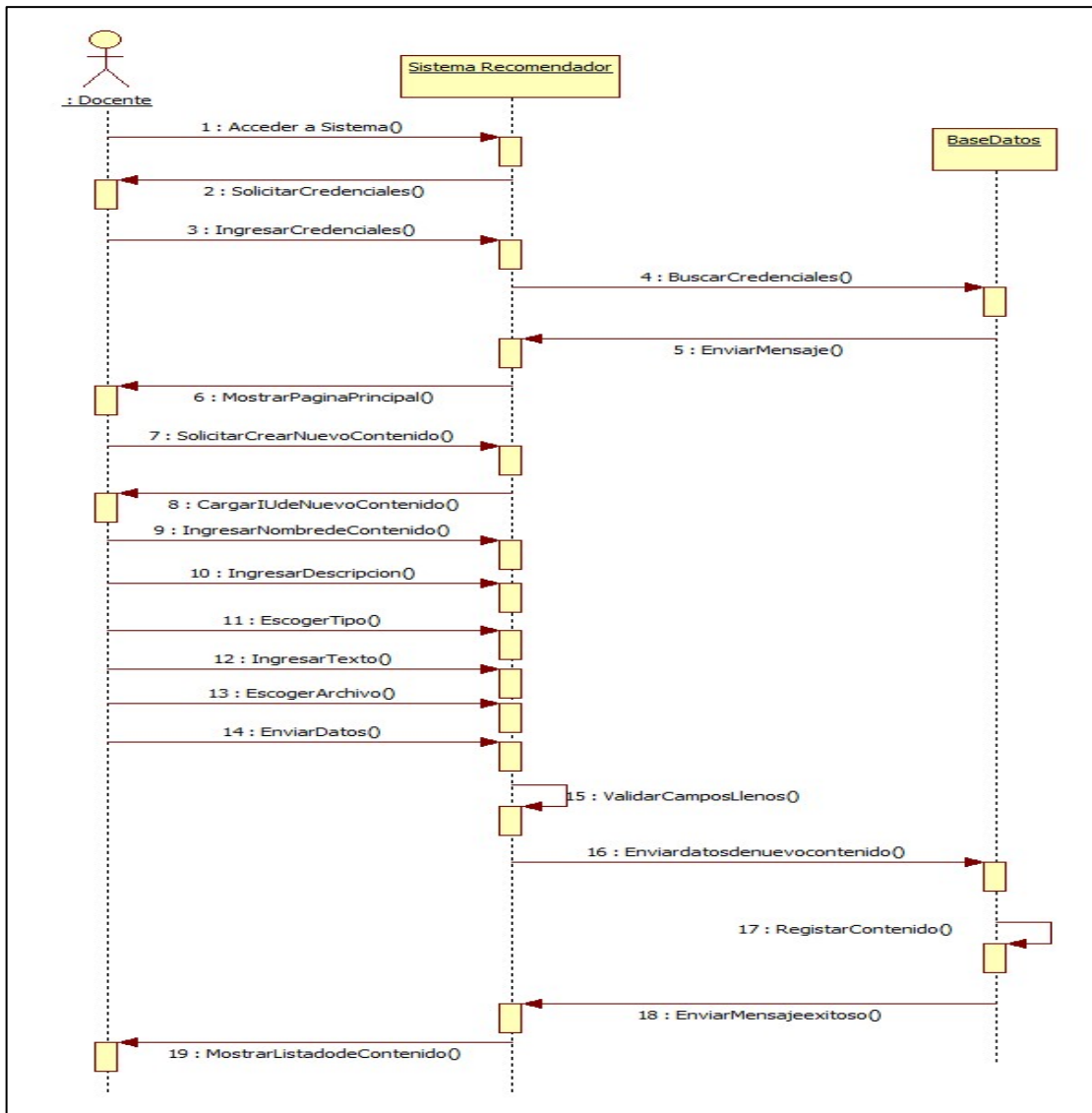


Figura 11-3: Diagrama de secuencia “Creación de contenido”

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

La **Figura 11-3** muestra el proceso de creación de contenido en el sistema recomendador, el cual empieza con el acceso del docente al sistema, a través de la autenticación con las credenciales respectivas, mismas que, una vez ingresadas se verificarán para permitir o negar el acceso, donde en caso de acceder se visualizará la página principal del SRC. Se accederá al panel de contenido para crear nuevo contenido, donde el sistema mostrará la interfaz respectiva, para poder ingresar el nombre del contenido, descripción, tipo, texto, y archivos adjuntos. El sistema valida que los campos obligatorios se encuentren completos, entonces se procede a registrar en la base de datos el nuevo contenido, obteniendo un mensaje de estado de registro del contenido, y redirigiendo a la pantalla que muestra el listado con todos los contenidos registrados.

- Proceso de recomendar contenido

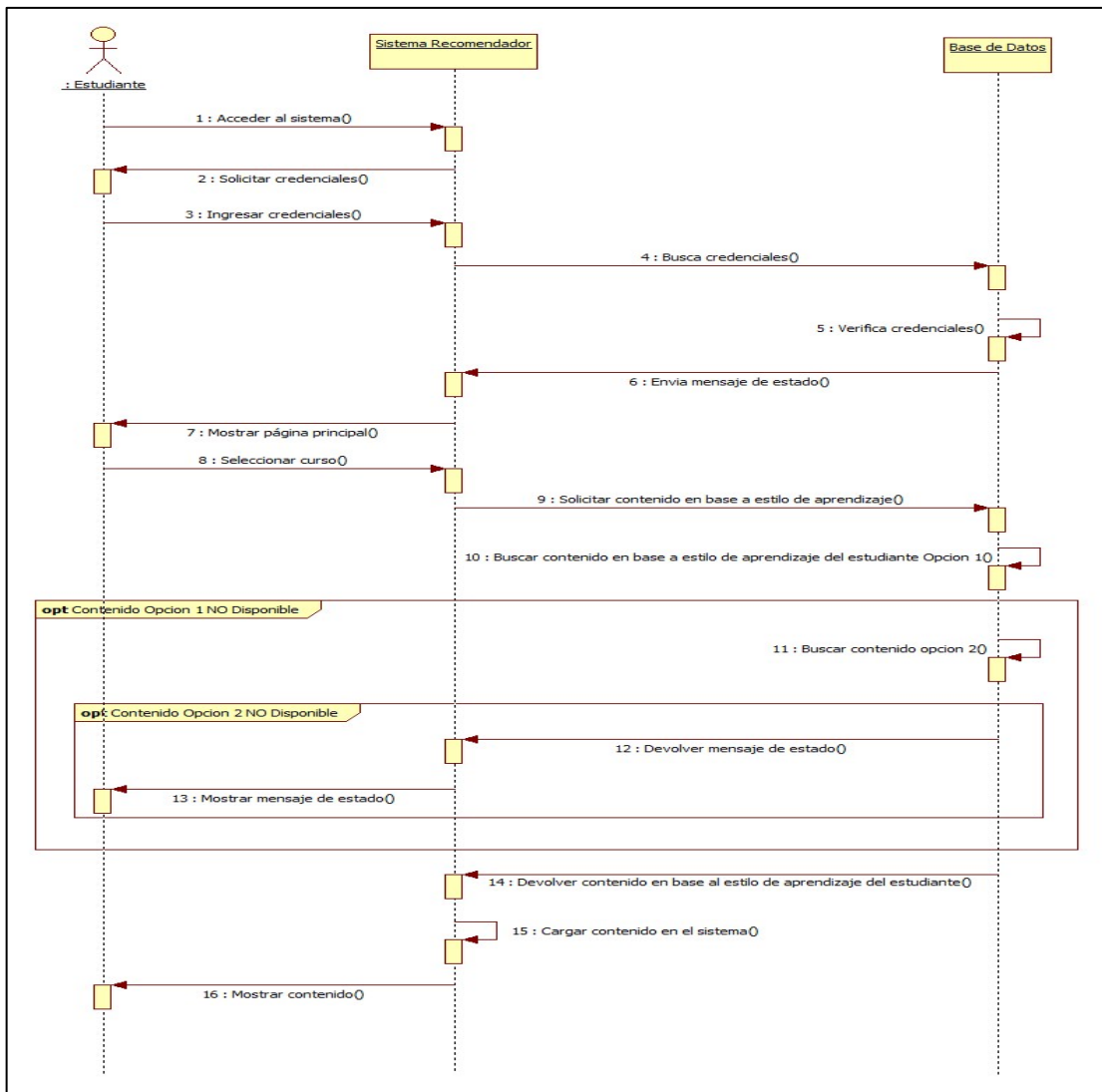


Figura 12-3: Diagrama de secuencia “Recomendar contenido”

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

La **Figura 12-3** muestra el proceso de recomendación de contenido dentro del sistema recomendador. Dicho proceso empieza con el acceso del estudiante al sistema, para lo cual se solicitará las credenciales respectivas para conceder el acceso. Una vez dentro del sistema, el estudiante ingresará a una materia en específico, donde el sistema buscará y mostrará el contenido en base al estilo de aprendizaje, si no existe el tipo correspondiente de este estilo, el sistema hace una búsqueda del contenido auxiliar. En caso de que no se encuentre ningún contenido que cumpla los criterios de búsqueda se mostrara un mensaje de estado sobre dicha situación.

➤ Especificación de usuarios

El sistema recomendador de contenido cuenta con módulos, actividades y tareas específicas para cierto tipo de usuarios, tal como se detalla en la **Tabla 16-3**.

Tabla 16-3: Tipos de usuarios

Tipo de usuario	Descripción	Rol	Responsable
Administrador	Responsable de administrar el sistema.	Es el encargado de administrar toda la información del sistema, ya sea información de docente, estudiante, materia, estilo de aprendizaje, contenidos.	Kevin Jácome Dayana Almache
Profesor	Docente responsable de la gestión del curso.	Gestiona el contenido (cargar, modificar, eliminar) de los cursos que tenga asignados.	Danilo Pastor
Estudiante de control	Usuario registrado dentro del sistema para acceder a los contenidos no personalizados.	Ingresa al sistema y visualiza el contenido en formato tradicional respectivo a su materia.	Estudiantes del paralelo "A"
Estudiantes de experimento	Usuario registrado dentro del sistema para acceder a los contenidos personalizados.	Realiza el test de EA al registrarse para poder acceder a las materias en las que se encuentre matriculado para visualizar el contenido respectivo.	Estudiantes del paralelo "B"

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En base a la **Tabla 16-3**, se muestra que dentro del sistema a desarrollar se va a constar con cuatro roles principales que les permitirá interactuar con el sistema de diferente manera, siendo los principales los estudiantes de control y experimento, mismos que interactuaran con el contenido con el fin de determinar si mejora o no su rendimiento académico.

➤ **Especificación de requerimientos**

Se obtuvo la especificación de requerimientos a través del análisis de los procesos del SRC y reuniones con el equipo de desarrollo y el cliente, obteniendo los siguientes requisitos funcionales y no funcionales para el sistema.

Requerimientos Funcionales

A continuación, se presenta los requerimientos funcionales según los módulos del sistema:

Módulo de Usuarios

- El sistema permitirá autorizar acceso de usuarios al sistema
- El sistema permitirá ingresar nuevos usuarios
- El sistema permitirá editar usuarios
- El sistema permitirá eliminar usuarios
- El sistema permitirá listar usuarios
- El sistema permitirá listar usuarios por estudiante

- El sistema permitirá listar usuarios por docente
- El sistema permitirá visualizar usuario
- El sistema permitirá asignar semestre a usuarios estudiantes
- El sistema permitirá acceder al test de Kolb
- El sistema permitirá obtener estilo de aprendizaje
- El sistema permitirá visualizar resultados del test de Kolb

Módulo de Materia

- El sistema permitirá crear materia
- El sistema permitirá editar materia
- El sistema permitirá eliminar materia
- El sistema permitirá listar materia
- El sistema permitirá asignar docente

Módulo de Contenido

- El sistema permitirá cargar contenido a través del ordenador
- El sistema permitirá insertar contenido a través de enlaces de YouTube
- El sistema permitirá insertar contenido a través de iframe
- El sistema permitirá modificar contenido
- El sistema permitirá eliminar contenido
- El sistema permitirá listar contenido
- El sistema permitirá visualizar contenido
- El sistema permitirá recomendar contenido
- El sistema permitirá mostrar instructivo del sistema para el docente
- El sistema permitirá mostrar instructivo del sistema para el estudiante
- El sistema permitirá añadir sección de pregunta para reforzar conocimientos
- El sistema permitirá modificar sección de preguntas para reforzar conocimientos
- El sistema permitirá eliminar sección de preguntas para reforzar conocimientos
- El sistema permitirá visualizar sección de preguntas para reforzar conocimientos
- El sistema permitirá validar respuesta de pregunta para reforzar conocimientos

Requerimientos no funcionales

Los requerimientos no funcionales describen los aspectos del sistema que no son visibles por el usuario, no influye con las funcionalidades del sistema, pero buscan brindar una mejor calidad al mismo. Entre estos se tiene:

- El sistema tendrá disponibilidad.
- El sistema será persistente con los datos.

- El sistema será fiable.
- El sistema será usable.

➤ Modelo conceptual de funcionalidades del sistema

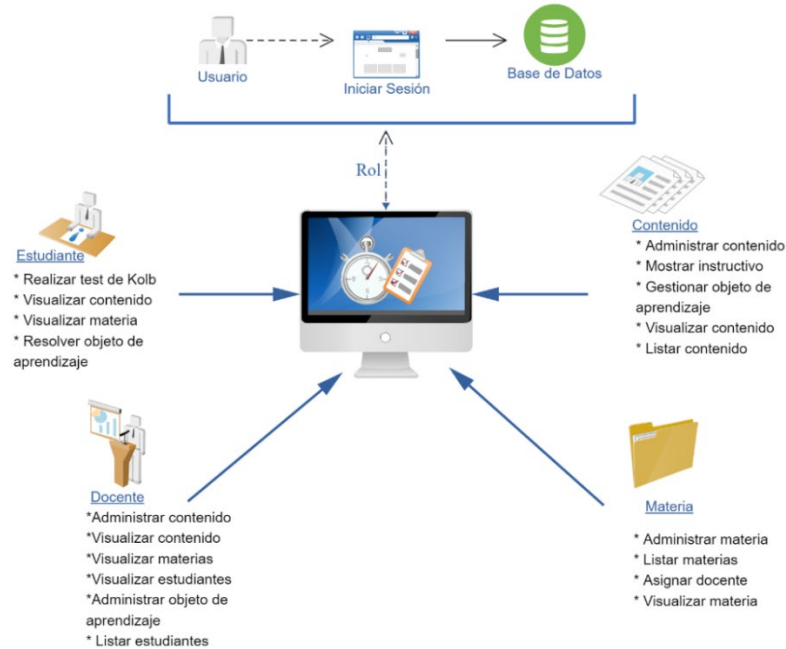


Figura 13-3: Mapa conceptual de funcionalidades del sistema

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En la **Figura 13-3** se detalla de forma gráfica las funcionalidades del sistema recomendador, en donde se puede observar que los usuarios tienen acceso a funciones específicas de acuerdo a su rol asignado.

➤ Historias de Usuario y metáforas del sistema

En base al análisis de los requerimientos listados anteriormente, se establecen 10 metáforas, y 32 historias de usuario para el desarrollo del sistema, mismas que se detallan en la **Tabla 17-3** y **Tabla 18-3**.

Tabla 17-3: Metáforas del sistema

Tipo	Descripción
Metáfora	Recolectar información para definir requerimiento funcionales y no funcionales
Metáfora	Investigar y analizar el estándar de codificación
Metáfora	Estudio e instalación de herramientas para el desarrollo
Metáfora	Investigar y analizar los estilos de aprendizaje

Tipo	Descripción
Metáfora	Investigar y analizar los sistemas recomendadores
Metáfora	Analizar y diseñar la arquitectura del sistema
Metáfora	Diseño de la base de datos
Metáfora	Analizar los roles del sistema
Metáfora	Diseño de la interfaz de usuario
Metáfora	Manual de usuario

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Tabla 18-3: Historias del sistema

Tipo	Descripción
Usuario	Autorizar acceso de usuarios al sistema
Usuario	Ingresar nuevos usuarios
Usuario	Modificar usuarios
Usuario	Eliminar usuarios
Usuario	Listar usuarios
Usuario	Listar usuarios por estudiante
Usuario	Listar usuarios por docente
Usuario	Visualizar usuarios
Usuario	Asignar semestre a usuarios estudiantes
Usuario	Automatizar el test de Kolb
Usuario	Obtener el estilo de aprendizaje de test del Kolb
Usuario	Visualizar resultados del test de Kolb
Usuario	Asignar curso
Usuario	Crear materia
Usuario	Modificar materia
Usuario	Eliminar materia
Usuario	Listar materias
Usuario	Asignar docente
Usuario	Insertar contenido a través de iframe
Usuario	Insertar contenido a través de enlaces de YouTube
Usuario	Cargar contenido a través del ordenador
Usuario	Modificar contenido
Usuario	Eliminar contenido

Tipo	Descripción
Usuario	Listar contenido
Usuario	Visualizar contenido
Usuario	Recomendar contenido
Usuario	Mostrar instructivo del sistema para el docente
Usuario	Mostrar instructivo del sistema para el estudiante
Usuario	Añadir sección de pregunta para reforzar conocimientos
Usuario	Modificar sección de preguntas para reforzar conocimientos
Usuario	Eliminar sección de preguntas para reforzar conocimientos
Usuario	Visualizar sección de preguntas para reforzar conocimientos
Usuario	Validar respuesta de pregunta para reforzar conocimientos

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

b. Fase de planificación

➤ Release planning

Se estable las métricas a emplear para definir la prioridad y estimaciones de tiempo y esfuerzo que requerirán las tareas para el desarrollo del sistema, para lo cual se empleó el método t-shirt (Talla de camiseta) el cual establece las estimaciones o tallas de: XS, S, M y L tal como se muestra en la **Tabla 19-3**.

Tabla 19-3: Método t-shirt

Talla	Puntos estimados	Horas de trabajo	Días
XS	6	6	1
S	18	18	3
M	30	30	5
L	60	60	10

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En base a la **Tabla 19-3** se observa que, un punto estimado equivale a una hora de trabajo realizada por dos personas, teniendo en cuenta que al día se trabaja 6 horas, se puede interpretar la información de la siguiente manera:

- **XS:** equivale a un día de trabajo compuesto por 6 horas de trabajo realizado por dos personas.
- **S:** equivale a tres días de trabajo de 6 horas cada uno teniendo un total de 18 horas de trabajo realizado por dos personas.
- **M:** equivale a 5 días de trabajo (una semana laboral), a 6 horas diarias dando un total de 30 horas de trabajo realizado por dos personas.

- **L:** equivale a 10 días de trabajo (dos semanas laborales), a 6 horas diarias dando un total de 60 horas de trabajo realizado por dos personas. Este es el valor máximo de trabajo el cual representa la duración de cada iteración (2 semanas).

➤ Priorizar y estimar esfuerzos

En base a la métrica para determinar el esfuerzo establecida en la **Tabla 19-3** se priorizó y estimó el esfuerzo necesario para cada una de las historias de usuario y metáforas del sistema, con el fin de orientar el desarrollo del sistema, tal como se muestra en la **Tabla 20-3**.

Tabla 20-3: Priorización y estimación de esfuerzo de MS e HU del sistema

ID	Descripción	Prioridad	Esfuerzo (Puntos estimados)
Metáforas del sistema			
MS_01	Recolectar información para definir requerimientos funcionales y no funcionales	Muy Alta	60
MS_02	Investigar y analizar el estándar de codificación	Muy Alta	06
MS_03	Estudio e instalación de herramientas para el desarrollo	Muy Alta	18
MS_04	Investigar y analizar los estilos de aprendizaje	Muy Alta	60
MS_05	Investigar y analizar los sistemas recomendadores	Muy Alta	30
MS_06	Analizar y diseñar la arquitectura del sistema	Muy Alta	18
MS_07	Analizar y diseñar la base de datos	Muy Alta	18
MS_08	Analizar los roles del sistema	Muy Alta	06
MS_09	Diseño de la interfaz de usuario	Muy Alta	30
MS_10	Documentar el manual de usuario	Muy Baja	18
Historias de usuario			
HU_01	Autorizar acceso de usuarios al sistema	Alta	18
HU_02	Ingresar nuevos usuarios	Alta	18
HU_03	Modificar usuarios	Media	18
HU_04	Eliminar usuarios	Media	06
HU_05	Listar usuarios	Alta	06
HU_06	Listar usuarios por estudiante	Media	06
HU_07	Listar usuarios por docente	Media	06
HU_08	Visualizar datos de usuarios	Alta	06
HU_09	Asignar semestre a usuarios estudiantes	Alta	06
HU_10	Automatizar el test de Kolb	Muy Alta	18
HU_11	Obtener el estilo de aprendizaje del test del Kolb	Muy Alta	60
HU_12	Visualizar resultados del test de Kolb	Alta	30

ID	Descripción	Prioridad	Esfuerzo (Puntos estimados)
HU_13	Crear materia	Alta	18
HU_14	Modificar materia	Media	06
HU_15	Eliminar materia	Media	06
HU_16	Listar materias	Alta	06
HU_17	Asignar docente a materia	Alta	06
HU_18	Insertar contenido a través de iframe	Media	18
HU_19	Insertar contenido a través de enlaces de YouTube	Media	18
HU_20	Cargar contenido a través del ordenador	Alta	18
HU_21	Modificar contenido	Alta	18
HU_22	Eliminar contenido	Media	06
HU_23	Listar contenidos	Alta	06
HU_24	Visualizar contenido	Muy Alta	18
HU_25	Recomendar contenido	Muy Alta	60
HU_26	Mostrar instructivo del sistema para el docente	Muy Baja	06
HU_27	Mostrar instructivo del sistema para el estudiante	Muy Baja	06
HU_28	Añadir sección de pregunta para reforzar conocimientos	Media	18
HU_29	Modificar sección de preguntas para reforzar conocimientos	Baja	06
HU_30	Eliminar sección de preguntas para reforzar conocimientos	Baja	06
HU_31	Visualizar sección de preguntas para reforzar conocimientos	Baja	06
HU_32	Validar respuesta de pregunta para reforzar conocimientos	Media	06

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Asignar pareja

La metodología propone desarrollar el trabajo en parejas, con el fin de potenciar el manejo de código y reducir errores. Para el desarrollo del presente sistema se cuenta con una pareja, por lo cual no habrá inconveniente en este proceso, la pareja asignada se visualiza en la siguiente tabla.

Tabla 21-3: Parejas asignadas

	Nombres	Tipo
Pareja de programadores	Dayana Elizabeth Almache Lopez	Desarrollador
	Kevin Paul Jácome Niama	Desarrollador

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Iteración del proyecto

Una vez establecido la estimación de tiempo y esfuerzo, se inició con el detalle de la planificación para asignar los recursos de una manera eficiente en el desarrollo del sistema. Se trabaja en el proyecto por iteraciones de dos semanas, comprendidas por días laborables de lunes a viernes, a

6 horas diarias de trabajo por dos personas. La planificación inicia el 1 de octubre del 2020 y culmina el 26 de marzo del 2021. Este plan consta de 120 días laborables, con un total de 720 horas de trabajo.

En la **Tabla 22-3** se detalla el número de historias realizadas en cada iteración, las horas asignadas para su desarrollo, su fecha de inicio y fin, así mismo como los responsables de cada actividad.

Tabla 22-3: Planificación de actividades del proyecto

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
01	MS_01	Recolectar información para definir requerimientos funcionales y no funcionales	60	01-10-2020	14-10-2020	Dayana Almache Kevin Jácome
02	MS_02	Investigar y analizar el estándar de codificación	06	15-10-2020	15-10-2020	Dayana Almache Kevin Jácome
	MS_03	Estudio e instalación de herramientas para el desarrollo	18	16-10-2020	20-10-2020	Dayana Almache Kevin Jácome
	MS_04	Investigar y analizar los estilos de aprendizaje	60	21-10-2020	04-11-2020	Dayana Almache Kevin Jácome
03	MS_05	Investigar y analizar los sistemas recomendadores	30	05-11-2020	11-11-2020	Dayana Almache Kevin Jácome
	MS_06	Analizar y diseñar la arquitectura del sistema	18	12-11-2020	16-11-2020	Dayana Almache Kevin Jácome
04	MS_07	Analizar y diseñar la base de datos	18	17-11-2020	19-11-2020	Dayana Almache Kevin Jácome
	MS_08	Analizar los roles del sistema	06	20-11-2020	20-11-2020	Dayana Almache Kevin Jácome
	MS_09	Diseño de la interfaz de usuario	30	23-11-2020	27-11-2020	Dayana Almache Kevin Jácome
05	HU_10	Automatizar el test de Kolb	18	30-11-2020	02-12-2020	Dayana Almache Kevin Jácome
	HU_11	Obtener el estilo de aprendizaje del test del Kolb	60	03-12-2020	16-12-2020	Dayana Almache Kevin Jácome
06	HU_24	Visualizar contenido	18	17-12-2020	21-12-2020	Dayana Almache Kevin Jácome
	HU_25	Recomendar contenido	60	22-12-2020	12-01-2021	Dayana Almache

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
07						Kevin Jácome
	HU_20	Cargar contenido a través del ordenador	18	13-01-2021	15-01-2021	Dayana Almache Kevin Jácome
08	HU_21	Modificar contenido	18	18-01-2021	20-01-2021	Dayana Almache Kevin Jácome
	HU_23	Listar contenidos	06	21-01-2021	21-01-2021	Dayana Almache Kevin Jácome
	HU_01	Autorizar acceso de usuarios al sistema	18	22-01-2021	26-01-2021	Dayana Almache Kevin Jácome
	HU_02	Ingresar nuevos usuarios	18	27-01-2021	29-01-2021	Dayana Almache Kevin Jácome
09	HU_05	Listar usuarios	06	01-02-2021	01-02-2021	Dayana Almache Kevin Jácome
	HU_08	Visualizar datos de usuarios	06	02-02-2021	02-02-2021	Dayana Almache Kevin Jácome
	HU_09	Asignar semestre a usuarios estudiantes	06	03-02-2021	03-02-2021	Dayana Almache Kevin Jácome
	HU_12	Visualizar resultados del test de Kolb	30	04-02-2021	10-02-2021	Dayana Almache Kevin Jácome
	HU_16	Listar materias	06	11-02-2021	11-02-2021	Dayana Almache Kevin Jácome
	HU_17	Asignar docente a materia	06	12-02-2021	12-02-2021	Dayana Almache Kevin Jácome
10	HU_13	Crear materia	18	15-02-2021	17-02-2021	Dayana Almache Kevin Jácome
	HU_22	Eliminar contenido	06	18-02-2021	18-02-2021	Dayana Almache Kevin Jácome
	HU_03	Modificar usuarios	18	19-02-2021	23-02-2021	Dayana Almache Kevin Jácome
	HU_04	Visualizar contenido	06	24-02-2021	24-02-2021	Dayana Almache Kevin Jácome
	HU_06	Listar usuarios por estudiante	06	25-02-2021	25-02-2021	Dayana Almache Kevin Jácome
	HU_07	Listar usuarios por docente	06	26-02-2021	26-02-2021	Dayana Almache Kevin Jácome
11	HU_18	Insertar contenido a través de iframe	18	01-03-2021	03-03-2021	Dayana Almache Kevin Jácome
	HU_19	Insertar contenido a través de enlaces de YouTube	18	04-03-2021	08-03-2021	Dayana Almache Kevin Jácome

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
	HU_28	Añadir sección de pregunta para reforzar conocimientos	18	09-03-2021	11-03-2021	Dayana Almache Kevin Jácome
	HU_32	Validar respuesta de pregunta para reforzar conocimientos	06	12-03-2021	12-03-2021	Dayana Almache Kevin Jácome
12	HU_14	Modificar materia	06	15-03-2021	15-03-2021	Dayana Almache Kevin Jácome
	HU_15	Eliminar materia	06	16-03-2021	15-03-2021	Dayana Almache Kevin Jácome
	HU_29	Modificar sección de preguntas para reforzar conocimientos	06	17-03-2021	17-03-2021	Dayana Almache Kevin Jácome
	HU_30	Eliminar sección de preguntas para reforzar conocimientos	06	18-03-2021	18-03-2021	Dayana Almache Kevin Jácome
	HU_31	Visualizar sección de preguntas para reforzar conocimientos	06	19-03-2021	19-03-2021	Dayana Almache Kevin Jácome
	HU_26	Mostrar instructivo del sistema para el docente	06	22-03-2021	22-03-2021	Dayana Almache Kevin Jácome
	HU_27	Mostrar instructivo del sistema para el estudiante	06	23-03-2021	23-03-2021	Dayana Almache Kevin Jácome
	MS_10	Documentar el manual de usuario	18	24-03-2021	26-03-2021	Dayana Almache Kevin Jácome

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En base a la **Tabla 22-3**, se puede vislumbrar que para el desarrollo del sistema se tuvo un total de 10 metáforas del sistema y 32 historias de usuario, las cuales fueron divididas en 12 iteraciones, distribuidas desde el 01-10-2020 hasta el 26-03-2021, teniendo un tiempo total de desarrollo de 720 horas.

Cada una de las HU y MS, fueron documentadas empleando fichas que presentan su información de tal manera que sean fácilmente entendibles. El formato de las fichas de HU se presenta en la **Tabla 23-3**, correspondiente a la HU “Autorizar acceso de usuarios al sistema”, la cual supone el control y autorización de ingresar al sistema mediante la credencial de los usuarios.

Tabla 23-3: Formato de ficha historia de usuario

HISTORIA DE USUARIO	
ID: HU_01	Nombre de la Historia: Autorizar acceso de usuarios al sistema
Usuario/Rol: Desarrollador	Iteración Asignada: 08

Prioridad del negocio: Alta	Puntos estimados: 18
Riesgo de desarrollo: Alta	Puntos reales: 18
Descripción: Como desarrollador quiero autorizar acceso de usuarios al sistema para que puedan acceder a todas sus funciones.	
Observación: Existen campos obligatorios.	
Historia de Usuario (Reverso) Pruebas de Aceptación	
<ul style="list-style-type: none"> • Verificar que los campos de las credenciales estén llenos. • Comprobar validez de los datos de autenticación. 	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

c. *Fase de diseño*

➤ **Arquitectura del Sistema**

La arquitectura se diseñó en base a los análisis previos de las funcionalidades del sistema, dicho análisis se lo representa en los diferentes diagramas UML para representalas de forma más técnica y entendible.

➤ **Diagrama de caso de uso**

Permiten visualizar el comportamiento del sistema ante la interacción de los usuarios con sus funcionalidades del software. En la **Figura 14-3** se especifica las funcionalidades generales que pueden realizar los actores (administrador, el docente y el estudiante) dentro del sistema.

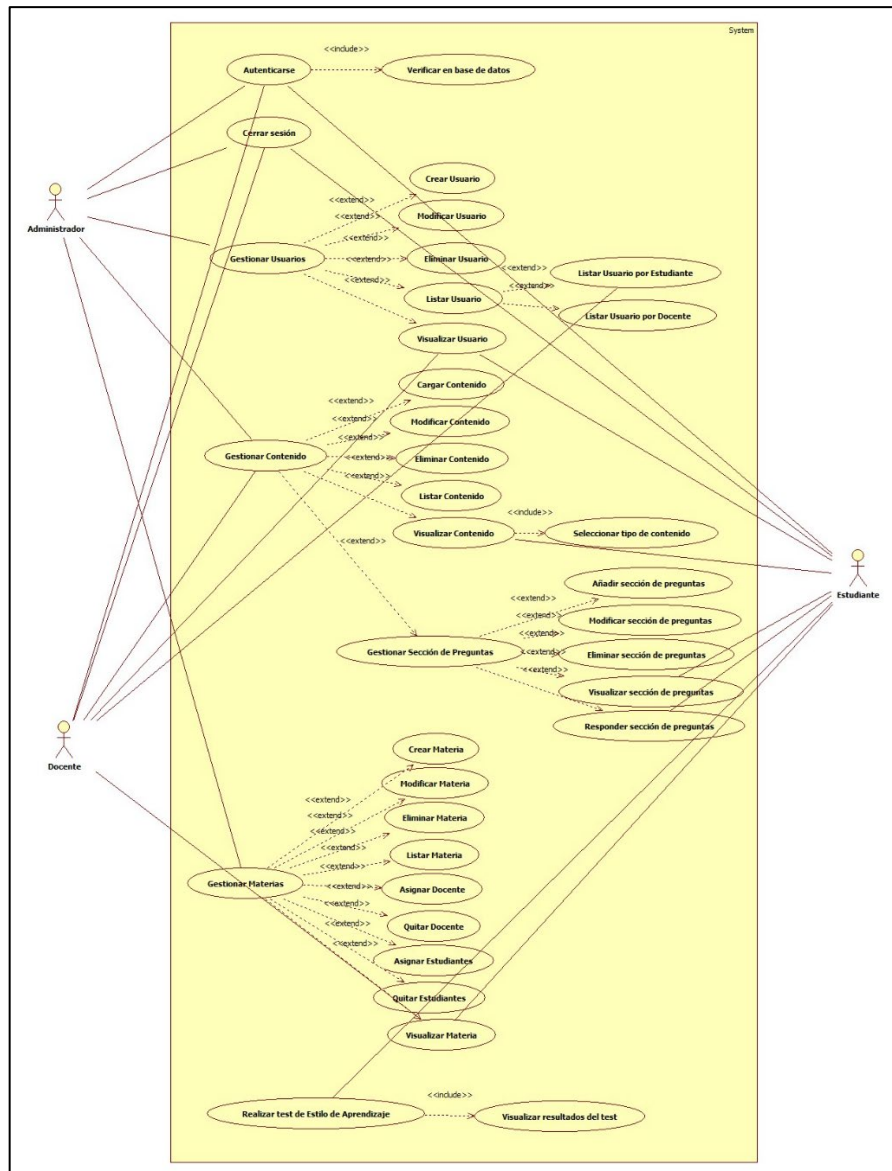


Figura 14-3: Diagrama de casos de uso

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En la **Tabla 24-3** se describe los actores, descripción, precondiciones, secuencia, y excepciones del caso de uso.

Tabla 24-3: Documentación de caso de uso “autenticación del administrador”

Casos de uso	Funcionalidades del administrador	
Descripción	Los usuarios al ingresar al sistema deberán autenticarse con sus credenciales correo electrónico y contraseña, el sistema verificará que se encuentre registrado, si es así le permitirá el acceso sistema, caso contrario se lo negará.	
Precondición	Estar registrado en la base datos.	
Secuencia normal	Paso	Acción

	1	Ingresar al sistema.
	2	Ingresar las credenciales con el que fue registrado.
	3	El sistema verifica los datos ingresados.
	4	El sistema realiza una búsqueda de las credenciales en la base de datos,
	5	El sistema permite el ingreso si existe registro caso contrario emite mensaje y niega permiso de ingreso.
Post condición	El usuario ingresa al sistema.	
Excepciones	Paso	Acción
	1	El sistema alertara con mensajes si no están llenos todos los campos.
	3	El sistema alertara con mensajes si las credenciales no se encuentran registradas en la base de datos.

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En el **Anexo B** se detallan todos los casos de uso involucrados dentro del presente sistema.

➤ Diagrama de la arquitectura del sistema

La arquitectura que manejará el presente proyecto será una arquitectura n-capas siguiendo el patrón MVC tal como se muestra en la **Figura 15-3**, a través de un diagrama de componentes de UML (Lenguaje de Modelado Unificado).

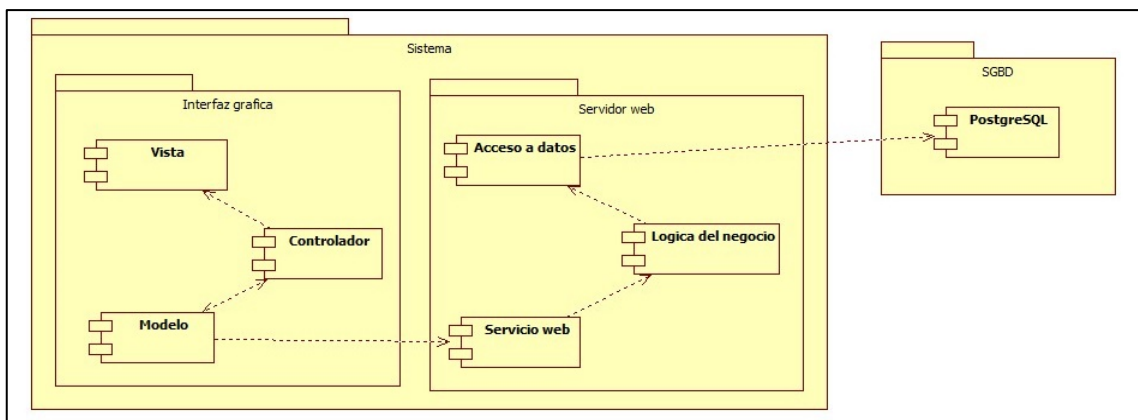


Figura 15-3: Diagrama de arquitectura del sistema

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Tal como se puede observar en la **Figura 15-3**, el sistema estará compuesto por:

- **Interfaz gráfica:** contiene los componentes de MVC que son vista, modelo y controlador.
- **Servidor web:** contiene los componentes de acceso a datos, lógica del negocio y los servicios web.
- **Sistema gestor de base de datos:** donde se encontrará alojado PostgreSQL donde se almacenará toda la información del sistema.

La metáfora del sistema del desarrollo de arquitectura generó una tarea de ingeniería y una prueba de aceptación, mismas que se muestran a continuación.

Tabla 25-3: Metáfora “Analizar y diseñar la arquitectura del sistema”

METÁFORA DEL SISTEMA	
ID: 06	Nombre de la Historia: Analizar y diseñar la arquitectura del sistema
Usuario/Rol: Desarrollador	Iteración Asignada: 04
Prioridad del negocio: Muy alta	Puntos estimados: 18
Riesgo de desarrollo: Alta	Puntos reales: 18
Descripción: Como desarrollador quiero analizar y posteriormente diseñar la arquitectura del sistema para asegurar que en un futuro el sistema sea escalable y posea un mantenimiento más fácil.	
Observación: La arquitectura que nos ayuda a que el sistema sea escalable es en n capas.	
Metáfora (Reverso) Pruebas de Aceptación	
Verificar que la arquitectura del sistema web cumpla con lo establecido.	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

A continuación, una de las tareas de ingeniería de la MS_06 referente al análisis y diseño de la arquitectura del sistema. Las demás tarjetas se encuentran en el **Anexo C**.

Tabla 26-3: Tarea de ingeniería de la MS_06

TAREA DE INGENIERÍA	
Iteración: 04	Número de Tarea: 01
Nombre de la Metáfora: Analizar y diseñar la arquitectura del sistema	
Nombre de la Tarea: Definir una arquitectura del sistema.	
Responsable: Dayana Almache y Kevin Jácome	Tipo de Tarea: Investigación
Fecha Inicio: 12-11-2020	Fecha Fin: 13-12-2020
Descripción: Como desarrollador quiero diseñar una arquitectura que facilite el desarrollo del sistema web y en un futuro pueda realizar cambios sin mayor dificultad.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> • Verificar el diseño de la arquitectura del sistema a desarrollar. 	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Estándar de programación

Estándar de programación para el sistema

A la hora de desarrollar el sistema uno de los puntos más importantes es la estandarización del código, debido a que permite reducir errores. En el caso de XP al emplear la programación en parejas, esto ayuda a que los dos integrantes puedan programar sin dificultades, entendiendo e identificando fácilmente la estructura de la codificación.

Se optó por el estándar denominado Upper CamelCase, el cual establece las siguientes consideraciones para la definición de programación:

- **Clases** <NombreClaseNombreCapa>, se escribe en cada palabra la primera letra con mayúscula y al final de cada nombre de la clase se escribe la capa a la cual pertenece. Ejemplo: <EstudianteControlador>.
- **Variable** <El nombre de la variable>, se escribe la primera palabra en minúscula y en las siguientes palabras la primera letra será mayúscula y las demás letras de la palabra minúsculas. Ejemplo: <idRol>.
- **Métodos** <El nombre del método>, se escribe la primera palabra en minúscula y en las siguientes palabras se escribirá la primera letra con mayúscula y las demás letras de la palabra con minúscula. Ejemplo: <getNombre>
- **Paquetes** <nombredelproyecto.paquetemodulo.nombrecapa>, se escribe iniciando con el nombre del proyecto, seguido por el módulo y capa, todo esto en minúsculas y separados por puntos. Ejemplo: <customlearning.datos.controlador>

Estándar de programación para la base de datos

Para el diseño y codificación de la base de datos del sistema también se optó por emplear un estándar de codificación, en este caso el estándar Snake case, el cual define que, para escribir los nombres de clases, campos, columnas, índices, variables, etc. se utilizan solamente letras minúsculas y de existir espacios se los reemplaza por el carácter de subrayado o también conocido como guion bajo (_). Un ejemplo de esto es el nombre de la tabla para almacenar los datos de los tipos de contenido <tipo_contenido>.

Esta metáfora se la describe con sus respectivas tareas de ingeniería y pruebas de aceptación, como se detalla a continuación.

Tabla 27-3: Metáfora “Investigar y analizar el estándar de codificación”

METÁFORA DEL SISTEMA	
ID: 02	Nombre de la Historia: Investigar y analizar el estándar de codificación.
Usuario/Rol: Desarrollador	Iteración Asignada: 02
Prioridad del negocio: Muy Alta	Puntos estimados: 06
Riesgo de desarrollo: Medio	Puntos reales: 06
Descripción: Como desarrollador investigar y analizar cada uno de los estándares definidos para poder desarrollar código homogéneo y de fácil comprensión.	
Observación: El estándar de codificación se basará en el estándar definido por el equipo de trabajo.	
Metáforas (Reverso) Pruebas de Aceptación	
<ul style="list-style-type: none"> • Verificar que el estándar sea entendido por el equipo de desarrollo • Verificar que todos los miembros del equipo de desarrollo aprueben el estándar seleccionado 	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Las demás tarjetas de esta metáfora se encuentran presentes en el **Anexo C**.

➤ **Diseño de la base de datos**

El diseño de la base de datos se desarrolla en el modelo entidad relación (ER) en donde se modela los datos, se representan a los actores como entidades y las interrelaciones que realizan estos en el sistema, para así lograr tener una abstracción de alto nivel. Dicho diagrama se puede observar en el **Anexo D**.

Una vez se tiene el modelo ER, se procede a representarle a través del modelo físico el cual brinda una visión más general de lo que se contiene en el gestor de base de datos.

Definición de nombres

- **Base de datos:** El nombre acordado por el equipo para definir a la base de datos del sistema es “bdCustomLearning”.
- **Tablas:** Comienza con el prefijo tb luego cada palabra con la primera letra en mayúscula y las demás en minúscula, por ejemplo: tbEstudiante.
- **Atributos:** Se escribe en letras minúsculas y sin guion ni espacios.

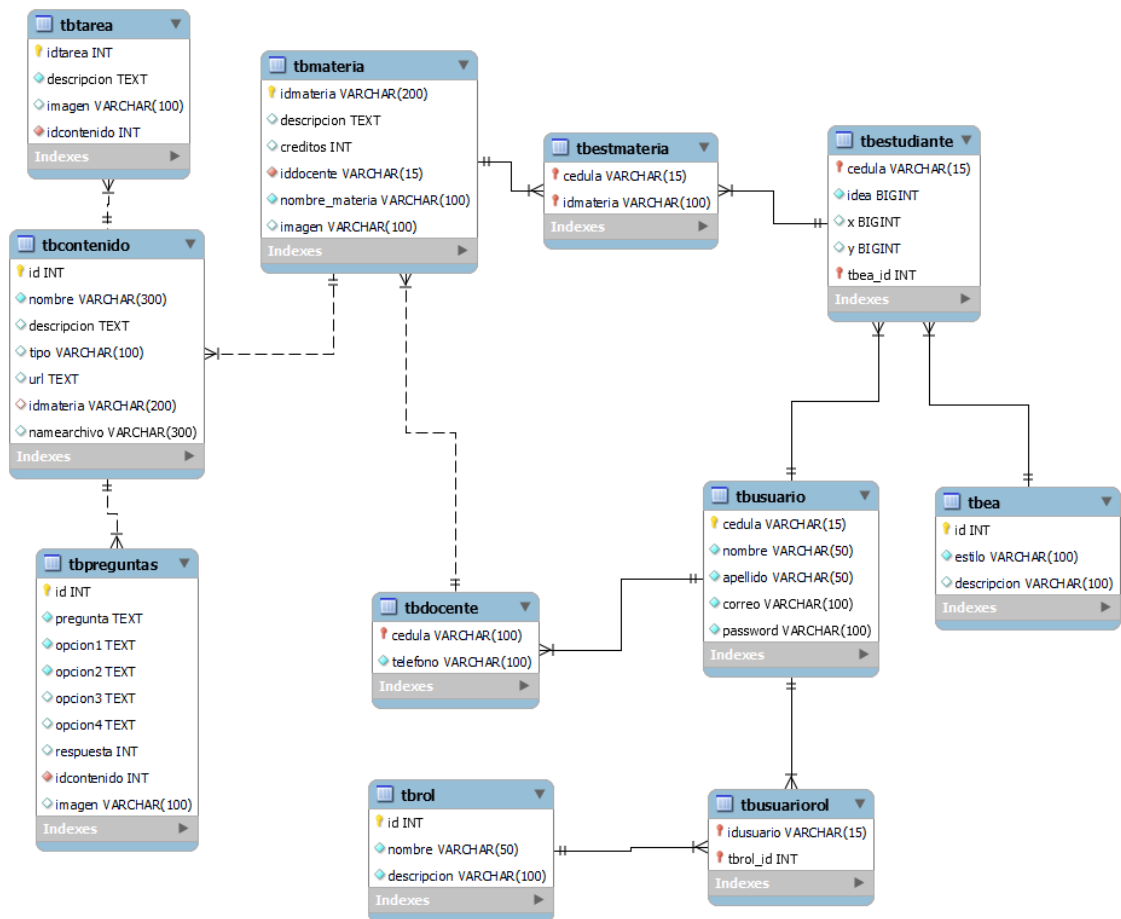


Figura 16-3: Esquema de la base de datos del sistema

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En base a la **Figura 16-3**, se puede observar que la base de datos diseñada se conforma de 11 tablas como parte del subsistema que se ha desarrollado en este trabajo, las cuales, a su vez, se relacionan entre sí para permitir el desarrollo de las diferentes funcionalidades.

Diccionario de datos

Permite tener un registro adecuado de todos los datos pertinentes al sistema con información precisa y rigurosa, para así facilitar el análisis del sistema en su base de datos. En la **Tabla 28-3** se puede observar un ejemplo de cómo se maneja este diccionario de datos.

Tabla 28-3: Diccionario de datos de la tabla usuario

Nombre de tabla: tbusuario				
Descripción: Datos personales de un usuario que se registra en el sistema.				
Nombre del campo	Descripción	Tipo de dato y tamaño	Permite NULL	Valor permitido del dato
cedula (Pk)	Número de cedula de identidad.	varchar (15)	No	Identificador único de 10 dígitos con rango [0 a 9]*

nombre	Identificador de nombres del usuario.	varchar (50)	No	Primer nombre + (segundo nombre) = {[A-Z a-z]}
apellido	Identificador de apellidos del usuario.	varchar (50)	No	Primer apellido + (segundo apellido) = {[A-Z a-z]}
correo	Dirección de correo electrónico del usuario.	varchar (100)	No	Caracteres numéricos y alfanuméricos con rango {[A-Z a-z] [0 a 9]}
password	Clave de ingreso al sistema.	varchar (100)	No	Caracteres numéricos y alfanuméricos con rango {[A-Z a-z] [0 a 9]} cifrada con criptografía *

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Las demás tablas del diccionario de datos se las puede visualizar en el **Anexo E**, en donde se detalla cada entidad con su respectiva información.

La metáfora de diseño de base de datos se describe como una metáfora del sistema, la cual se desarrolló y documento a empleando tablas respectivas tal como se puede observar en la **Tabla 29-3**.

Tabla 29-3: Metáfora “Análisis y diseño de la base de datos”

METÁFORA DEL SISTEMA	
ID: 07	Nombre de la Historia: Analizar y diseñar la base de datos
Usuario/Rol: Desarrollador	Iteración Asignada: 04
Prioridad del negocio: Muy alta	Puntos estimados: 18
Riesgo de desarrollo: Alta	Puntos reales: 18
Descripción: Como desarrollador quiero analizar y posteriormente diseñar la base de datos, para el correcto almacenamiento y gestión de los datos a emplear en el sistema	
Observación: La información dada a conocer por el encargado es fundamental para no tener ninguna corrección a futuro.	
Metáforas (Reverso) Pruebas de Aceptación	
<ul style="list-style-type: none"> • Verificar que la base de datos cumpla con los requerimientos para el funcionamiento del sistema. 	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Las demás tablas referentes a la presente Metáforas se encuentran detalladas en el **Anexo C**.

➤ Diseño de la interfaz

El diseño de la interfaz de usuario es una parte muy importante en el sistema ya que es la presentación que se les dará a los usuarios, por esta razón se debe presentar una interfaz intuitiva y consistente.

La interfaz principal del sistema recomendador “Custom Learning” se desarrolló con la finalidad de brindar al usuario la manera de poder comunicarse e interactuar con el sistema y todas sus funcionalidades. Las interfaces del sistema se desarrollaron empleando el framework PrimeFaces empleando una plantilla para mantener consistencia entre sus páginas, de manera que, la navegación se realice de manera intuitiva. En la **Figura 17-3** se muestra una captura de la pantalla principal del sistema “Custom Learning”.















Figura 17-3: Diseño de la interfaz de usuario

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En la **Tabla 33-3** se indican las diversas secciones que tendrá la plantilla de las interfaces, mientras que, en la **Figura 18-3**, se muestra un bosquejo de la interfaz de la pantalla principal del sistema “Custom Learning”. Dicho bosquejo contiene secciones definidas que permitirán al usuario interactuar y navegar por las demás páginas del sistema.

Tabla 30-3: Descripción de estilo de las secciones de la interfaz

Número	Descripción	Color de fondo	Color de texto
1	Icono y nombre del sistema	#2a3f54 	#fff 
2	Usuario	#2a3f54 	#fff 
3	Panel de navegación	#2a3f54 	#fff 
4	Menú de usuario	#ededed 	#000 

Número	Descripción	Color de fondo	Color de texto
5	Nombre de la sección	#f7f7f7 	#7292ab 
6	Contenido	#fff 	#000 

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

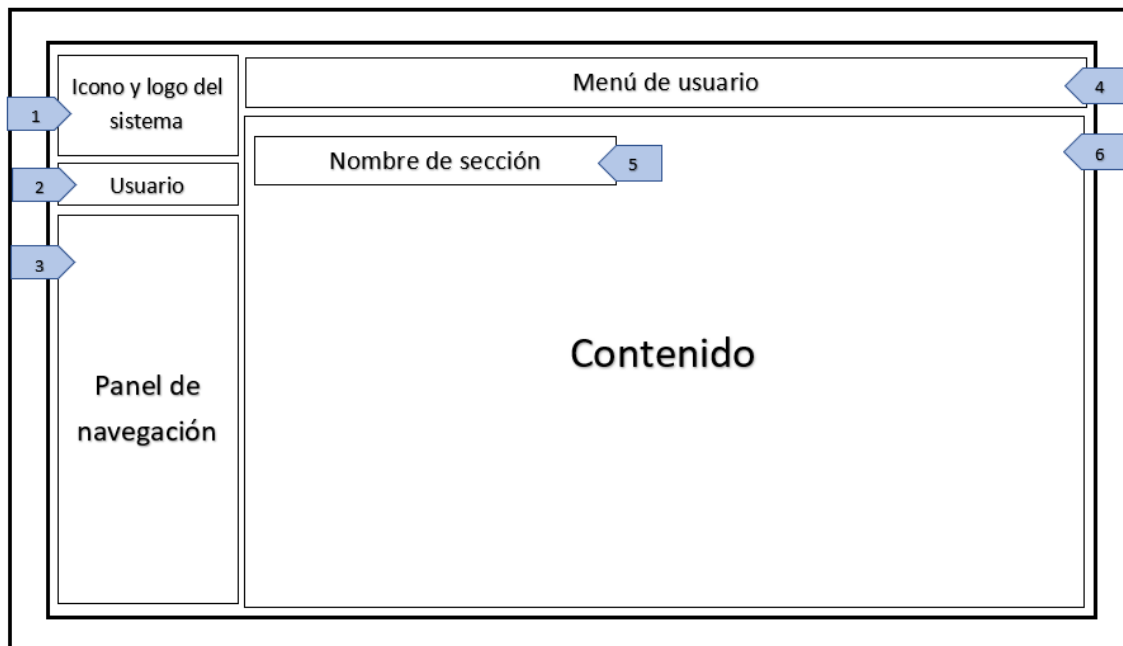
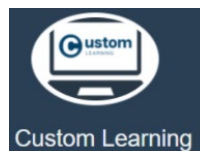


Figura 18-3: Bosquejo de la interfaz de usuario

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Descripción del bosquejo de interfaz

1. **Icono y logo del sistema:** sección que contendrá el logo y el nombre del sistema “Custom Learning”. Al hacer clic sobre él, se redireccionará a la página principal del sistema. El icono y nombre a mostrar son los siguientes:



2. **Usuario:** sección que indicará el nombre del usuario registrado dentro del sistema.
3. **Panel de navegación:** sección del panel de navegación del sistema que permitirá al usuario el cambiar entre las principales páginas del sistema con mayor facilidad. Contendrá las secciones de Usuarios, Materias y Contenidos.

4. **Menú de usuario:** sección que permitirá al usuario el poder visualizar las opciones del “Menú de usuario”. En el podrá cerrar sesión.
5. **Nombre de la sección:** sección para mostrar el nombre de la página/sección en la que se encuentre el usuario actualmente, facilitando la ubicación y navegación por el sitio.
6. **Contenido:** sección para el contenido. Esta representa la sección principal de las páginas en la cual se mostrará el contenido dependiendo de la página en la que se encuentre.

Esta metáfora de diseño de la interfaz de usuario es una Metáforas que genera una prueba de aceptación y dos tareas de ingeniería con sus respectivas pruebas de aceptación. La documentación de la metáfora se la puede observar en la **Tabla 31-3**.

Tabla 31-3: Metáfora “Diseño de la interfaz de usuario”

METÁFORA DEL SISTEMA	
ID: 09	Nombre de la Historia: Diseño de la interfaz de usuario
Usuario/Rol: Desarrollador	Iteración Asignada: 04-05
Prioridad del negocio: Muy alta	Puntos estimados: 30
Riesgo de desarrollo: Alta	Puntos reales: 30
Descripción: Como desarrollador quiero diseñar la interfaz de usuario para poder interactuar con el sistema.	
Observación: Para el diseño de la interfaz se debe tener conocimiento sobre PrimeFaces	
Metáforas (Reverso) Pruebas de Aceptación	
<ul style="list-style-type: none"> • Verificar que la interfaz de usuario emplee la plantilla de PrimeFaces 	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Las demás tablas con respecto a esta metáfora se la puede visualizar en el **Anexo C**.

➤ Gestión de riesgos

Riesgos

Dentro del desarrollo de cualquier proyecto, uno de los aspectos de mayor relevancia es la gestión de riesgos, ya que estos pueden provocar diversos problemas, retrasos e incluso el fracaso del proyecto, por lo cual es necesario plantear acciones para evitar, reducir y afrontar potenciales riesgos.

Para el presente trabajo se desarrolló una gestión de riesgos compuesta por la identificación de los principales riesgos que puedan aparecer, luego se analizó y priorizó dichos riesgos para determinar aquellos que puedan ser más posibles de ocurrir y su impacto dentro del proyecto, para finalmente gestionar medidas que permitan evitar, reducir o afrontar dichos riesgos en caso de ocurrir.

Identificación de riesgos

Iniciando este proceso se procede a identificar todos aquellos posibles riesgos que puedan presentarse en el transcurso del presente trabajo, para lo cual se realizó una reunión con los miembros del equipo de desarrollo donde se obtuvo una lista de 8 posibles riesgos, los mismos que se los clasificarán 3 grupos que son:

- **Riesgos Técnicos:** identifican problemas de diseño, interfaz, implementación, mantenimiento, etc. los cuales amenazan directamente a la calidad del sistema.
- **Riesgos Externos:** identifican problemas externos al ámbito de desarrollo los cuales están fuera del control de los involucrados en el proyecto incurriendo en pérdidas, retrasos y hasta la cancelación del proyecto.
- **Riesgos del Proyecto:** identifican problemas relacionados con la gestión del proyecto, así como planificaciones, comunicación y problemas entre los miembros, etc., los cuales afectan directamente a la planificación y ejecución del proyecto.

En la **Tabla 32-3**, se muestra la identificación y categorización de los riesgos para el presente proyecto, teniendo un total de 8 riesgos.

Tabla 32-3: Identificación y categorización de riesgos

ID	Categoría	Descripción
R01	Técnico	Daño en el hardware o software utilizado para el desarrollo.
R02		Pérdida de información.
R03		Especificación de requisitos poco precisa.
R04	Externo	Robo en el hardware utilizado para el desarrollo.
R05	Proyecto	Retiro inesperado de algún integrante del equipo de desarrollo.
R06		Planificación realizada de manera inadecuada.
R07		Informes poco claros sobre la evolución del proyecto.
R08		Desacuerdo entre los miembros del equipo de desarrollo.

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En base a la **Tabla 32-3**, se obtuvo un total de 8 riesgos clasificados de la siguiente manera: 3 riesgos técnicos, 1 riesgo externo y 4 riesgos del proyecto.

Análisis de riesgos

En esta fase se realiza una serie de procesos encaminados a describir y detallar cada uno de los riesgos identificados anteriormente, para en base a ello tomar medidas preventivas y correctivas respectivas. Este proceso esté compuesto por las siguientes fases:

- **Determinación de la probabilidad:** determina la probabilidad de que ocurra un riesgo en el proyecto, basándose en los valores de una tabla de valoración de probabilidad (**Anexo A**).

- **Determinación del impacto:** basándose en la tabla de valoración del impacto (**Anexo A**) determina un valor que representa el impacto que dicho riesgo tendrá dentro del proyecto en caso de ocurrir, así como también el retraso que este provocaría.
- **Determinación de la exposición al riesgo:** se determina multiplicando el valor correspondiente a la probabilidad del riesgo y el del impacto al riesgo para poder categorizarlo según la tabla de exposición de riesgos (**Anexo A**).
- **Determinación de la prioridad del riesgo:** basándose en los análisis anteriores se procede a realizar una priorización de los riesgos identificados (**Anexo A**).

Todos estos análisis están encaminados a determinar las estrategias necesarias que permitan evitar y/o minimizar el impacto de estos riesgos para llegar a su culminación exitosa del trabajo.

Gestión de riesgos

En base a los análisis descritos anteriormente, se procede con el proceso de gestión de riesgos, el cual busca establecer aquellas medidas preventivas y correctivas que permitan evitar y/o solucionar los diversos problemas que puedan surgir por los riesgos identificados. Para este objetivo se desarrollan las hojas de gestión de riesgos las cuales describen de manera detallada y ordenada los aspectos descritos anteriormente con la finalidad de poder gestionar cada uno de los riesgos durante el transcurso del proyecto. Estas hojas de gestión de riesgos y otros documentos adicionales sobre la misma temática se encuentran detallados en el **Anexo A**.

d. Fase de codificación

Esta fase da paso al desarrollo del sistema mediante la codificación de cada una de las historias de usuario establecidas por el cliente, detalladas en las tarjetas de historias de usuario, mismas que se las puede visualizar en el **Anexo C**.

La programación se desarrolló utilizando el lenguaje de programación Java a través del IDE de desarrollo de NetBeans, además se utilizó el framework de JavaSever Faces conocido como PrimeFaces para el desarrollo de las interfaces de usuario, y como gestor de base de datos se empleó PostgreSQL conjuntamente con pgAdmin III.

El proceso de desarrollo del sistema recomendador se lo dividió en 12 iteraciones, cada una con una duración de dos semanas, a las cuales se asignó el desarrollo de un número definido de historias de usuarios las cuales, comprendidas entre una fecha de inicio y fin específicas, así como un valor de esfuerzo requerido para su desarrollo.

Siguiendo las pautas que describe la metodología XP para el desarrollo de software, se efectuó la programación en parejas para la codificación del sistema, siendo un miembro del equipo de desarrollo quien codifique mientras el otro visualiza dicha tarea con el fin de revisar y ofrecer retroalimentación y comentarios encaminados a potenciar la fase de desarrollo y reducir e

identificar oportunamente errores que puedan surgir. Estos roles del equipo de desarrollo se alternaban cada cierto intervalo de tiempo.

A continuación, se describe cada iteración realizada para el desarrollo del presente sistema.

➤ Iteración 1

En la primera iteración se asignaron la realización de una metáfora, esta iteración inicio el 01 de octubre de 2020 y terminó el 14 de octubre de 2020, teniendo un esfuerzo estimado de 60 horas de trabajo. Dicha iteración está destinada a realizar la fase de análisis en la cual se recolecta la información necesaria para poder definir cómo se va a realizar el sistema.

En la **Tabla 33-3** se detallan las actividades pertinentes a la presente iteración, mientras que la **Tabla 34-3** describe a detalle la metáfora desarrollada.

Tabla 33-3: Actividades de la iteración 1

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
01	MS_01	Recolectar información para definir requerimientos funcionales y no funcionales	60	01-10-2020	14-10-2020	Dayana Almache Kevin Jácome
Total			60	01-10-2020	14-10-2020	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Tabla 34-3: Metáfora desarrollada en la iteración 1

METÁFORAS	
ID: 01	Nombre de la Historia: Recolectar la información para definir lo requerimientos funcionales y no funcionales
Usuario/Rol: Desarrollador	Iteración Asignada: 01
Prioridad del negocio: Muy alta	Puntos estimados: 60
Riesgo de desarrollo: Alta	Puntos reales: 60
Descripción: Como desarrollador quiero recolectar la información de los procesos de aprendizaje para definir los requerimientos funcionales y no funcionales, para el desarrollo del sistema web.	
Observación: Existe un docente que da detalles de los requerimientos	
Metáforas (Reverso) Pruebas de Aceptación	
Verificar que los requerimientos funcionales y no funcionales son los que requiere el cliente.	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Iteración 2

Esta iteración inició el 15 de octubre de 2020 y terminó el 12 de noviembre de 2020, donde se desarrollaron tres metáforas del sistema, mismas que comprenden procesos de análisis e

investigación previos al proceso de desarrollo. En la **Tabla 38-3** se detalla las actividades efectuadas en esta iteración.

Tabla 35-3: Actividades de la iteración 2

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
02	MS_02	Investigar y analizar el estándar de codificación	06	15-10-2020	15-10-2020	Dayana Almache Kevin Jácome
	MS_03	Estudio e instalación de herramientas para el desarrollo	18	16-10-2020	20-10-2020	Dayana Almache Kevin Jácome
	MS_04	Investigar y analizar los estilos de aprendizaje	36	21-10-2020	28-11-2020	Dayana Almache Kevin Jácome
Total			60	15-10-2020	28-11-2020	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Iteración 3

La presente iteración inició el 21 de octubre de 2020 y terminó el 12 de noviembre de 2020. En él se continua con el desarrollo de metáforas del sistema. Estas actividades desarrolladas se detallan en la **Tabla 39-3**.

Tabla 36-3: Actividades de la iteración 3

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
03	MS_04	Investigar y analizar los estilos de aprendizaje	24	21-10-2020	04-11-2020	Dayana Almache Kevin Jácome
	MS_05	Investigar y analizar los sistemas recomendadores	30	05-11-2020	11-11-2020	Dayana Almache Kevin Jácome
	MS_06	Analizar y diseñar la arquitectura del sistema	06	12-11-2020	12-11-2020	Dayana Almache Kevin Jácome
Total			60	21-10-2020	12-11-2020	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Iteración 4

La presente iteración inició desde el 13 de noviembre de 2020 y terminó el 26 de noviembre de 2020, desarrollándose completamente tres metáforas del sistema y parte de una cuarta, debido a que cada iteración solo tiene una duración de 60 horas. Dichas actividades desarrolladas se detallan en la **Tabla 37-3**.

Tabla 37-3: Actividades de la iteración 4

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
04	MS_06	Analizar y diseñar la arquitectura del sistema	12	13-11-2020	16-11-2020	Dayana Almache Kevin Jácome
	MS_07	Analizar y diseñar la base de datos	18	17-11-2020	19-11-2020	Dayana Almache Kevin Jácome
	MS_08	Analizar los roles del sistema	06	20-11-2020	20-11-2020	Dayana Almache Kevin Jácome
	MS_09	Diseño de la interfaz de usuario	24	23-11-2020	26-11-2020	Dayana Almache Kevin Jácome
Total			60	13-11-2020	26-11-2020	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Iteración 5

La presente iteración inició el 27 de noviembre de 2020 y terminó el 10 de diciembre de 2020. En él se terminó de desarrollar la MS_09 e inicio con el desarrollo de las historias de usuario, específicamente aquellas que poseían una prioridad muy alta, siendo estas la automatización del test de Kolb y la obtención del estilo de aprendizaje. Dichas actividades desarrolladas se detallan en la **Tabla 38-3**.

Tabla 38-3: Actividades de la iteración 5

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
05	MS_09	Diseño de la interfaz de usuario	06	27-11-2020	27-11-2020	Dayana Almache Kevin Jácome
	HU_10	Automatizar el test de Kolb	18	30-11-2020	02-12-2020	Dayana Almache Kevin Jácome
	HU_11	Obtener el estilo de aprendizaje del test del Kolb	36	03-12-2020	10-12-2020	Dayana Almache Kevin Jácome
Total			60	27-11-2020	10-12-2020	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Iteración 6

Esta iteración inició el 11 de diciembre de 2020 y terminó el 24 de diciembre de 2020, teniendo en su desarrollo la finalización de la HU_11, la HU_24 y parte de la HU_25, historias de usuario centradas especialmente en la gestión del contenido dentro del sistema. Dichas actividades desarrolladas se detallan en la **Tabla 39-3**.

Tabla 39-3: Actividades de la iteración 6

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
06	HU_11	Obtener el estilo de aprendizaje del test del Kolb	24	11-12-2020	16-12-2020	Dayana Almache Kevin Jácome
	HU_24	Visualizar contenido	18	17-12-2020	21-12-2020	Dayana Almache Kevin Jácome
	HU_25	Recomendar contenido	18	22-12-2020	24-12-2020	Dayana Almache Kevin Jácome
Total			60	11-12-2020	24-12-2020	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Iteración 7

Esta iteración inició el 04 de enero de 2021 y terminó el 15 de enero de 2021, desarrollando las HU_25 y HU_20 relacionadas con la gestión del contenido dentro del sistema. Dichas actividades desarrolladas se detallan en la **Tabla 40-3**.

Tabla 40-3: Actividades de la iteración 7

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
07	HU_25	Recomendar contenido	42	04-01-2021	12-01-2021	Dayana Almache Kevin Jácome
	HU_20	Cargar contenido a través del ordenador	18	13-01-2021	15-01-2021	Dayana Almache Kevin Jácome
Total			60	04-01-2021	15-01-2021	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Iteración 8

Esta iteración inició el 18 de enero de 2021 y terminó el 29 de enero de 2021, desarrollando un total de 4 historias de usuario, mismas que se centran en la gestión del contenido y gestión de usuarios. Dichas actividades desarrolladas se detallan en la **Tabla 41-3**.

Tabla 41-3: Actividades de la iteración 8

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
08	HU_21	Modificar contenido	18	18-01-2021	20-01-2021	Dayana Almache Kevin Jácome
	HU_23	Listar contenido	06	21-01-2021	21-01-2021	Dayana Almache Kevin Jácome
	HU_01	Autorizar acceso de usuarios al sistema	18	22-01-2021	26-01-2021	Dayana Almache Kevin Jácome

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
	HU_02	Ingresar nuevos usuarios	18	27-01-2021	29-01-2021	Dayana Almache Kevin Jácome
Total			60	18-01-2021	29-01-2021	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Iteración 9

Esta iteración inició el 01 de febrero de 2021 y terminó el 12 de febrero de 2021, desarrollando un total de 6 historias de usuario referentes a la gestión de usuarios y gestión de materias del sistema. Dichas actividades desarrolladas se detallan en la **Tabla 42-3**.

Tabla 42-3: Actividades de la iteración 9

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
09	HU_05	Listar usuarios	06	01-02-2021	01-02-2021	Dayana Almache Kevin Jácome
	HU_08	Visualizar datos de usuarios	06	02-02-2021	02-02-2021	Dayana Almache Kevin Jácome
	HU_09	Asignar semestre a usuarios estudiantes	06	03-02-2021	03-02-2021	Dayana Almache Kevin Jácome
	HU_12	Visualizar resultados del test de Kolb	30	04-02-2021	10-02-2021	Dayana Almache Kevin Jácome
	HU_16	Listar materias	06	11-02-2021	11-02-2021	Dayana Almache Kevin Jácome
	HU_17	Asignar docente a materia	06	12-02-2021	12-02-2021	Dayana Almache Kevin Jácome
Total			60	01-02-2021	12-02-2021	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Iteración 10

Esta iteración inició el 15 de febrero de 2021 y terminó el 26 de febrero de 2021, desarrollando un total de 6 historias de usuario, que por su prioridad son las referentes a procesos faltantes dentro de la gestión de materias, contenidos y usuarios dentro del sistema. Dichas actividades desarrolladas se detallan en la **Tabla 43-3**.

Tabla 43-3: Actividades de la iteración 10

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
10	HU_13	Crear materia	18	15-02-2021	17-02-2021	Dayana Almache Kevin Jácome

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
	HU_22	Eliminar contenido	06	18-02-2021	18-02-2021	Dayana Almache Kevin Jácome
	HU_03	Modificar usuarios	18	19-02-2021	23-02-2021	Dayana Almache Kevin Jácome
	HU_04	Visualizar contenido	06	24-02-2021	24-02-2021	Dayana Almache Kevin Jácome
	HU_06	Listar usuarios por estudiante	06	25-02-2021	25-02-2021	Dayana Almache Kevin Jácome
	HU_07	Listar usuarios por docente	06	26-02-2021	26-02-2021	Dayana Almache Kevin Jácome
Total			60	15-02-2021	26-02-2021	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Iteración 11

Esta iteración inició el 01 de marzo de 2021 y terminó el 12 de marzo de 2021, desarrollando un total de 4 historias de usuario, todas relacionadas a la gestión de contenidos dentro del sistema. Dichas actividades desarrolladas se detallan en la **Tabla 44-3**.

Tabla 44-3: Actividades de la iteración 11

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
11	HU_18	Insertar contenido a través de iframe	18	01-03-2021	03-03-2021	Dayana Almache Kevin Jácome
	HU_19	Insertar contenido a través de enlaces de YouTube	18	04-03-2021	08-03-2021	Dayana Almache Kevin Jácome
	HU_28	Añadir sección de pregunta para reforzar conocimientos	18	09-03-2021	11-03-2021	Dayana Almache Kevin Jácome
	HU_32	Validar respuesta de pregunta para reforzar conocimientos	06	12-03-2021	12-03-2021	Dayana Almache Kevin Jácome
Total			60	01-03-2021	12-03-2021	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Iteración 12

Esta iteración inició el 15 de marzo de 2021 y terminó el 26 de marzo de 2021. En esta iteración se desarrollaron un total de 7 historias de usuario y una metáfora, actividades con las cuales se terminó el desarrollo del sistema. Dichas actividades desarrolladas se detallan en la **Tabla 45-3**.

Tabla 45-3: Actividades de la iteración 12

Iter.	ID	Actividad	Esf.	Fecha inicio	Fecha fin	Responsables
12	HU_14	Modificar materia	06	15-03-2021	15-03-2021	Dayana Almache Kevin Jácome
	HU_15	Eliminar materia	06	16-03-2021	15-03-2021	Dayana Almache Kevin Jácome
	HU_29	Modificar sección de preguntas para reforzar conocimientos	06	17-03-2021	17-03-2021	Dayana Almache Kevin Jácome
	HU_30	Eliminar sección de preguntas para reforzar conocimientos	06	18-03-2021	18-03-2021	Dayana Almache Kevin Jácome
	HU_31	Visualizar sección de preguntas para reforzar conocimientos	06	19-03-2021	19-03-2021	Dayana Almache Kevin Jácome
	HU_26	Mostrar instructivo del sistema para el docente	06	22-03-2021	22-03-2021	Dayana Almache Kevin Jácome
	HU_27	Mostrar instructivo del sistema para el estudiante	06	23-03-2021	23-03-2021	Dayana Almache Kevin Jácome
	MS_10	Documentar el manual de usuario	18	24-03-2021	26-03-2021	Dayana Almache Kevin Jácome
Total			60	15-03-2021	26-03-2021	

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

e. *Fase de pruebas*

Dentro del proceso de desarrollo de XP esta fase tiene una gran importancia, ya que permite verificar el correcto progreso del desarrollo del sistema, efectuando pruebas a las funcionalidades desarrolladas con el fin de determinar si cumplen con los criterios de aceptación necesarios para poder seguir con el desarrollo de una siguiente actividad. Para este proceso se emplearon las pruebas de aceptación presentes en las historias de usuario, metáforas del sistema y tareas de ingeniería del sistema. En la **Tabla 46-3** se muestra la tarjeta de prueba de aceptación de la HU_01.

Tabla 46-3: Formato de tarjeta de prueba de aceptación

PRUEBA DE ACEPTACIÓN	
Código: PA1	Nombre de la Historia: Autorizar acceso de usuarios al sistema
Nombre de la Prueba: Verificar que los campos de las credenciales estén llenos.	
Responsable: Dayana Almache y Kevin Jácome	Fecha: 26-01-2021

<p>Descripción: Como desarrollador quiero verificar que los campos de las credenciales estén llenos para poder realizar la posterior validación de credenciales.</p>
<p>Condiciones de Ejecución:</p> <ul style="list-style-type: none"> • Conexión previa a la base de datos. • Exista las clases pertinentes en la capa de datos. • Exista una interfaz de ingreso de usuario.
<p>Pasos de Ejecución:</p> <ul style="list-style-type: none"> • Ingresar a la página web Custom Learning. • Llenar los campos de las credenciales. • Clic en el botón “Iniciar Sesión”.
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Se visualiza la venta principal del sistema.
<p>Evaluación de la Prueba: EXITOSA</p>

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

La **Tabla 46-3**, muestra una prueba de aceptación, en donde se detallan sus condiciones para ejecución, pasos a seguir y su resultado esperado para determinar si la evaluación de la prueba es exitosa o fallida. En total se tuvieron 262 pruebas de aceptación, resultando todas exitosas, mismas que se detallan en el **Anexo C**. En la **Tabla 47-3**, se detalla el número de pruebas de aceptación por HU y MS.

Tabla 47-3: Numero de pruebas de aceptación por HU/MS

HU/MS	# PA
MS_01	06
MS_02	03
MS_03	05
MS_04	04
MS_05	03
MS_06	03
MS_07	04
MS_08	02
MS_09	03
MS_10	03
HU_01	04
HU_02	08
HU_03	08
HU_04	08

HU/MS	# PA
HU_05	08
HU_06	08
HU_07	08
HU_08	07
HU_09	06
HU_10	06
HU_11	07
HU_12	07
HU_13	08
HU_14	08
HU_15	08
HU_16	08
HU_17	06
HU_18	07
HU_19	07
HU_20	07
HU_21	08
HU_22	08
HU_23	08
HU_24	07
HU_25	08
HU_26	04
HU_27	04
HU_28	07
HU_29	07
HU_30	07
HU_31	07
HU_32	07
Total	262

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

f. Fase de finalización

Esta fase representa las ultimas actividades correspondientes a la puesta en marcha y entrega del sistema desarrollado en el presente proyecto.

➤ Diagrama de despliegue

En la **Figura 19-3**, se muestra la representación del despliegue del sistema a través del diagrama de despliegue, el cual muestra la distribución que tiene cada componente del software en las terminales correspondientes. El usuario mediante el uso de un dispositivo empleando un navegador web, accede al servidor, donde se encuentra alojado el sistema desarrollado, para comunicarse con la interfaz gráfica (compuesta por modelo, vista y controlador), misma que

permite la conexión con el servidor web (WS), alojado como otro componente dentro del mismo servidor, para internamente en este, comunicarse con la lógica del negocio y gestionar el acceso a datos para la comunicación con la base de datos PostgreSQL y realizar el almacenamiento o consulta de información respectiva.

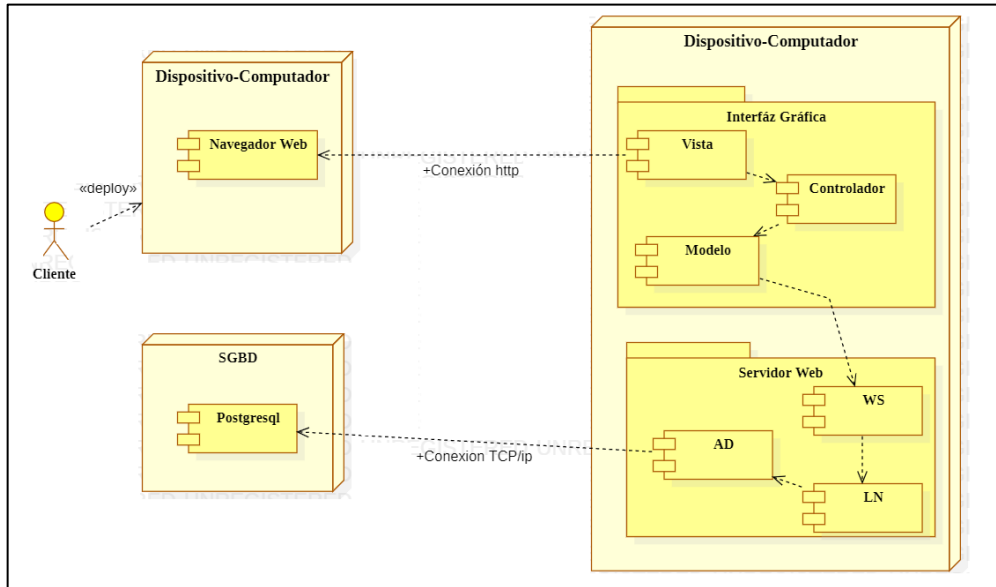


Figura 19-3: Diagrama de despliegue

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

➤ Documentar manual de usuario

El manual de usuario es importante al momento de hacer un sistema ya que establece los pasos específicos para realizar un correcto manejo de las diversas funcionalidades del sistema “Custom Learning”. En dicho documento se detalla paso a paso y con imágenes cada uno de los procesos a seguir para efectuar cada una de las funcionalidades del sistema. Esta metáfora se estableció como una metáfora que tiene dos pruebas de aceptación y una tarea de ingeniería, detalladas en la **Tabla 48-3**.

Tabla 48-3: Metáfora “Documentar el manual de usuario”

METÁFORA DEL SISTEMA	
ID: 10	Nombre de la Historia: Documentar el manual de usuario
Usuario/Rol: Desarrollador	Iteración Asignada: 12
Prioridad del negocio: Muy baja	Puntos estimados: 18
Riesgo de desarrollo: Alta	Puntos reales: 18
Descripción: Como desarrollador quiero documentar el manual de usuario del sistema para que sea una guía del manejo del mismo por parte del usuario.	

Observación: Para documentar el manual se debe tener en cuenta cada aspecto del sistema.
Metáforas (Reverso) Pruebas de Aceptación
<ul style="list-style-type: none"> • Verificar que se la documentación cuente con todos los parámetros establecidos. • Verificar que la información sea clara y se explique lo que se realizó.

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Las demás tablas referentes a la presente metáfora se encuentran detalladas en el **Anexo C**, mientras que el manual de usuario del sistema se encuentra presente en el **Anexo I**.

➤ **Gestión del proyecto**

El presente proyecto tuvo una duración total de 12 iteraciones, de dos semanas laborables cada una y un esfuerzo estimado de 60 puntos que equivalen a 60 horas de trabajo realizadas por dos personas, dando una duración total del proyecto de seis meses, mismos que van desde el 01 de octubre del 2020 hasta el 26 de marzo de 2021. El desarrollo del proyecto se encuentra reflejado en la **Gráfica 5-3**, mostrando el esfuerzo estimado (color azul) y real (color naranja) representado en el eje Y, mientras que en el eje X se indican cada uno de las iteraciones desarrollados.

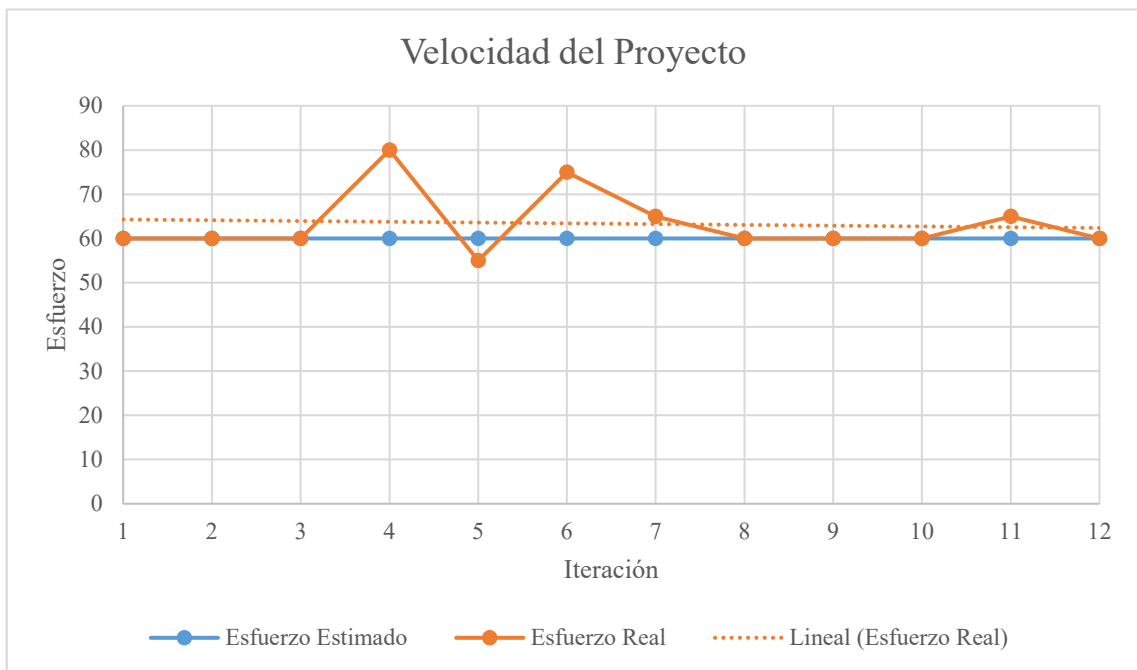


Gráfico 1-3: Velocidad del proyecto

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En base a la **Gráfica15-3**, se puede observar que el proyecto en su mayor parte transcurrió sin problemas, únicamente cinco iteraciones no fueron tal como se las planificó, estas corresponden a la iteración 5 en la cual se tuvo una sobrestimación del esfuerzo a realizar y las iteraciones 4, 6, 7 y 11 en las cuales se subestimó el esfuerzo necesario para completar las tareas. Para corregir

dichos desfares dentro de la planificación, el equipo de desarrollo tuvo que trabajar más tiempo durante las iteraciones que requirieron un mayor trabajo para poder completar las actividades a tiempo. En total se tuvieron 720 puntos de esfuerzo estimado y 760 reales. En base a la línea de tendencia se observa que el proyecto mantuvo un ritmo de desarrollo constante tendiendo a acercarse a lo ideal.

3.3.3. *Proceso para evaluar el rendimiento académico*

Una vez desarrollado el sistema recomendador, se procederá a aplicar y evaluar si con el uso del sistema recomendador “Custom Learning” hay una mejora de rendimiento académico de los estudiantes, para lo cual se sigue los siguientes pasos.

a. Determinación de la población y muestra

La población estuvo conformada por los alumnos de la materia de fundamentos de programación del primer semestre de la carrera de Ingeniería de Software de la ESPOCH durante el periodo académico octubre 2021 – marzo 2022, mismos que fueron divididos en dos grupos que son: **grupo A** de control, conformado por 32 estudiantes del paralelo A, y el **grupo B** experimental, conformado por 30 estudiantes del paralelo B.

b. Creación de contenido personalizado

Para la aplicación del sistema se vio en la necesidad de tener contenidos personalizados en los diversos formatos establecidos en el estudio preliminar del presente trabajo como son: simulación, video y juego, para lo cual se tomó el capítulo de “Estructuras de programación” de la materia “Fundamentos de programación” para realizar los diversos contenidos. El contenido en formato de simulación se lo realizo usando presentaciones PowerPoint interactivas con Visual Basic, mientras que para el contenido en formato de videos se utilizaron imágenes y diagramas para explicar de manera audiovisual el contenido, mismo que se subió a la plataforma YouTube para poder ser visualizada.

En total se desarrollaron 20 contenidos para probar en el sistema (10 para el formato de simulación y 10 para el formato de video) correspondientes a los temas introducción a estructuras de programación, estructuras secuenciales, estructuras condicionales simples, dobles, anidada, múltiple, compuesta, estructuras repetitivas mientras, para y repetir.

Para los estudiantes del grupo de control se les brindara el contenido en formato tradicional a través de documentos en formato pdf.

c. *Aplicación del sistema a los estudiantes*

Se dio acceso al sistema “Custom Learning” de forma online (<https://customlearning.sp.skdrive.net/>) a los estudiantes de los grupos de control y experimental con la finalidad de que puedan interactuar con dicho sistema y su contenido, donde este último se presentaba de diferente forma, siendo así que, si eran estudiante del grupo de control se presentaba el contenido de forma tradicional (documento pdf), pero si eran estudiantes del grupo experimental, el contenido se presentaba en el formato más acorde al estilo de aprendizaje que el estudiante posea.

De esta manera, los estudiantes interactuaron con el sistema durante una semana laboral (lunes-viernes).



Figura 20-3: Sistema “Custom Learning”

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

d. *Evaluación del rendimiento académico*

Una vez finalizado el periodo de interacción de los estudiantes con el sistema, se procede a evaluar el rendimiento académico de ambos grupos para contrastar sus resultados, para lo cual se aplica como herramienta de recolección de datos una evaluación estructurada online (<https://testkolb.000webhostapp.com/>) compuesta de 20 preguntas valoradas entre 0 y 1 cada una, dando como calificación máxima 20 puntos.



Figura 21-3: Evaluación de rendimiento académico online

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

CAPITULO IV

4. RESULTADOS

El presente capítulo tiene como finalidad mostrar los resultados del estudio para determinar la preferencia de contenidos en base a los estilos de aprendizaje (EA) y el estudio para evaluar el rendimiento académico en estudiantes después de emplear el sistema recomendador de contenidos educativos basado en estilos de aprendizaje.

4.1. Resultados de la etapa 1: Determinar la preferencia de contenidos

4.1.1. Análisis de estilos de aprendizaje

En la **Tabla 49-4** se muestra la frecuencia y porcentaje de los estudiantes encuestados de acuerdo a su estilo de aprendizaje.

Tabla 49-4: Análisis de los resultados de estilo de aprendizaje

Estilo de aprendizaje	Frecuencia	Porcentaje
Convergente	19	14.84
Divergente	29	22.66
Asimilador	7	5.47
Acomodador	64	50
Asimilador/Divergente	1	0.78
Acomodador/Divergente	7	5.47
Acomodador/Convergente	1	0.78
Total	128	100

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

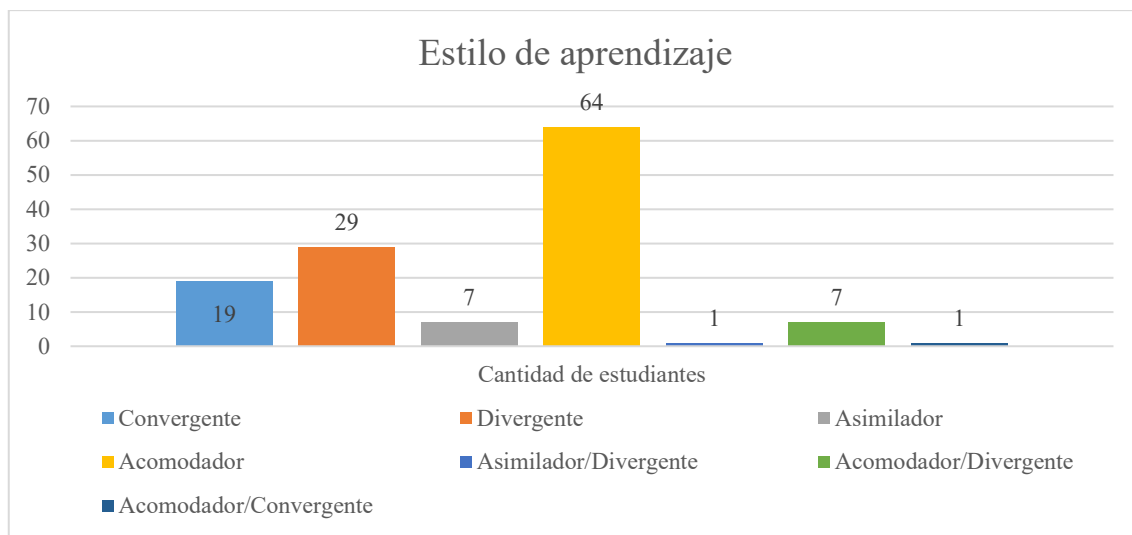


Gráfico 2-3: Gráfica de análisis de los resultados de estilo de aprendizaje

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Los resultados mostrados en la **Tabla 49-4** e ilustrados en el **Gráfico 2-3**, indican que la mayoría de los estudiantes pertenecen al grupo del estilo de aprendizaje acomodador (50%), mientras que la minoría corresponden a aquellos que poseen una combinación de estilos.

4.1.2. *Análisis de preferencia de contenido*

En la **Tabla 50-4**, se muestran los resultados de preferencia de contenido por parte de cada estilo de aprendizaje.

Tabla 50-4: Resultados de preferencia de formato de contenido por cada estilo de aprendizaje

Estilo de aprendizaje	Formato de contenido					
	Texto	Imagen	Audio	Video	Simulación	Juego
Convergente	2	5	0	6	6	5
Divergente	8	4	2	10	16	8
Asimilador	1	1	0	4	1	2
Acomodador	13	9	3	20	30	26

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

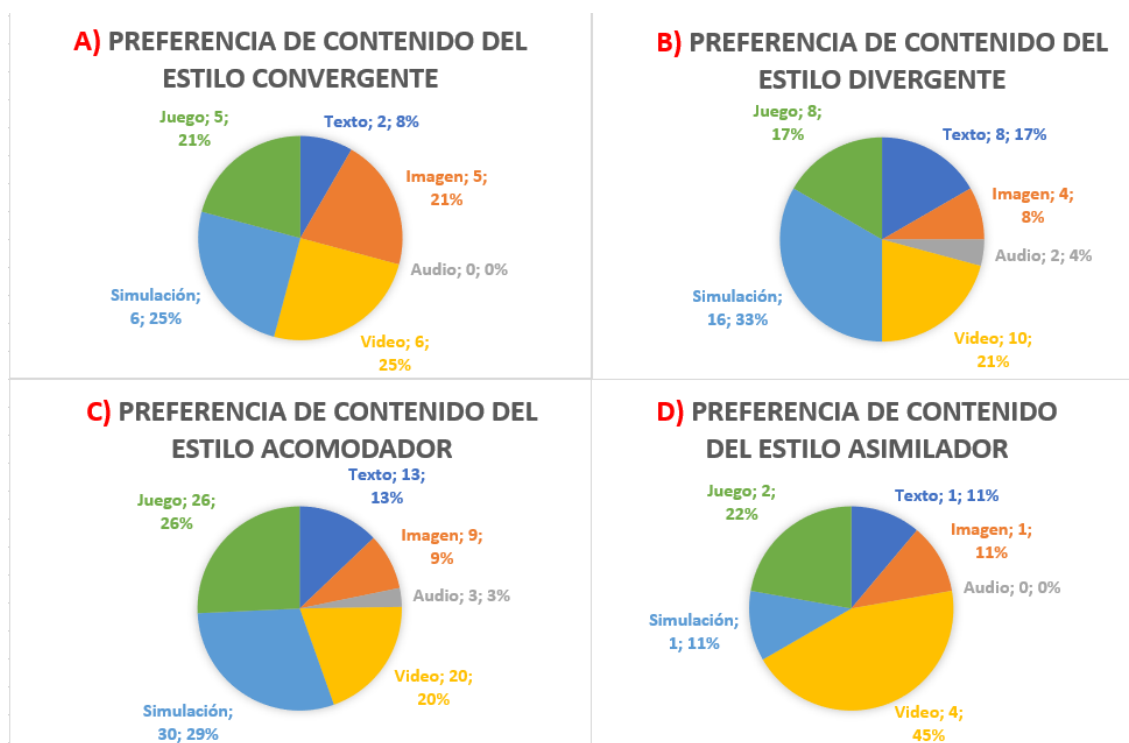


Gráfico 3-4: Gráfica de preferencia de formato de contenido por cada estilo de aprendizaje

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Los resultados de este estudio, basado en encuestas, presentaron diferencias con respecto al estudio realizado por (Sahabudin y Bilal, 2013, p.715). Como se muestra en la **Tabla 50-4**, para los estudiantes del grupo convergente, hubo un empate con un total de 6 (25%) para el contenido de tipo video y simulación tal como se visualiza en el **Gráfico 3-4 A**, en cuyo caso se procedió a analizar la pregunta de verificación (pregunta 18) de la encuesta para determinar con el contenido que se asignará a dicho estilo de aprendizaje, siendo en este caso el contenido de video. De igual manera se analizó al grupo divergente, el cual dio como resultado una preferencia hacia el contenido de simulación con un total de 16 (33.33%) tal como se muestra en el **Gráfico 3-4 B**. El grupo asimilador al igual que el grupo convergente mostro preferencia hacia el contenido de video con un total de 4 (44.44%), resultados que se ilustran en el **Gráfico 3-4 D**. Finalmente, los resultados del grupo acomodador mostraron que, al igual que el grupo divergente, tienen preferencia hacia el contenido de formato simulación con un total de 30 (29.7%) como se muestra en el **Gráfico 3-4 C**.

Además, se analizó si los estudiantes al final de revisar su material en el formato correspondiente, se les muestre unas preguntas no calificadas para reforzar dicho conocimiento como se muestra en la **Tabla 51-4**.

Tabla 51-4: Resultados de preferencia a sección de preguntas adicional al contenido

Estilo de aprendizaje	Si	No	Total
Convergente	24	00	24
Divergente	44	04	48
Asimilador	05	04	09
Acomodador	96	5	101

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

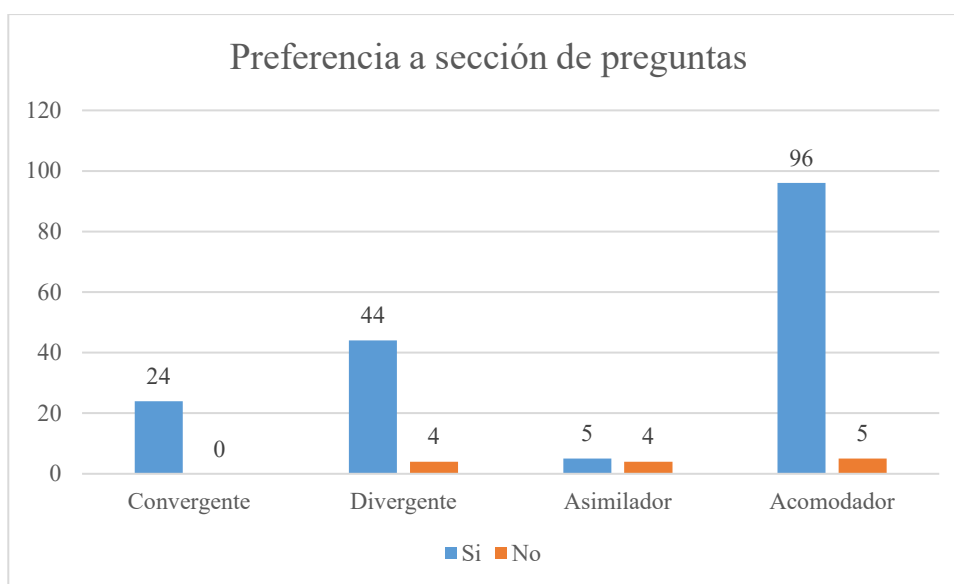


Gráfico 4-4: Gráfica de preferencia a sección de preguntas

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Como se muestra en la **Tabla 51-4** e ilustra en el **Gráfico 4-4**, se puede observar que todos los grupos si prefieren contar con dicho material adicional para reforzar sus conocimientos con resultados favorables de 96 (95.05%) de 101 para el grupo acomodador, 24 (100%) de 24 para el grupo convergente, 5 (55,56%) de 9 para el grupo asimilador y 44 (91.67%) de 48 para el grupo divergente.

Otro análisis similar fue el determinar si la preferencia del tipo de imagen en caso de salir este formato elegido, siendo las opciones imágenes estáticas o animaciones. Los resultados de este análisis se muestran en la **Tabla 52-4** e ilustran en el **Gráfico 5-4**, mostrando que todos los grupos prefieren animaciones sobre imágenes estáticas.

Tabla 52-4: Resultados de preferencia entre imágenes estáticas o animaciones

Estilo de aprendizaje	Animación	Imagen estática	Total
Convergente	21	3	24
Divergente	35	13	48
Asimilador	6	3	9
Acomodador	88	13	101

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

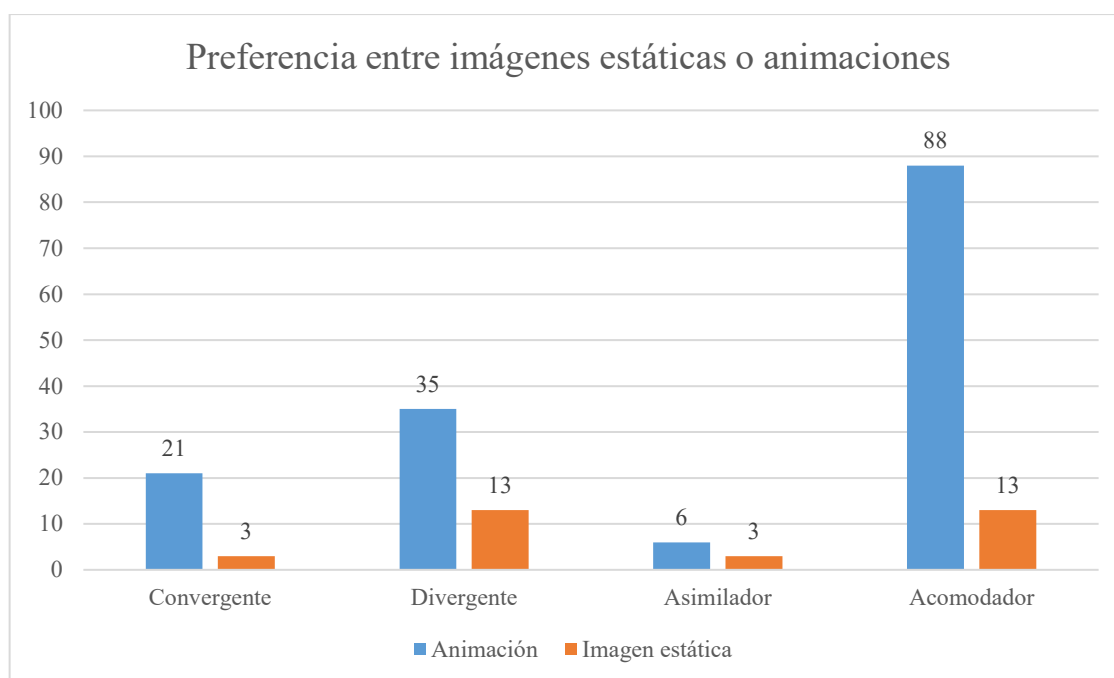


Gráfico 5-4: Gráfica de preferencia entre imágenes estáticas o animaciones

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En conclusión, los resultados obtenidos muestran que los estudiantes de los grupos convergente y asimilador seleccionaron video como su opción de formato de contenido, mientras que los estudiantes de los grupos divergente y acomodador prefirieron el formato de simulación. A más de eso, se seleccionó como un formato de contenido adicional a aquel que ocupa el segundo lugar de preferencia en este estudio, teniendo que el grupo convergente prefiere simulación, el grupo divergente video y los grupos asimilador y acomodador juego como formato de contenido secundario. Dichos resultados se muestran descritos en la **Tabla 53-4**.

Tabla 53-4: Relación entre estilo de aprendizaje con formato de contenido educativo

Estilo de aprendizaje	Formato de Contenido 1	Formato de Contenido 2
Convergente	Video	Simulación
Divergente	Simulación	Video
Asimilador	Video	Juego
Acomodador	Simulación	Juego

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

4.2. Resultados del estudio de evaluación del rendimiento académico

Para medir el criterio de rendimiento académico de los estudiantes, se emplea el análisis descriptivo e inferencial, donde su propósito es medir los resultados de los estudiantes frente a una evaluación sobre los temas revisados y contrastar dichos resultados entre los grupos de estudio, para analizar si hubo o no una mejora del rendimiento académico en aquellos que emplearon el sistema.

4.2.1. Análisis descriptivo

Para el estudio se empleó dos grupos, un grupo de control denominado Grupo A y un grupo experimental denominado Grupo B, conformados por 32 estudiantes del paralelo A y 30 del paralelo B, respectivamente, de la materia “Fundamentos de programación” de la carrera de Ingeniería en Software de la ESPOCH. Para la recolección de datos se empleó una evaluación online (<https://testkolb.000webhostapp.com/>) compuesta por 20 preguntas de selección múltiple con una calificación máxima de 20 (1 punto por cada pregunta). En la **Tabla 54-4** se muestran los resultados obtenidos por ambos grupos en la evaluación anteriormente mencionada.

Tabla 54-4: Resultados del rendimiento académico

Grupo A (Control)		Grupo B (Experimental)	
Estudiante	Calificación	Estudiante	Calificación
1	14,55	1	16,33
2	11,13	2	20,00
3	17,00	3	14,28
4	19,00	4	14,64
5	16,44	5	13,00
6	14,53	6	15,06
7	16,33	7	9,62
8	11,14	8	12,67
9	12,33	9	16,00

Grupo A (Control)	
Estudiante	Calificación
10	14,87
11	11,73
12	9,66
13	17,86
14	13,43
15	9,73
16	9,77
17	16,00
18	13,33
19	13,99
20	10,46
21	11,28
22	12,93
23	10,28
24	15,33
25	9,67
26	10,86
27	7,87
28	9,67
29	14,00
30	12,14
31	12,99
32	7,50

Grupo B (Experimental)	
Estudiante	Calificación
10	16,93
11	14,00
12	12,10
13	13,00
14	13,33
15	13,20
16	14,94
17	16,93
18	14,33
19	19,00
20	18,00
21	17,60
22	13,28
23	15,93
24	15,93
25	16,68
26	18,13
27	16,34
28	18,59
29	19,00
30	19,30

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En base a los resultados mostrados en la **Tabla 54-4**, se realizó un análisis de los mismos obteniendo de dicha manera su valor mínimo, máximo, media y desviación estándar, valores que nos ayudan a tener una mejor comprensión de los resultados obtenidos. Dichos valores se muestran en la **Tabla 55-4**.

Tabla 55-4: Análisis de los resultados para evaluar el rendimiento académico

Grupo	Nota mínima	Nota máxima	Media	Desviación estándar
A	7,50	19,00	12,74	2,92
B	9,62	20,00	15,60	2,50

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En la **Gráfica 6-4** se muestra el análisis de los dos grupos de estudiantes en base a su media de calificación.

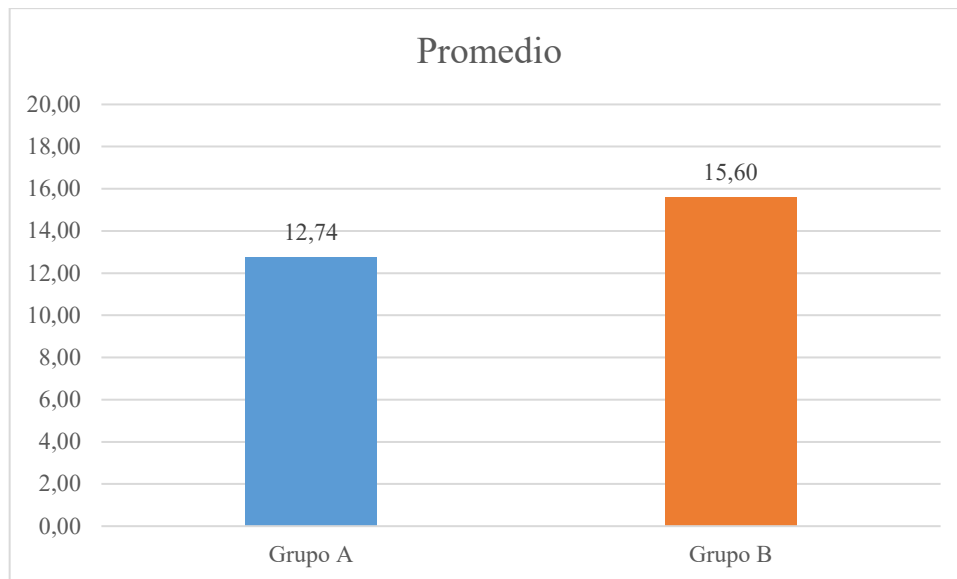


Gráfico 6-4: Análisis descriptivo de los datos

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En la **Gráfica 6-4**, se puede observar que en base a las medias de los datos recolectados el grupo B, al cual se le aplicó el sistema recomendador, muestra que tuvo un mejor rendimiento académico al tener una calificación promedio de 15,60 / 20, mientras que el grupo A obtuvo una calificación promedio de 12,74 / 20. Consecuentemente, se puede decir que aplicando un sistema recomendador los estudiantes presentan una mejora significativa en su rendimiento académico.

4.2.2. *Análisis inferencial*

Para corroborar los datos obtenidos se procedió a emplear test estadísticos para su análisis, donde en primera instancia se efectúa un análisis sobre la normalidad de los datos para establecer la prueba estadística a emplear, que en este caso es t-student para muestras independientes.

➤ *Hipótesis de estudio*

Las hipótesis que se manejan dentro del presente estudio son:

- **H0:** El uso del sistema recomendador no mejora el rendimiento académico de los estudiantes.
- **H1:** El uso del sistema recomendador permite mejorar el rendimiento académico de los estudiantes.

➤ *Distribución de datos*

En el presente análisis se obtuvieron un total de 62 datos (32 datos del grupo de control y 30 datos del grupo experimental), por lo que para comprobar la normalidad de estos se optó por realizar la prueba de Shapiro-Wilk empleando la herramienta SPSS.

El nivel de significancia manejado para el estudio es de $\alpha = 0,05$, ya que es considerado un mínimo valor para el margen de error, lo que garantiza un nivel de confiabilidad del 95%.

Tabla 56-4: Prueba de normalidad de Shapiro-Wilk

Pruebas de normalidad				
Parámetro	Grupos	Shapiro-Wilk		
		Estadístico	gl	Sig. (p)
Nota obtenida en la evaluación	Grupo A (Control)	,976	32	,690
	Grupo B (Experimentación)	,976	30	,709

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En base a la **Tabla 56-4**, se puede observar que el nivel de significancia (p) para el grupo A es de 0.690 y para el grupo B es de 0.709, donde ambos valores p son mayores a nuestro $\alpha = 0,05$, confirmando así que las muestras de datos recabadas siguen una distribución normal.

➤ *Prueba t-student para muestras independientes*

Empleando la herramienta SPSS se procedió a efectuar el análisis estadístico con la prueba t-student para muestras independientes, que permitirá discernir si los resultados obtenidos son significativamente diferentes o no.

En la **Tabla 57-4** se puede visualizar un análisis de estadística de grupo de los datos estudiados, teniendo el valor de la media, la desviación estándar y error estándar, valores que, conjuntamente con la interpretación de los resultados de la prueba t-student de la **Tabla 58-4**, nos permiten determinar la existencia o no de una diferencia significativa, teniendo un nivel de confianza de 95%.

Tabla 57-4: Estadísticas de grupo de los datos analizados

Estadísticas de grupo					
Parámetro	Grupos	N	Media	Desviación estándar	Media de error estándar
Nota obtenida en la evaluación	Grupo A (Control)	32	12,7438	2,91581	,51545
	Grupo B (Experimentación)	30	15,6047	2,50109	,45663

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

Tabla 58-4: Análisis estadístico t-student para muestras independientes

Prueba de muestras independientes										
		Prueba de Levene de calidad de varianzas		prueba t para la igualdad de medias						
		F	Sig.	t	gl	Sig. (p)	Diferencia de medias	Diferencia de error estándar	95% de intervalo de confianza de la diferencia	
									Inferior	Superior
Nota obtenida en la eval.	Se asumen varianzas iguales	,877	,353	-4,134	60	0,000112	-2,86092	,69207	-4,24526	-1,47657
	No se asumen varianzas iguales			-4,155	59,546	0,000106	-2,86092	,68862	-4,23858	-1,48325

Realizado por: Almache, Dayana.; Jácome, Kevin., 2022

En base al resultado de la **Tabla 58-4**, la significancia (p) presenta un valor de 0,000112, el cual es menor a 0.05, comprobando que si existe una diferencia significativa entre las medias de los datos recabados con un 95% de confianza.

Por lo tanto, analizando las medias de los datos, el grupo experimental obtuvo un mayor rendimiento (15.60) que el grupo de control (12.74), mostrando una mejora del 14.3%. Consecuentemente, se concluye que el uso de un sistema recomendador de contenido educativo SI permite mejorar rendimiento académico de los estudiantes.

CONCLUSIONES

- Los sistemas recomendadores asignan ítems a los usuarios en base a uno o varios criterios específicos, y poseen las siguientes características: tener autonomía de toma de decisiones en base a la información disponible, filtrar la información empleando un método y técnica específicos a ser aplicados en un dominio determinado. De este modo, el sistema desarrollado en el presente trabajo corresponde a un recomendador basado en contenido, el cual lleva a cabo las sugerencias de los ítems en base a las necesidades y preferencias del usuario obtenidas a través de su estilo de aprendizaje.
- Los estilos de aprendizaje son la manera en la que cada individuo percibe y procesa la información para convertirla en conocimiento, de este modo, varios autores han establecido diversos modelos con el fin de orientar y potenciar el proceso de aprendizaje. Para el presente proyecto se empleó el modelo propuesto por David Kolb, donde, basándonos en el test de estilos de aprendizaje propuesto por dicho autor, se desarrolló una encuesta online como herramienta de recolección de datos, para identificar el estilo de aprendizaje y la preferencia de contenidos educativos de los estudiantes. Dichos resultados mostraron que los grupos Convergente y Asimilador prefieren el contenido “video”, mientras que los grupos Divergente y Acomodador prefieren el contenido “simulación”.
- El desarrollo del sistema se efectuó empleando la metodología XP, la cual brinda flexibilidad a cambios. Se implementó una arquitectura de n-capas siguiendo el patrón MVC empleando el framework Primefaces para interfaz, con la finalidad de tener un diseño que facilite el entendimiento, manejo y posterior mantenimiento del sistema. Dicho desarrollo tuvo una duración de 12 iteraciones de dos semanas cada una, en donde se desarrolló un total de 10 metáforas del sistema y 32 historias de usuario, lo cual permitió dar cumplimiento a los requerimientos para el sistema.
- Analizando los resultados de la evaluación realizada por parte de los estudiantes al finalizar el período de interacción con el sistema y sus contenidos, se obtuvo que, el grupo de control tuvo un promedio de 12.74 / 20 (63.7%), mientras que el grupo de experimentación obtuvo 15.60 / 20 (78%), por lo cual se determinó que, aplicando un sistema recomendador de contenidos educativos basado en estilos de aprendizaje mejora el rendimiento académico de los estudiantes en un 14.3%. Aplicando la prueba t-student para muestras no relacionadas con un nivel de confianza de 95% se obtiene un valor p de 0,000112, indicando así que, los datos de las medias poseen una diferencia significativa, siendo el valor mayor el correspondiente a la media de los estudiantes del grupo experimental de este trabajo.
- Se programó el sistema recomendador de contenidos educativos basados en estilos de aprendizaje denominado “Custom Learning” utilizando el lenguaje de programación Java y

el framework Primefaces. El sistema se lo desplegó en un host con servidor Glashfish 5.1.0 y PostgreSQL 12.3 con la finalidad de que los usuarios puedan acceder e interactuar con el contenido alojado en dicho sistema para posteriormente realizar la recolección de datos para este trabajo.

RECOMENDACIONES

- Si bien el manejo y navegación por el sistema de recomendación “Custom Learning” es fácil e intuitivo, se recomienda a los usuarios que vayan a emplear el sistema, que revisen el manual de usuario, mismo que presenta información detallada para poder manejar adecuadamente el sistema y cada una de sus respectivas funcionalidades.
- A pesar de que el contenido cargado es suficiente para el correcto funcionamiento del sistema, permitiendo cumplir con los objetivos del presente trabajo, se recomienda continuar cargando contenido dentro del sistema para a futuro poder contar con un repositorio de contenido educativo personalizado en base a los estilos de aprendizaje de los estudiantes.
- Se analizó 8 tipos de formatos para el contenido, de los cuales se seleccionaron 4 para el desarrollo del presente trabajo, sin embargo, se recomienda diseñar el contenido en varios formatos, para que de esta manera los usuarios puedan tener más alternativas a la hora de escoger el que más les atraiga en base a la recomendación realizada.
- Aunque los módulos del sistema diseñados para el presente trabajo son suficientes para un correcto funcionamiento, se recomienda que se desarrollen e integren nuevos módulos que permitan añadir nuevas funcionalidades y mejorar la experiencia de usuario, tal es el caso del módulo de evaluaciones que actualmente se encuentra planteado como trabajos a futuro.
- El módulo de recomendación está diseñado en base a un solo parámetro, correspondiente al estilo de aprendizaje de los estudiantes, permitiendo su correcto funcionamiento, pero se recomienda a futuro analizar y añadir otros parámetros de recomendación para tratar de conseguir un filtrado de ítems a recomendar más preciso y eficiente.

GLOSARIO

Iteración: significa repetir varias veces un proceso con la intención de alcanzar una meta deseada, objetivo o resultado. (Real Academia Española, 2021)

Extreme Programming (XP): es un término que está escrito en inglés que significa programación extrema la misma que es una metodología ágil para el desarrollo de software. (Letelier y Penádes, 2006, p.1-17)

Metáforas del sistema: es una historia que describe cómo funciona el programa. (Letelier y Penádes, 2006, p.12)

Historia de usuario: es una forma para especificar los requisitos del usuario. (Letelier y Penádes, 2006, p.1-17)

Prueba de aceptación: permite verificar que la historia de usuario se cumplió adecuadamente. (Letelier y Penádes, 2006, p.18)

Tarea de ingeniería: en las tareas de ingeniería se almacena información relevante para identificar y detallar la historia de usuario. (Letelier y Penádes, 2006, p.1-23)

BIBLIOGRAFÍA

ACIAR, Gabriela.; & ACIAR, Silvana. “Recomendador de usuarios en una plataforma colaborativa en base a su perfil y reputación”. *14th Argentine Symposium on Artificial Intelligence*. [en línea], 2013, (Argentina) vol. 14, pp. 1-11. [Consulta: 13 octubre 2020]. ISSN: 1850-2784. Disponible en: <http://sedici.unlp.edu.ar/handle/10915/75954>

ACOSTA-QUIROZ, Christian. *Estilos de aprendizaje en estudiantes de psicología de una institución de educación superior del sur de Sonora*. Sonora-México: PEARSON, 2014. ISBN 978-607-32-3021-6, pp. 95–104.

ALMAGUER, Yilena Pérez et al. “Una revisión de los sistemas recomendadores grupales como herramienta innovadora en el área del turismo”. *Revista de Ciencia y Tecnología* [en línea], 2021, pp.51-60. [Consulta: 20 de diciembre de 2021]. Disponible en: http://www.scielo.org/ar/scielo.php?script=sci_arttext&pid=S1851-75872021000100051&lang=pt

CABRERA ALBERT, Juan Silvio.; & FARIÑAS LEÓN, Gloria. “El estudio de los estilos de aprendizaje desde una perspectiva vigostkiana: una aproximación conceptual”. *Revista Iberoamericana de Educación*. [en línea], 2005, (Cuba) vol. 37 (no 1), pp. 1–10. [Consulta: 14 agosto 2020}. ISSN: 1022-6508. Disponible en: <https://rieoei.org/RIE/article/view/2731>

CARMISINI, Andrey.; & VAHL DICK, Adilson. “COMPARATIVO ENTRE FRAMEWORKS DE JSF: APACHE TOBAGO, PRIMEFACES E RICHFACES”. *Revista Eletrônica do Alto Vale do Itajaí*. [en línea], 2012, (Brasil) vol. 1 (no. 2), pp. 10-18. [Consulta: 08 noviembre 2020]. ISSN: 2316-4190. Disponible en: <https://www.revistas.udesc.br/index.php/reavi/article/view/2889>

CASTELLS, Manuel. *Internet y la Sociedad red*. [página web]. [Consulta: 22 agosto 2020]. 2001. Disponible en: <https://www.uoc.edu/web/cat/articles/castells/castellsmain2.html>

CHARNELLI, María Emilia. *Sistemas recomendadores aplicados en educación*. [en línea] (Trabajo de titulación) (Maestría). Universidad Nacional de La Plata, Facultad de Informática. Buenos Aires-Argentina. 2019. pp. 12-65. [Consulta: 15 noviembre 2020]. Disponible en: <http://sedici.unlp.edu.ar/handle/10915/67261>

CLISP. [página web]. 2020. [Consulta: 02 noviembre 2010]. Disponible en: <https://clisp.sourceforge.io/>

CUEVAS, Enrique. *JavaServer Faces (JSF)*. [documento en página web]. 2009. [Consulta: 17 octubre 2020]. Disponible en: https://www.academia.edu/25689987/Manual_Jsf

CUNO SOTO, Carmen Elena. Estilos de aprendizaje en los estudiantes de la Institución Educativa Aymara de Acora 2016. [en línea] (Trabajo de titulación). (Maestría) Universidad Nacional del Altiplano, Facultad de Ciencias de la Educación. Puno-Perú. 2016. pp. 17-50. [Consulta: 11 agosto 2020]. Disponible en: <http://repositorio.unap.edu.pe/handle/UNAP/8944>

DELGADILLO, Rodolfo.; & GUILLÉN, Jessica. "CONCORDANCIAS ENTRE LOS ESTILOS DE APRENDIZAJE Y DE ENSEÑANZA EN ESTUDIANTES UNIVERSITARIOS". *Revista de la Facultad de Ciencias de la Salud*. (2007), (México). pp. 1–17.

DEPTO. CCIA. *Funcionamiento y arquitectura de JSF*. [documento en página web]. 2010. [Consulta: 17 octubre 2020]. Disponible en: <http://www.jtech.ua.es/j2ee/2009-2010/restringido/jsf/sesion02-apuntes.pdf>

DURÁN, Elena B.; & COSTAGUTA, Rosanna N. “Experiencia de Enseñanza Adaptada al Estilo de Aprendizaje de los Estudiantes en un Curso de Simulación”. *Formación universitaria*. [en línea], 2008, (Argentina) vol. 1 (no 1), pp. 19–28. [Consulta: 20 agosto 2020]. ISSN 0718-5006. Disponible en: http://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0718-50062008000100004&lng=es&nrm=iso

EGUÍLUZ PÉREZ, Javier. *Introducción a CSS*. [en línea]. Autoedición, 2009. [Consulta: 21 noviembre 2020]. Disponible en: <https://openlibra.com/es/book/introduccion-a-css>

ESPOCH. “Resolución 286.CP.2020”. *Computers in Human Behavior*. [documento en página web]. 2020. [Consulta: 20 junio 2020]. ISSN: 581–591. Disponible en: http://cimogsys.esepoch.edu.ec/vi/public/download/resoluci%C3%93n_286.cp.2020.pdfR

ESSALMI, Fathi.; et al. “A fully personalization strategy of E-learning scenarios”. *Computers in Human Behavior*. [en línea], 2010, (Canadá) vol. 1 (no. 26), pp. 581-591. [Consulta: 06 octubre 2020]. ISSN: 581–591. Disponible en: https://www.researchgate.net/publication/224282002_A_fully_personalization_strategy_of_E-learning_scenarios

FAIZA, Anwer.; et al. “Comparative Analysis of Two Popular Agile Process Models: Extreme Programming and Scrum”. *International Journal of Computer Science and Telecommunications*. [en línea], 2017, (Pakistán), vol. 8, pp. 1–7. [Consulta: 15 septiembre 2020]. ISSN 2047-3338. Disponible en: <https://www.researchgate.net/publication/316845761>

GALLEGO RODRÍGUEZ, Alejandrino.; & MARTÍNEZ CARO, Eva. “ESTILOS DE APRENDIZAJE Y E-LEARNING. HACIA UN MAYOR RENDIMIENTO ACADÉMICO”.

Revista de Educación a Distancia (RED). [en línea], 2003, (España) vol. 3 (no. 7), pp. 1-10. [Consulta: 17 agosto 2020]. ISSN: 1578-7680. Disponible en: <https://revistas.um.es/red/article/view/25411>

GARCÍA, Alex Estrada. “Estilos de aprendizaje y rendimiento académico”. *Revista Boletín Redipe*. [en línea], 2018, (Ecuador) vol. 7 (no. 7), pp. 218–228. [Consulta: 22 agosto 2020]. ISSN 2256-1536. Disponible en: <https://revista.redipe.org/index.php/1/article/view/536>

GIMENO, Juan.; & GONZÁLEZ, José. *Introducción a Netbeans*. [documento en página web]. 2011. [Consulta: 07 septiembre 2020]. Disponible en: <https://docplayer.es/2050878-Introduccion-a-netbeans.html>

GLORIA, Arcos Medina.; et al. “Comparative Study of Performance and Productivity of MVC and MVVM design patterns”. *KnE Engineering*. [en línea], 2018, (Ecuador) vol. 3 (no. 1), pp. 241-252. [Consulta: 14 octubre 2020]. ISSN: 2519-1330. Disponible en: <https://revistas.um.es/red/article/view/25411>

GÓMEZ FAJARDO, Hebert.; & LARA SILVA, Emigdio. “Diseño de un modelo funcional de gestión soportado en servicios RESTful para gestión integrada de redes y servicios de T-learning”. *Sistemas & Telemática*. [en línea], 2014, (Colombia) vol. 12 (no. 29), pp. 49-65. [Consulta: 18 octubre 2020]. ISSN: 1692-5238. Disponible en: <https://www.redalyc.org/articulo.oa?id=411533999003>

GÓMEZ SÁNCHEZ, David.; et al. “Factores que influyen en el rendimiento académico del estudiante universitario”. *Tecnociencia*. [en línea], 2011, (México) vol. 5 (no. 2), pp. 90-97. [Consulta: 14 agosto 2020]. ISSN: 1870-6606. Disponible en: <https://www.researchgate.net/publication/305335495>

GOSLING, J.; et al. *The Java Language Specification, Java SE 14 Edition*. [en línea]. 14 edición. Oracle, (2020). [Consulta: 29 julio 2020]. Disponible en: <https://docs.oracle.com/javase/specs/>

GUERRERO VERGEL, René. Metodología para la implementación de estilos de aprendizaje en ambientes de formación virtual para adultos. [en línea] (Trabajo de titulación). (Maestría) Universidad Nacional de Colombia. Colombia. 2012. pp. 9-30. [Consulta: 13 agosto 2020]. Disponible en: <https://repositorio.unal.edu.co/handle/unal/20772>

GUTIÉRREZ TAPIAS, Mariano. “Estilos de aprendizaje, estrategias para enseñar. Su relación con el desarrollo emocional y ‘aprender a aprender’”. *Tendencias Pedagógicas*. [en línea], 2018, (España) vol. 3, pp. 83-96. [Consulta: 10 octubre 2020]. ISSN: 1989-8614. Disponible en: <https://revistas.uam.es/tendenciaspedagogicas/article/view/tp2018.31.004>

HERNANDEZ HERRERA, Claudia Alejandra. Diagnóstico del rendimiento académico de estudiantes de una escuela de educación superior en México. [en línea] (Artículo). Instituto Politécnico Nacional. DF-México. 2015. [Consulta: 04 noviembre 2021]. Disponible en: <https://revistas.ucm.es/index.php/RCED/article/view/48551>

HINOJOSA TINOCO, Diego Andrés. Propuesta de arquitectura de software para el desarrollo de aplicaciones empresariales basadas en JEE. [en línea] (Trabajo de titulación) (Grado). Pontificia Universidad del Ecuador. Quito-Ecuador. 2013. pp. 1-21. [Consulta: 04 noviembre 2020]. Disponible en: <http://repositorio.puce.edu.ec/handle/22000/6379>

HUMAM, Majee.; & UFUOMA, Apoki. “A Semantic Approach to Multi-parameter Personalisation of E-Learning Systems”. *Modelling and Development of Intelligent Systems*. [en línea], 2021, (Rumania) vol. 1341, pp. 381-393. [Consulta: 03 marzo 2021]. ISSN: 978-3-030-68527-0. Disponible en: https://doi.org/10.1007/978-3-030-68527-0_24

JAVASERVER FACES TECHNOLOGY. [página web]. México, 2021. [Consulta: 19 octubre 2020]. Disponible en: http://aragorn.pb.bialystok.pl/~dmalyszko/PSS_Project/JavaServer%20Faces.htm

JEONG, Hwa-Young.; et al. “Personalized Learning Course Planner with E-learning DSS using user profile”. *Expert Systems with Applications*. [en línea], 2012, (Korea) vol. 39 (no. 3), pp. 2567-2577. [Consulta: 13 noviembre 2020]. ISSN: 0957-4174. Disponible en: <https://www.sciencedirect.com/science/article/abs/pii/S0957417411012401?via%3Dihub>

JS.FOUNDATION, J.F. *jQuery*. [página web]. [Consulta: 10 agosto 2020]. Disponible en: <https://jquery.com/>

LEÓN GRIJALVA, Mario César. Incidencia del Ciclo de Aprendizaje de Kolb en el razonamiento lógico de las Ciencias Naturales. Estudio de caso con estudiantes de octavo año de básica superior de la Unidad Educativa Salesiana Cardenal Spellman. [en línea] (Trabajo de titulación). (Maestría) Universidad Politécnica Salesiana del Ecuador. Cuenca-Ecuador. 2014. pp. 36-59. [Consulta: 10 junio 2020]. Disponible en: <https://dspace.ups.edu.ec/handle/123456789/7501>

LETELIER, Patricio.; & PENÁDES, Carmen. “Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP)”. *Técnica Administrativa*. [en línea], 2006, (Argentina) vol. 5 (no. 26), pp. 1-17. [Consulta: 20 noviembre 2020]. ISSN: 1666-1680. Disponible en: <https://dialnet.unirioja.es/servlet/articulo?codigo=1983605>

MALIK, Sehrish.; & KIM, Do-Hyeun. “A comparison of RESTful vs. SOAP web services in actuator networks”. *Ninth International Conference on Ubiquitous and Future Networks*

(ICUFN). [en línea], 2017, (Korea) vol. 9, pp. 753-755. [Consulta: 13 octubre 2020]. ISSN: 2165-8536. Disponible en: <https://ieeexplore.ieee.org/document/7993893>

MATOS, Viviana. *Sistemas de Recomendación y Personalización*. [documento en página web]. Buenos Aires-Argentina, 2006. [Consulta: 16 octubre 2020]. Disponible en: <https://docplayer.es/9004819-Arquitectura-de-un-sistema-recomendador.html>

MENESES, E. *Tecnologías para la integración de soluciones*. [documento en página web]. México, 2017. [Consulta: 01 agosto 2020]. Disponible en: <https://www.uv.mx/personal/ermeneses/files/2017/03/TISClase22.pdf>

MINTEL. [página web]. Ecuador, 2013. [Consulta: 19 octubre 2020]. Disponible en: <https://www.telecomunicaciones.gob.ec/wp-content/uploads/2019/05/PPT-Estrategia-Ecuador-Digital.pdf>

MOLINA RIOS, Jimmy.; et al. “Comparación de metodologías en aplicaciones Web”. *3C Tecnología*. [en línea], 2018, (España) vol. 7 (no. 1), pp. 1-19. [Consulta: 11 agosto 2020]. ISSN: 2254 – 4143. Disponible en: <https://ojs.3ciencias.com/index.php/3c-tecnologia/article/view/596>

MOROCHO GUASHPA, William Patricio.; & CARRANZA LUCAS, Jerson Kenny. Desarrollo de un sistema web de gestión financiera para la fundación “Jaspe” utilizando la tecnología de base de datos No-SQL. [en línea] (Trabajo de titulación) (Grado). Escuela Superior Politécnica de Chimborazo, Facultad de Informática y Electrónica, Escuela de Ingeniería en Sistemas. Riobamba-Ecuador. 2017. pp. 21-69. [Consulta: 21 noviembre 2020]. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/7462>

MUMBAIKAR, Snehal.; & PADIYA, Puja. “Web Services Based On SOAP and REST Principles”. *International Journal of Scientific and Research Publications*. [en línea], 2013, (India) vol. 3 (no. 5), pp. 1-4. [Consulta: 12 octubre 2020]. ISSN: 2250-3153. Disponible en: <http://citeseerx.ist.psu.edu/viewdoc/summary;jsessionid=3E9B2689A47E22C4F7617017386D52DA?doi=10.1.1.414.7936>

NAVARRO CADAVID, Andrés.; et al. “Revisión de metodologías ágiles para el desarrollo de software”. *Prospectiva*. [en línea], 2013, (Colombia) vol. 11 (no. 2), pp. 30-39. [Consulta: 20 agosto 2020]. ISSN 1692-8261. Disponible en: <https://dialnet.unirioja.es/servlet/articulo?codigo=4752083>

ORACLE. [página web]. México, 2019. [Consulta: 16 noviembre 2020]. Disponible en: <https://www.oracle.com/mx/tools/technologies/netbeans-ide.html>

PAIK. Hye-young.; et al. “Web Services – REST or Restful Services”. *Web Service Implementation and Composition Techniques*. [en línea], 2017, (Australia) vol. 3, pp. 67-91.

[Consulta: 14 octubre 2020]. ISBN: 978-3-319-55542-3. Disponible en: https://www.researchgate.net/publication/318142185_Web_Services_-_REST_or_Restful_Services

PANTOJA OSPINA, Martín Alonso; et al. “Modelos de estilos de aprendizaje: una actualización para su revisión y análisis”. *Revista Colombiana de Educación*. [en línea], 2013, (Colombia) vol. 1 (no. 64), pp. 79-105. [Consulta: 13 noviembre 2020]. ISSN: 0120-3916. Disponible en: http://www.scielo.org.co/scielo.php?pid=S0120-39162013000100004&script=sci_abstract

PAZMIÑO HARO, Anthony Ismael; & TORRES ARROBA, Grecia Aracelly. Desarrollo de un sistema web de apoyo para las áreas de Psicología y Becas en la Dirección de Bienestar Estudiantil de la ESPOCH. [en línea] (Trabajo de titulación) (Grado). Escuela Superior Politécnica de Chimborazo, Facultad de Informática y Electrónica, Escuela de Ingeniería en Sistemas. Riobamba-Ecuador. 2018. pp. 9-36. [Consulta: 04 agosto 2020]. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/9277>

POSTGRESQL. [página web]. 2020. [Consulta: 03 septiembre 2020]. Disponible en: <https://www.postgresql.org/>

PRIMEFACES. [página web]. 2020. [Consulta: 03 noviembre 2010]. Disponible en: <https://www.primefaces.org/>

RIOS LUNA, Angela. IMPLEMENTACIÓN DEL PATRÓN MVC EN APLICACIONES WEB CON JAVA MEDIANTE LA INTEGRACIÓN DE LOS FRAMEWORK HIBERNATE, SPRING Y PRIMEFACES. [en línea] (Trabajo de titulación). (Licenciatura). Universidad Autónoma del Estado de México. México. 2014. pp. 13-53. [Consulta: 13 agosto 2020]. Disponible en: <http://ri.uaemex.mx/handle/20.500.11799/62542>

RODRÍGUEZ CEPEDA, Rodrigo. “Los modelos de aprendizaje de Kolb, Honey y Mumford: implicaciones para la educación en ciencias”. *Sophia*. [en línea], 2017, (Colombia) vol. 14 (no. 1), pp. 51-64. [Consulta: 01 septiembre 2020]. ISSN: 2346-0806. Disponible en: <https://doi.org/10.18634/sophiaj.14v.1i.698>

RÚALES PINTADO, Ernesto Nelson; & SUAREZ QUISPE, Geomara Abigail. Análisis, diseño y desarrollo de un prototipo de endpoint para mejorar la producción de web services. [en línea] (Trabajo de titulación) (Grado). Universidad de Guayaquil. Guayaquil-Ecuador. 2015. pp. 24-32. [Consulta: 10 agosto 2020]. Disponible en: <http://repositorio.ug.edu.ec/handle/redug/10766>

SAHABUDIN, Noor.; & BILAL ALI, Mohamad. “Personalized Learning and Learning Style among Upper Secondary School Students”. *Procedia - Social and Behavioral Sciences*. [en línea], 2013, (Malasia) vol. 103, pp. 710-716. [Consulta: 16 noviembre 2020]. ISSN: 1877-0428. Disponible en:

https://www.researchgate.net/publication/275538821_Personalized_Learning_and_Learning_Style_among_Upper_Secondary_School_Students

SALVADOR SÁNCHEZ, Miguel A. *Conceptos básicos Fundamentos de programación*. [documento en página web]. Universidad de Alcalá, 2014. [Consulta: 11 agosto 2020]. Disponible en: <https://www.docsity.com/es/conceptos-basicos-148-5/3383171/>

SARI VILLA, Oscar David. Diseño e implementación de un Sistema Recomendador de actividades lúdicas y culturales para personas de la tercera edad. [en línea] (Trabajo de titulación) (Grado). Universidad Politécnica Salesiana. Cuenca-Ecuador. 2017. pp. 5-45. [Consulta: 11 noviembre 2020]. Disponible en:

<https://dspace.ups.edu.ec/handle/123456789/13469>

SILVA SPROCK, Antonio. “Conceptualización de los Modelos de Estilos de Aprendizaje”. *Revista De Estilos De Aprendizaje*. [en línea], 2018, (Venezuela) vol. 11 (no. 21), pp. 35-109. [Consulta: 05 septiembre 2020]. ISSN: 1988-8996. Disponible en:

<http://revistaestilosdeaprendizaje.com/article/view/1088>

TIHOMIROVS, Juris.; & GRABIS, Janis. “Comparison of SOAP and REST Based Web Services Using Software Evaluation Metrics”. *Information Technology and Management Science*. [en línea], 2016, (Letonia) vol. 19 (no. 1), pp. 92-97. [Consulta: 13 octubre 2020]. ISSN: 2255-9094. Disponible en:

https://www.researchgate.net/publication/312566917_Comparison_of_SOAP_and_REST_Based_Web_Services_Using_Software_Evaluation_Metrics

TIOBE. *The Software Quality Company*. [página web]. [Consulta: 22 julio 2020]. Disponible en: <https://www.tiobe.com/tiobe-index/>

TRUJILLO CRUZ, Andrea Estefanía.; & VEGA CÁRDENAS, Rosa Marcela. Evaluación de los estilos de aprendizaje de los adolescentes de 15 y 16 años en situación de vulnerabilidad social del Centro de Formación Artesanal Fiscomisional - TESPA. [en línea] (Trabajo de titulación) (Grado). Universidad Politécnica Salesiana. Quito-Ecuador. 2015. pp. 3-9. [Consulta: 29 julio 2020]. Disponible en: <https://dspace.ups.edu.ec/handle/123456789/9579>

VALAREZO PARDO, Milton Rafael.; et al. “COMPARACIÓN DE TENDENCIAS TECNOLÓGICAS EN APLICACIONES WEB”. *3C Tecnología*. [en línea], 2018, (España) vol.

7 (no. 3), pp. 28-49. [Consulta: 16 agosto 2020]. ISSN: 2254 – 4143. Disponible en: <https://ojs.3ciencias.com/index.php/3c-tecnologia/article/view/618>

VEGAS, Emiliana.; & PETROW, Jenny. *Incrementar el aprendizaje estudiantil en América Latina: el desafío para el siglo XXI.* América Latina: Mayol Ediciones, 2008. ISBN 978-958-8307-31-2, pp. 5-37.

VENTURA SEMINARIO, Rossana Inés.; et al. El uso de las TIC y su relación con el rendimiento académico en el área de inglés en los estudiantes del quinto grado de secundaria de la Institución Educativa San Luis Gonzaga, Ica, 2014. [en línea] (Trabajo de titulación) (Maestría). Universidad Nacional de Educación Enrique Guzmán y Valle. Lima-Perú. 2017. pp. 12-40. [Consulta: 19 julio 2020]. Disponible en: <https://repositorio.une.edu.pe/handle/UNE/1182>

VERA REA, Pablo Alejandro.; & ULLOA AMAYA, Marlon Enrique. Diseño y desarrollo de un sistema recomendador de contenidos accesibles basados en perfiles de usuarios para ambientes virtuales y objetos de aprendizaje a partir de metadatos de accesibilidad haciendo uso de ontologías. [en línea] (Trabajo de titulación) (Grado). Universidad Politécnica Salesiana. Cuenca-Ecuador. 2018. pp. 46-90. [Consulta: 12 noviembre 2020]. Disponible en: <https://dspace.ups.edu.ec/handle/123456789/16350>

VICUÑA LOJANO, César Eduardo. Diseño e implementación de un sistema recomendador de eventos de interés para personas de la tercera edad mediante geolocalización. [en línea] (Trabajo de titulación) (Grado). Universidad Politécnica Salesiana. Cuenca-Ecuador. 2018. pp. 2-7. [Consulta: 12 noviembre 2020]. Disponible en: <https://dspace.ups.edu.ec/handle/123456789/15185>

VILLARREAL FUENTES, César Amauri. *¿Qué es jQuery?.* [documento en página web]. Northware, México, 2013. [Consulta: 13 agosto 2020]. Disponible en: <https://www.northware.mx/wp-content/uploads/2021/04/que-es-jquery.pdf>

ZEA SILVA, Liced Angélica.; et al. *Aprendizaje, cognición y mediaciones en la escuela. Una mirada desde la investigación de instituciones educativas del Distrito Capital.* Bogotá-Colombia: Rocca® S. A, 2016. [Consulta: 16 septiembre 2020]. Disponible en: <https://repositorio.idep.edu.co/handle/001/119>