



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA EN SISTEMAS

**“ANÁLISIS COMPARATIVO DE METRO Y AXIS2 PARA EL
DESARROLLO DE APLICACIONES QUE CONSUMAN SERVICIOS WEB**

WCF EN LA ESPOCH”

“TESIS DE GRADO PREVIA OBTENCIÓN DEL TÍTULO DE

INGENIERO EN SISTEMAS INFORMÁTICOS”

LUIS GUSTAVO PATIÑO CASTILLO

RIOBAMBA – ECUADOR

-2014-

AGRADECIMIENTO

Agradezco a Dios por ser el pilar fundamental en la consecución de mis objetivos, a mis padres por su eterno apoyo, amor, comprensión y por ser guías en el sendero de la vida, a mi hermano y hermanas por su entusiasmo y motivación, a mis profesores y amigos por creer y respaldar mi proyecto y un agradecimiento especial a mi directora de tesis Ing. Gloria Arcos y miembro de tesis Ing. Washington Luna por ser guías en el desarrollo de mi tesis.

DEDICATORIA

Dedico este trabajo a mis padres Darío y Dolores, a mi hermano Adrián, a mis hermanas Lourdes y Blanca quienes con amor y comprensión estuvieron junto a mí durante mi formación académica permitiendo así llegar a la meta.

A mis amig@s Pamela y Fabián, por su valiosa amistad y motivación

FIRMAS RESPONSABLES Y NOTAS

NOMBRES

FIRMA

FECHA

ING. IVÁN MENES CAMEJO

**DECANO DE LA FACULTAD DE
INFORMÁTICA Y ELECTRÓNICA.**

ING. JORGE HUILCA

**DIRECTOR DE LA ESCUELA DE
INGENIERÍA EN SISTEMAS.**

ING. GLORIA ARCOS

DIRECTOR DE TESIS.

ING. WASHINGTON LUNA

MIEMBRO DE TESIS.

**DIRECTOR DEL CENTRO DE
DOCUMENTACIÓN.**

NOTA: _____

RESPONSABILIDAD DEL AUTOR

Yo, Luis Gustavo Patiño Castillo soy el responsable de las ideas, doctrinas y resultados expuestos en esta Tesis y el patrimonio intelectual de la misma pertenecen a la Escuela Superior Politécnica de Chimborazo.

Luis Gustavo Patiño Castillo

ÍNDICE DE ABREVIATURAS Y ACRÓNIMOS

ADB	Axis2 DataBinding
API	Application Programming Interface
DTIC	Dirección de Tecnologías de la Información y Comunicación
ESPOCH	Escuela Superior Politécnica de Chimborazo
FTP	File Transfer Protocol
GB	Gigabyte
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
IIS	Internet Information Server
JAX	Java API for XML
JAXB	Java Architecture for XML Binding
JDK	Java Development Kit
JSF	Java Server Face
MVC	Model View Controller
OASIS	Advancing Open Standards for the Information Society
OASIS	Orion Academic System with Internet Services
PDF	Portable Document Format
POJO	Plain Old Java Objects

RAM	Random Access Memory
RPC	Remote Procedure Call
SAAJ	SOAP with Attachments API for Java
SOAP	Simple Object Access Protocol
SSL	Security Sockets Layer
UDDI	Universal Discovery, Description and Integration
VM	Virtual Machine
WCF	Windows Communication Foundation
WS	Web service
WSDL	Web Services Definition Language
WSIT	Web Service Interoperability Technologies
XHTML	eXtensible HyperText Markup Language
XML	eXtensible Markup Language
XP	Extreme Programming
XSD	XML Schema

ÍNDICE GENERAL

PORTADA

AGRADECIMIENTO

DEDICATORIA

FIRMAS RESPONSABLES

RESPONSABILIDAD DE LOS AUTORES

ÍNDICE DE ABREVIATURAS Y ACRÓNIMOS

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

INTRODUCCIÓN

CAPITULO I

MARCO REFERENCIAL.....	20
1.1 Antecedentes	20
1.2 Justificación.....	23
1.2.1 Justificación Teórica.....	23
1.2.2 Justificación Práctica	23
1.3 Objetivos	25

1.3.1 Objetivo General	25
1.3.2 Objetivos Específicos	25
1.4 Hipótesis.....	25
1.5 Métodos y Técnica	26
1.5.1 Métodos	26
1.5.2 Técnicas y Fuentes de Información	26
 CAPÍTULO II	
MARCO TEÓRICO	28
2.1 Introducción	28
2.2 Servicios Web	28
2.2.1 Introducción.....	28
2.2.2 Definición	29
2.2.3 Estándares de los Servicios Web.....	30
2.2.4. Ventajas y Desventajas.....	33
2.2.4. Seguridad de los Servicios Web.....	35
2.3 Windows Communication Foundation (WCF)	36
2.3.1 Definición	36
2.3.2 Características de WCF	37
2.3.3. Diseño e Implementación de Servicios Web.....	38

2.3.4 Configuración Simplificada de un Servicio Web	41
2.3.5 Publicación de Servicio WCF	42
2.3.6 Interoperabilidad con otras Plataformas de los Servicios Web	43
2.4 AXIS2	44
2.4.1 Definición	44
2.4.2 Arquitectura	45
2.4.3 Manejo de Mensajes SOAP	47
2.4.4 Creación de Clientes	47
2.5 METRO	52
2.5.1 Definición	52
2.5.2 Web Service Interoperability Technologies (WSIT)	52
2.5.3 Java Architecture XML Binding (JAXB)	53
2.5.4 Relación Metro y .NET WCF	54
2.5.5 Usando Metro	55
2.5.6 Seguridad de Clientes mediante de SSL	58
CAPÍTULO III	
ANÁLISIS COMPARATIVO DE METRO Y AXIS2 PARA EL DESARROLLO DE APLICACIONES QUE CONSUMAN SERVICIOS WEB WCF	59
3.1 Introducción	59

3.2 Definición de los Parámetros de Comparación	60
3.3 Método para la Evaluación de Resultados	61
3.4 Instrumentos de Medición.....	63
3.2. Descripción del Escenario de Pruebas	64
3.2.1 Equipo Utilizado.....	65
3.2.2 Software Utilizado.....	65
3.2.2 Prototipos.....	67
3.3. Desarrollo de las pruebas con los parámetros de comparación.....	68
3.3.1. Rendimiento	70
3.3.2 Requerimientos de Hardware	85
3.3.3 Simplicidad.....	99
3.4 Demostración de la Hipótesis.....	107
3.5 Comprobación de la Hipótesis	110
 CAPÍTULO IV	
SITIO WEB DE INFORMACIÓN PARA ESTUDIANTES DE LA ESPOCH.	111
4.1 Introducción	111
4.2 Fase I. Planeación.....	112
4.2.1 Descripción del sistema.....	112
4.2.2 Definición del flujo del proceso de visualización de información estudiantil ...	113

4.2.2 Especificación de Requerimientos	115
4.2.3 Definición de usuarios	117
4.2.4 Definición de roles	117
4.2.5 Historias de Usuario	117
4.2.6 Plan de iteraciones	119
4.3 Fase II. Diseño de Software	123
4.3.1 Diseño de la Base de Datos	123
4.3.3 Diccionario de datos	126
4.3.4 Diseño de arquitectura	127
4.3.5 Diseño de Interfaces	133
4.4 Fase III. Codificación.....	136
4.4.1 BackEnd	136
4.4.2 FrontEnd	138
4.5 Fase IV. Pruebas.....	143

CONCLUSIONES

RECOMENDACIONES

RESUMEN

SUMMARY

GLOSARIO DE TÉRMINOS

ANEXOS

BIBLIOGRAFÍA

ÍNDICE DE FIGURAS

Figura II. 1 Interacción de estándares de servicios web	33
Figura II. 2 Manejo de Mensajes SOAP	47
Figura II. 3 Resultado en Consola de la generación con Axis2	51
Figura II. 4 Características de los servicios web WSIT de Metro	53
Figura III. 1 Resultados Parámetro Rendimiento	84
Figura III. 2 Porcentajes cumplimiento Rendimiento.....	84
Figura III. 3 Resultados del parámetro requerimientos de hardware.....	97
Figura III. 4 Porcentajes cumplimiento requerimientos de hardware.....	98
Figura III. 5 Resultados del parámetro simplicidad.....	106
Figura III. 6 Porcentajes cumplimiento parámetro simplicidad	106
Figura III. 7Resumen porcentajes parámetros	108
Figura III. 8 Resumen porcentajes finales de los frameworks.....	109
Figura IV. 1 Fases de metodología XP	112
Figura IV. 2 Proceso de autenticación	114
Figura IV. 3 Roles para el sitio de información para estudiantes	117
Figura IV. 5 Diagrama de base de datos OASIS-Sitio Estudiantil	124
Figura IV. 6 Diagrama la de base de datos Central-Sitio Estudiantil	125
Figura IV. 7 Diagrama de componentes del Sitio de Información para Estudiantes.....	130
Figura IV. 8 Diagrama de despliegue del sitio de información para estudiantes	132
Figura IV. 10 Vista del sitio web: Horio por paralelo	134
Figura IV. 11 Reporte horario de clase del estudiante.....	135

ÍNDICE DE TABLAS

Tabla III. I Parámetros e indicadores de comparación	60
Tabla III. II Niveles de Cumplimiento.....	62
Tabla III. III Especificaciones de hardware del equipo utilizado	65
Tabla III. IV Especificaciones de software utilizado.....	65
Tabla III. V Indicadores del parámetro rendimiento	70
Tabla III. VI Resultados de las pruebas para el throughput.....	72
Tabla III. VII Resumen de los resultados de la medición del throughput	76
Tabla III. VIII Calificación del indicador throughput	76
Tabla III. IX Resultados de las pruebas para el tiempo de respuesta	77
Tabla III. X Resultados de la medición del tiempo de respuesta.....	81
Tabla III. XI Calificación del indicador tiempo de respuesta.....	81
Tabla III. XII Indicadores del parámetro requerimientos de hardware	85
Tabla III. XIII Resultados de las pruebas para el uso de memoria RAM.....	87
Tabla III. XIV Resumen de los resultados de la medición del uso de memoria RAM.....	90
Tabla III. XV Calificación del indicador uso de memoria RAM	90
Tabla III. XVI Resultados de las pruebas para el uso de procesador	91
Tabla III. XVII Resultados de la medición de uso del procesador	94
Tabla III. XVIII Calificación del indicador uso del procesador	95
Tabla III. XIX Indicadores del parámetro simplicidad.....	99
Tabla III. XX Valoración para el indicador disponibilidad de información.....	101
Tabla III. XXI Fuentes de disponibilidad de información.....	101

Tabla III. XXII Calificación del indicador disponibilidad de información	101
Tabla III. XXIII Resultados de la medición para el indicador esfuerzo de desarrollo	102
Tabla III. XXIV Valoración para la fuente líneas de código	103
Tabla III. XXV Valoración para la fuente archivos generados	103
Tabla III. XXVI Resultados de las fuentes de disponibilidad	103
Tabla III. XXVII Valoración para el indicador esfuerzo de desarrollo	104
Tabla III. XXVIII Resumen calificación de indicador en cada parámetro	107
Tabla III. XXIX Resumen de calificaciones de los frameworks en cada parámetro	109
Tabla IV. I Historia de Usuario 1	118
Tabla IV. II Iteración 1	119
Tabla IV. III Iteración 2	120
Tabla IV. IV Iteración 3	120
Tabla IV. V Iteración 4	121
Tabla IV. VI Iteración 5	121
Tabla IV. VII Iteración 6	122
Tabla IV. VIII Iteración 7	122
Tabla IV. IX Descripción tabla Periodos Base de datos OASIS -Sitio Estudiantil	126
Tabla IV. X Prueba de la historia de usuario 1	145

INTRODUCCIÓN

En la actualidad el uso de los servicios web se ha vuelto popular debido a que estos permiten la interacción de aplicaciones desarrollados en plataformas heterogéneas, haciendo que estas sean interoperables, escalables, fáciles de mantener e integrar con sistemas y fuentes de datos existentes. Dado estas condiciones se han desarrollado frameworks de servicios web que permiten implementar aplicaciones de este tipo.

Mediante los frameworks de servicios web se pueden desarrollar tanto el servicio como el cliente, es decir el emisor y el receptor, que son fundamentales en toda aplicación interoperable. Dentro del desarrollo de este tipo de aplicaciones el cliente del servicio web juega un papel vital debido a que este es el encargado de entender los contratos expuestos por los servicios web y con ello lograr una comunicación directa entre aplicaciones desarrolladas en plataformas heterogéneas.

La presente investigación pretende realizar un análisis comparativo entre los frameworks de servicios web Metro y Axis2 orientados desde lado cliente, para el desarrollo de aplicaciones que consumen servicios web WCF; contemplando el posterior desarrollo de un

sitio de información estudiantil para la Escuela de Superior Politécnica de Chimborazo (ESPOCH) con el framework que brinde mejores prestaciones.

Este proyecto de investigación está estructurado en cuatro capítulos.

En el **Capítulo I**, se tratará sobre el marco referencial, en el cual se encuentra descrita de manera general los antecedentes, la justificación del proyecto de tesis, los objetivos a alcanzar y la hipótesis a demostrar con el desarrollo de la misma.

En el **Capítulo II**, se detallarán las definiciones conceptuales de los servicios web, interoperabilidad entre aplicaciones informáticas, framework de servicios web WCF, Metro y Axis2, estos dos últimos enfocados desde el cliente del servicio web.

En el **Capítulo III**, se enfoca en la realización del análisis comparativo entre los Frameworks de servicios web Metro y Axis2 enfocados en el cliente, teniendo como objetivo demostrar las fortalezas y debilidades mediante los parámetros de comparación establecidos en este capítulo.

En el **Capítulo IV**, con el framework que se seleccione en el Capítulo III, es decir, el que brinde mayores beneficios, se procederá al desarrollo del sitio de información estudiantil para la Escuela Superior Politécnica de Chimborazo; este capítulo se constituye por la documentación del sistema aplicado la metodología Programación Extrema (XP) logrando de esta forma relacionar lo investigativo con lo práctico.

El presente trabajo finaliza emitiendo las conclusiones y recomendaciones que son resultado del trabajo realizado.

El presente trabajo, servirá de soporte a la hora de seleccionar entre los dos framework de servicios web mencionados anteriormente y decidir cuál es el más adecuado para el desarrollo de aplicaciones que consuman servicios web WCF

CAPÍTULO I

MARCO REFERENCIAL

1.1 Antecedentes

La dinámica actual de las empresas origina su expansión, fusiones con otras empresas y aperturas a nuevos mercados por distintos medios, se plantea exigencias muy altas para las aplicaciones empresariales que existen detrás para soportar tal estructura. Algunas de las exigencias planteadas son: Escalabilidad y portabilidad, facilidad de expansión y mantenimiento, facilidad de integración con aplicaciones y fuentes de datos existentes. Por estas exigencias, las diferentes ofertas de plataformas de desarrollo que existen en el mercado y la gran diversidad de funcionalidades de cada uno de estos productos ofrecen, lleva a que cada vez sea más necesario combinar distintas plataformas para obtener una aplicación que se adapte lo mejor posible a las necesidades empresariales. Para el desarrollo

de este tipo de aplicaciones existen diferentes plataformas, entre los que se destacan Microsoft .NET, JAVA, PHP, entre otros.

Microsoft .NET y JAVA son dos de las plataformas más utilizadas para el desarrollo de aplicaciones empresariales web y han incidido en el crecimiento tecnológico y la comunicación de las empresas en el mundo.

En Ecuador debido al decreto 1014, que estipula que las empresas de carácter público deben usar software libre, surge una preocupación sobre los sistemas ya construidos en Microsoft .NET y su comunicación e interacción con las aplicaciones que se desarrollan en JAVA. Para lograr solventar este tipo de problemas ambas tecnologías han desarrollado un gran número de componentes y librerías, que permiten tener una interoperabilidad entre plataformas heterogéneas. Una de las principales dificultades que encuentran las empresas es poder desarrollar aplicaciones cliente JAVA que consuman servicios web WCF de manera rápida, fácil y que optimice los tiempos de respuesta para los usuarios finales.

Para Microsoft .NET el framework que proporciona funcionalidades de comunicación web más estable según “Microsoft” es WCF (Windows Communication Foundation) que provee un enfoque manejable a la informática distribuida, interoperabilidad ancha y asistencia directa para la orientación sobre servicios. Mientras que para JAVA se han desarrollado frameworks capaces de proporcionar un entorno de ejecución para servicios web; tales como Axis 1.x, Axis 2.x, Celtix, Glue, JBossWS, Xfire 1.2 y Metro (JAX-WS RI + WSIT). De los cuales según el Proyecto de Entorno de desarrollo de software Colaborativo “CODEHAUSE”, Axis2 y Metro son las que proveen más funcionalidades. Debido a las

funcionalidades que proveen Metro y Axis2; Los desarrolladores de aplicaciones web interoperables y distribuidas, se ven indecisos al momento de seleccionar el framework más adecuado para el desarrollo y desempeño de aplicaciones que consuman servicios web WCF. Razón por la cual se pretende hacer es un análisis comparativo de los frameworks de servicios web JAVA; AXIS2 y METRO.

En la actualidad el Dirección de Tecnologías de la Información y Comunicación DTIC en la Escuela Superior Politécnica de Chimborazo, administra sistemas informáticos tales como OSAIS E-virtual, entre otros, los mismos que están Orientados a brindar servicio a toda la Institución. Una de las principales dificultades que se presenta es el poder desarrollar aplicaciones web JAVA que interactúen con aplicaciones desarrolladas en .NET de forma rápida, fácil y que optimice los tiempos de respuesta para los usuarios finales; y con ello el mejorar el desempeño de aplicaciones web interoperables.

Para dar solución a uno de los inconvenientes presentados anteriormente se implantara un sitio de información académica para estudiantes de la ESPOCH utilizando el framework de servicios web JAVA más adecuado, resultado de la comparación hecha en la presente investigación. El sitio web consumirá los servicios web WCF, se vinculara además con la base de datos centralizada con el propósito de mejorar las prestaciones del sistema actual e implementar los beneficios de la interoperabilidad.

1.2 Justificación

1.2.1 Justificación Teórica

Los frameworks de servicios web JAVA proporcionan un entorno de ejecución para servicios web, lo cual permite una comunicación exitosa y rápida con aplicaciones desarrolladas en plataformas heterogéneas. El principal objetivo de estos frameworks es desarrollar aplicaciones web interoperables de manera rápida, fácil, y con la máxima reutilización de código. Los frameworks estables en el mercado son Axis 1.x, Axis2 1.2, Celtix, Glue, JBossWS, Xfire 1.2 y Metro 2.2 (JAX-WS RI + WSIT). De los cuales según el Proyecto de Entorno de desarrollo de software Colaborativo “CODEHAUSE”, Axis2 y Metro son las que proveen más funcionalidades.

El presente análisis pretende estudiar las tecnologías JAVA de consumo de servicios web, METRO y AXIS2 para el desarrollo de aplicaciones que consuman servicios web WCF, enfocándonos principalmente en el incremento del desempeño de las aplicaciones web interoperables. Ambas tecnologías son similares y presentan un gran crecimiento y acogida entre los desarrolladores de aplicaciones web, por lo cual se considera relevante el poder determinar por medio de esta investigación el comportamiento de estas tecnologías que permitan describir con claridad la mejor opción para el incremento del desempeño de una aplicación web interoperable.

1.2.2 Justificación Práctica

Con el propósito de determinar la tecnología más adecuada, se construirán prototipos de desarrollo, los cuales permitan evaluar mediante parámetros de desempeño las aplicaciones

web interoperables. Los prototipos se constituirán de pequeñas aplicaciones web cliente, las mismas que serán desarrolladas JAVA con los frameworks de servicios web METRO y AXIS 2.

DTIC presenta en la actualidad una dificultad al poder desarrollar aplicaciones web JAVA que interactúen con aplicaciones desarrolladas en .NET de forma rápida, fácil, óptima y con ello el mejorar el rendimiento de aplicaciones web interoperables. Debido a esta situación es necesario implementar un nuevo sitio de información académica para estudiantes de la ESPOCH utilizando el framework de servicios web JAVA más adecuado, resultado de la comparación hecha en la presente investigación.

El sitio web consumirá los servicios web WCF construidos en base al sistema académico OASIS y se vinculara además con la base de datos centralizada, con el propósito de mejorar las prestaciones del sistema actual. Y con esto implementar las los beneficios de la interoperabilidad.

Entre los principales beneficios del sitio web propuesto, será distribuido, desarrollado en tecnología libre y propietaria, compartirá datos con otros sistemas. Además estará compuesto de los siguientes módulos: Perfil, Horarios del Estudiante, Horarios Materia, Notas Académicas, Matrícula y Centro de Noticias Académicas.

1.3 Objetivos

1.3.1 Objetivo General

Realizar un análisis comparativo de metro y axis2 para el desarrollo de aplicaciones que consuman servicios web WCF en la ESPOCH

1.3.2 Objetivos Específicos

- Estudiar las características de los servicios web WCF y frameworks METRO y AXIS2.
- Establecer parámetros de comparación los cuales permitan determinar con claridad cual framework mejora el desempeño de las aplicaciones que consuman servicios web WCF.
- Seleccionar el framework más adecuado para el incremento del desempeño de aplicaciones que consuman servicios web WCF, en base a los parámetros previamente establecidos.
- Desarrollar los servicios web WCF de información académica estudiantil.
- Realizar un sitio web de información académica para estudiantes en tecnología JAVA que consuma los servicios web WCF y que interactúe con una base de datos centralizada de la ESPOCH y a su vez se vincule con el sistema académico (OASIS).

1.4 Hipótesis

La aplicación de la tecnología Metro permite mejorar el desempeño de las aplicaciones que consumen servicios web WCF, en relación a la tecnología AXIS2.

1.5 Métodos y Técnica

1.5.1 Métodos

El método utilizado como guía para la realización de la presente investigación es el método científico, el cual contempla los siguientes puntos:

- El planteamiento del problema que en este caso es el análisis comparativo de METRO y AXIS2 en el desarrollo de aplicaciones que consuman servicios web WCF.
- El apoyo del proceso previo a la formulación de la Hipótesis.
- Levantamiento de información necesaria.
- Análisis e interpretación de Resultados.
- Proceso de comprobación de la Hipótesis.
- Para el desarrollo del sitio web se aplicará la metodología de desarrollo SCRUM

1.5.2 Técnicas y Fuentes de Información

Técnicas

Las técnicas que se usarán son:

- Pruebas
- Observación

Fuentes de Información

Para lo que tiene que ver en cuanto a fuentes de información se utilizarán libros, Sitos web Oficiales de las tecnologías, etc.

CAPÍTULO II

MARCO TEÓRICO

2.1 Introducción

En el presente capítulo se detallarán las definiciones conceptuales de servicios web incluyendo sus ventajas, desventajas, seguridad y despliegue en los servidores web. Además se dará a conocer los frameworks de servicios web utilizados en la presente investigación

2.2 Servicios Web

2.2.1 Generalidades

En la actualidad la globalidad de la información hace que la forma de comunicación entre las personas y los negocios que realizan estas cambien rotundamente, haciendo que las aplicaciones web tradicionales no sean suficientes para esta demanda, bajo este contexto

surge la necesidad de desarrollar aplicaciones que se integren a otras independientemente de su plataforma de desarrollo. Para lograr esto, las empresas de desarrollo de software más grandes en el mundo, tales como Microsoft, IBM, Oracle han desarrollado un lenguaje común que permita el intercambio de información basándose en estándares existentes, surgiendo así los servicios web

2.2.2 Definición

Existen varias definiciones:

W3C.- Un servicio Web es un sistema de software diseñado para apoyar la interoperabilidad de máquina a máquina sobre una red de interacción. Tiene una interfaz descrita en un formato procesable por máquina (específicamente WSDL). Otros sistemas interactúan con el servicio Web en la forma prescrita por su descripción utilizando mensajes SOAP, típicamente transportados usando HTTP con una serialización XML en conjunción con otros estándares relacionados con la web[5].

Sun Microsystems.- Un servicio Web describe la funcionalidad específica del negocio expuesta por una compañía, generalmente a través de una conexión de Internet, con el fin de proporcionar una manera para que otra compañía, o programa informático utilice el servicio [16].

Microsoft.- Los servicios Web son componentes de un servidor web a los que puede llamar una aplicación cliente realizando solicitudes HTTP a través de Internet [13].

De lo expuesto anteriormente, un servicio web es un componente de software que expone funcionalidades de negocio a través de una red, para que sean consumidas por una aplicación cliente mediante peticiones HTTP.

2.1.3 Estándares de los Servicios Web

Los servicios web para lograr un funcionamiento y una comunicación se basan en los siguientes estándares web:

XML: eXtensible Markup Language

El Extensible Markup Language (XML) es un simple formato basado en texto para representar información estructurada: documentos, datos, configuración, libros, transacciones, facturas y mucho más [18]. Físicamente, el documento está compuesto de unidades llamadas entidades, todo documento tiene una entidad raíz. En servicios web XML, todos los datos que se intercambiarán están formateados con etiquetas XML

SOAP: Simple Object Access Protocol

SOAP es un protocolo liviano, basado en XML, para el intercambio de información estructurada en un ambiente descentralizado y distribuido. Se encuentra en el núcleo de los servicios web proporcionando un mecanismo estándar de empaquetar mensajes.

Algunas de las mayores Compañías que soportan SOAP son Microsoft, IBM y Oracle.

WSDL: Web Services Definition Language

A medida que los protocolos de comunicación y formatos de mensaje se estandarizados en la comunidad web, se hace cada vez más posible e importante ser capaz de describir las comunicaciones de una manera estructurada. WSDL aborda esta necesidad definiendo una

gramática XML para describir servicios de red como colecciones de extremos de comunicación capaces de intercambiar mensajes. Las definiciones de servicio WSDL proporcionan documentación para sistemas distribuidos y sirven como una receta para la automatización de los detalles involucrados en las aplicaciones de comunicación [19].

Es importante observar que WSDL no introduce un lenguaje de definición de tipo nuevo. WSDL reconoce la necesidad de sistemas de tipos ricos para describir formatos de mensaje, y es compatible con la especificación de esquemas XML (XSD) como su sistema de tipo canónico. Además, WSDL define un mecanismo de enlace común. Esto se utiliza para enlazar un protocolo específico, o formato de datos o la estructura de un mensaje abstracto, operación, del extremo. Se permite la reutilización de definiciones abstractas [19].

Estructura del Documento WSDL.- Un documento WSDL es simplemente un conjunto de definiciones. Hay un elemento de definiciones en la raíz, y definiciones internas. La Gramática es la siguiente [19]:

```
<definitions>
  <types>los_tipos_de_datos...
</types>
  <message> las_definiciones_del_mensaje...
</message>
  <portType> las_definiciones_de_operación ...
</portType>
  <binding>las_definiciones_de_protocolo...
</binding>
  <service>direcciones_relacionadas ...
</service>
</definitions>
```

UDDI: Universal Discovery, Description and Integration

Es una especificación industrial para publicar y localizar información acerca de los servicios Web. Se puede utilizar los servicios UDDI para publicar, descubrir, compartir e interactuar con los servicios Web dentro de la empresa o entre los socios comerciales. Con los servicios UDDI, los desarrolladores pueden interactuar con los servicios web directamente a través de sus herramientas de desarrollo y aplicaciones de negocio [14].

UDDI adopta un enfoque que se basa en un registro distribuido de las empresas y su servicio descripciones implementado en un formato XML común.

Un proveedor de servicios aloja un servicio Web y lo hace accesible con protocolos como SOAP/HTTP y SOAP/JMS. El servicio Web se describe mediante los documentos WSDL que se almacenan en el servidor del proveedor o en un depósito especial. Los servicios de empresa UDDI hacen referencia a los documentos WSDL (documentos de servicio) y tModels (documentos de enlace). Estos punteros permiten que los solicitantes de servicio encuentren servicios Web [7]

Interacción de estándares de la web

Las empresas que desean exponer funcionalidades para que estas sean consumidas por clientes, realizan en primera instancia el registro mediante el WSDL en un repositorio UUID, una vez ahí, el consumidor realizara una búsqueda en el repositorio para obtener WSDL publicado, y con esto realiza una petición SOAP (XML) vía HTTP al servicio web publicado esperando una respuesta del mismo tipo. En la Figura II.1 se puede observar de forma gráfica la interacción de los estándares.

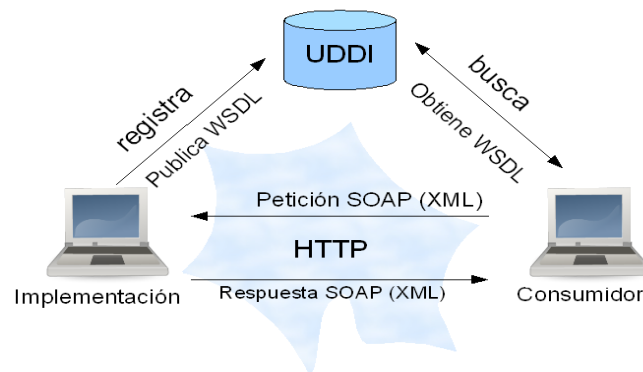


Figura II. 1 Interacción de estándares de servicios web

Fuente: Labra, José Emilio [9]

2.2.4. Ventajas y Desventajas

Ventajas

Los servicios web ofrecen muchos beneficios frente a otras arquitecturas de software distribuidas, entre las principales ventajas se puede destacar.

Interoperabilidad. Este es el beneficio más importante de los Servicios Web. Estos suelen trabajar fuera de las redes privadas, ofreciendo a los desarrolladores una ruta no propietario a sus soluciones. Los servicios desarrollados son probables, por lo tanto, pueden tener una vida más larga que ofrecer. Además permiten que los desarrolladores usen sus lenguajes de programación preferidos. Además, gracias a la utilización de métodos de comunicaciones basadas en estándares, los servicios web son prácticamente independientes de la plataforma [13].

Independencia.- Totalmente independientes de la plataforma, no hay restricciones en cuanto a la plataforma en la que pueden ser desarrollados, las aplicaciones que utilizan los servicios web pueden ejecutarse en cualquier plataforma [8].

Basados en estándares de la web.- Los servicios web pueden ser desarrollados como componentes de aplicación débilmente acoplados utilizando cualquier lenguaje de programación, cualquier protocolo o plataforma [8].

Integración con sistemas existentes.- Permiten que servicios y software de diferentes compañías ubicadas en diferentes lugares geográficos puedan ser combinados fácilmente para proveer servicios integrados.

Desventajas.

Reducción de Velocidad de Transmisión.- Los servicios Web utilizan protocolos de texto plano que utilizan un método bastante detallado para identificar los datos. Esto significa que las solicitudes de servicio Web son más grandes que las solicitudes codificadas con un protocolo binario. El tamaño extra es realmente sólo un problema a través de conexiones de baja velocidad o en conexiones extremadamente ocupados [8].

Implementación de Seguridad.- La creación de servicios web seguros, es una programación avanzada, es decir que se requiere de un gran conocimiento previo para poder configurarlos e implementarlos en un ambiente de producción.

2.2.4. Seguridad de los Servicios Web

La especificación SOAP básica no establece la protección de mensajes, y deja esas especificaciones extendidas. El problema yace en la naturaleza de una aplicación de servicios web. En la mayoría de los casos, se trata con SOAP sobre HTTP, lo que significa que cada mensaje debe pasar por uno o más nodos intermedios, cualquiera de los cuales puede leer y/o alterar un mensaje. Y eso suponiendo que la solicitud SOAP misma fuera directa. En algunos casos, los mensajes SOAP están diseñados específicamente para viajar a través de más de un nodo antes de llegar a su destino final. El resultado final es que se necesita evitar que alguien que no sea el destinatario deseado lea información confidencial, a fin de evitar la interceptación. Además es necesaria una manera de evitar que alguien que no sea el remitente deseado envíe mensajes, a fin de evitar el acceso no autorizado [15].

Para lograr una seguridad integrada, que implique una autenticación, autorización y encriptación de los servicios web, se han implementado diferentes métodos tales como: especificaciones de seguridad SOAP WS-Security y servicios web sobre SSL, este último método, será utilizado en el desarrollo de los servicios web WCF para la presente investigación.

La especificación SOAP.- Hay tres grandes problemas en la protección de intercambios de mensajes SOAP, y WS-Security brinda respuestas para todos ellos, aunque no directamente. Es, de hecho, una especificación que habla no sobre cómo proteger el mensaje, sino cómo hacer saber al destinatario que se ha protegido el mensaje. Para realizar la protección real, WS-Security referencia especificaciones adicionales [15].

Servicios web sobre SSL.- La seguridad de servicios web sobre SSL (Security Sockets Layer) se concentra en la seguridad a nivel de transporte, estableciendo un canal seguro, por el cual se transmiten los datos, es decir que la información se transmite mediante HTTPS. Para lograr este nivel de seguridad se debe en primera instancia registrar el certificado SSL en un servidor web, agregar el enlace SSL al sitio web, configurar el servicio web para que haga uso de HTTPS y dentro de este último se establece el método de autenticación, que permite un nivel superior de seguridad.

Cabe aclarar que a diferencia de la seguridad SOA WS-Security, SSL se configura en el servidor y cliente, sin afectar de manera exclusiva a los mismos. Para el desarrollo del aplicativo de la presente investigación se hará uso de este tipo de seguridad, debido a que provee un nivel más óptimo de rendimiento y seguridad.

Mediante la seguridad en los servicios web, se pretende garantizar una entrega y recepción de información fiable, confiable, autorizada y autenticada.

2.3 Windows Communication Foundation (WCF)

2.3.1 Definición

Windows Communication Foundation (WCF) es un framework para la creación de aplicaciones orientadas a servicios. Con WCF, es posible enviar datos como mensajes asincrónicos de un extremo de servicio a otro [13].

Un extremo de servicio puede formar parte de un servicio disponible continuamente hospedado por IIS, o puede ser un servicio hospedado en una aplicación. Un extremo puede

ser un cliente de un servicio que solicita datos de un extremo de servicio. Los mensajes pueden ser tan simples como un carácter o una palabra que se envía como XML, o tan complejos como una secuencia de datos binarios [13].

2.3.2 Características de WCF

WCF incluye el siguiente conjunto de características:

Orientación a servicios.- Como consecuencia del uso de los estándares de WS, WCF le permite crear aplicaciones orientadas a servicios [13].

Interoperabilidad.- WCF implementa los estándares del sector modernos para la interoperabilidad de servicios web [13].

Varios modelos de mensajes.- Los mensajes se intercambian mediante uno de los distintos modelos. El más común es el de solicitud/respuesta, en que un extremo solicita datos de otro extremo. y el otro extremo responde [13].

Metadatos de servicios.- WCF admite la publicación de metadatos de servicios utilizando los formatos especificados en los estándares de la industria, como WSDL, Esquemas XML y WS-Policy. Los metadatos se pueden publicar sobre HTTP y HTTPS, o utilizando el estándar Intercambio de metadatos de servicios web [13].

Contratos de datos.- Dado que WCF se basa en .NET Framework, también incluye métodos con código sencillo para proporcionar los contratos que desea aplicar. WCF incluye un completo sistema para trabajar con datos de esta manera fácil. Cuando se han creado las clases que representan los datos, el servicio genera automáticamente los

metadatos que permiten a los clientes ajustarse a los tipos de datos que se han diseñado [13].

Seguridad.- Es posible cifrar los mensajes para proteger la privacidad, así como obligar a los usuarios a que se autentiquen antes de permitirles recibir mensajes. La seguridad puede implementarse utilizando estándares conocidos como SSL o WS-SecureConversation [13].

Transacciones.- WCF también admite las transacciones que usan uno de los tres modelos de transacción: las transacciones WS-Atomic, las API del espacio de nombres System.Transactions y Coordinador de transacciones distribuidas de Microsoft [13].

Extensibilidad.- La arquitectura de WCF tiene varios puntos de extensibilidad. Si se necesita una función adicional, existen una serie de puntos de entrada que le permiten personalizar el comportamiento de un servicio [13].

2.3.3. Diseño e Implementación de Servicios Web

Los servicios web WCF se fundamentan principalmente en la implementación de contratos. Un contrato de servicio especifica lo que un extremo comunica al mundo exterior. En un nivel más concreto, es una instrucción sobre un conjunto de mensajes concretos organizada en modelos de intercambio de mensajes básicos, como solicitud/respuesta, unidireccional y dúplex. Si un contrato de servicio es un conjunto relacionado de forma lógica de intercambios de mensajes, una operación de servicio es un intercambio único de mensajes [13].

Contratos de servicio

Un contrato de servicio especifica lo siguiente:

- Operaciones que el contrato expone [13].
- Firma de las operaciones en términos de los mensajes intercambiados [13].
- Tipos de datos de estos mensajes [13].
- Ubicación de las operaciones [13].
- Protocolos concretos y formatos de serialización que se utilizan para admitir la comunicación correcta con el servicio [13].

Para transmitir este tipo de información a otras aplicaciones en varias plataformas (incluidas plataformas que no sean de Microsoft), los contratos de servicio XML se expresan públicamente en formatos XML estándar, como Web Services Description Language (WSDL) y XML Schema (XSD) [13].

Implementación de contratos de servicio

Un servicio es una clase que expone la funcionalidad disponible a los clientes en uno o más extremos. Para crear un servicio, se escribe una clase que implemente un contrato Windows Communication Foundation (WCF). Hay dos maneras de hacerlo. Se puede definir el contrato separadamente como una interfaz y, a continuación, crear una clase que implemente esa interfaz. También puede crear la clase y el contrato directamente colocando el atributo `ServiceContractAttribute` en la misma clase y el atributo `OperationContractAttribute` en los métodos disponibles para los clientes del servicio [13].

Crear una clase de servicio:

A continuación, se muestra un ejemplo de un servicio que implementa un contrato **IMath** que se ha definido separadamente [13].

```
[ServiceContract]
public interface IMath{
    [OperationContract]
    double Add(double A, double B);
    [OperationContract]
    double Multiply (double A, double B);
}
public class MathService : IMath{
    public double Add (double A, double B) { return A + B; }
    public double Multiply (double A, double B) { return A * B; }
}
```

De forma alternativa, un servicio puede exponer contrato directamente. A continuación, se muestra un ejemplo de una clase de servicio que define e implementa un contrato **MathService** [13].

```
[ServiceContract]class MathService{
    [OperationContract] public double Add(double A, double B) { return A + B; }
    [OperationContract]
    private double Multiply (double A, double B) { return A * B; }
}
```


2.3.4 Configuración Simplificada de un Servicio Web

Todas las configuraciones que determinan el comportamiento del servicio web se lo hace en un archivo llamado Web.config.

Dentro de Web.config la sección `<system.serviceModel>` contiene el elemento `<Services>`, en el cual se configura los servicios web, los mismos que a su vez necesitan de una configuración previa de `<behaviors>`, configuración de enlace `<Bindings>` sobre un `<endpoint>` [13].

Los comportamientos del servicio se configuran para los extremos predeterminados mediante secciones `<behavior>` anónimas dentro de secciones `<serviceBehaviors>`. Todos los elementos `<behavior>` sin nombre dentro de `<serviceBehaviors>` se utilizan para configurar comportamientos del servicio [13].

Por ejemplo, el siguiente archivo de configuración se realiza una configuración típica para la exposición de un servicio web.

```
<system.serviceModel>
  <services>
    <service name="serviceName" behaviorConfiguration="">
      <endpoint address="" bindingConfiguration=" " binding=" "
contract="WCFInteropService.ISeguridad">
        </endpoint>
      </service>
    </services>
    <behaviors>
      <serviceBehaviors>
        <behavior>
          <serviceMetadata httpGetEnabled="True"/>
          <serviceDebug includeExceptionDetailInFaults="false"/>
        </behavior>
      </serviceBehaviors>
    </behaviors>
  </system.serviceModel>
```

```
</serviceBehaviors>
</behaviors> <bindings>
  <basicHttpBinding>
    <binding maxBufferSize="100"
      maxReceiveBufferSize="100" />
  </basicHttpBinding>
</bindings>
<protocolMapping>
  <add scheme="https" binding="basicHttpBinding" / </protocolMapping>
</system.serviceModel>>
```

2.3.5 Publicación de Servicio WCF

La publicación de servicios de WCF permite publicar servicios web WCF creados a partir del conjunto de plantillas de biblioteca de servicio de WCF y sus plantillas de elementos correspondientes [13].

El servicio se puede publicar en las ubicaciones de destino siguientes [13]:

- IIS local.
- Sistema de archivos.
- Sitio FTP.
- Sitio remoto.

Archivos generados para la publicación

Antes de que una biblioteca de servicio de WCF se pueda hospedar en Web, la herramienta genera los archivos siguientes: los archivos de ensamblado, el archivo Web.config y el archivo .svc. Todos los archivos se copian en la ubicación de destino [13].

2.3.6 Interoperabilidad con otras Plataformas de los Servicios Web

Las empresas hoy en día suelen tener los sistemas y aplicaciones que han adquirido a un amplio abanico de proveedores. En la aplicación de alquiler de coches, por ejemplo, se requiere establecer comunicación con algunas otras aplicaciones de software escritas en varios lenguajes que se ejecutan en varios sistemas operativos [13].

Debido a que el mecanismo de comunicación fundamental de WCF es un servicio Web basado en SOAP, las aplicaciones basadas en WCF pueden comunicarse con otro software que se ejecute en una variedad de contextos. Una aplicación generada en WCF puede interactuar con todo lo siguiente [13]:

- Las aplicaciones basadas en WCF que se ejecutan en un proceso diferente en el mismo equipo de Windows [13].
- Las aplicaciones basadas en WCF que se ejecutan en otro equipo de Windows [13].
- Las aplicaciones generadas en otras tecnologías, como servidores de aplicaciones de J2EE, que son compatibles con los servicios Web estándar. Estas aplicaciones se pueden estar ejecutando en equipos con Windows o en los equipos que ejecutan otros sistemas operativos [13].

Para permitir más que únicamente la comunicación básica, WCF implementa tecnologías de servicios Web definidas por las especificaciones de WS-*. Todas estas especificaciones fueron definidas originalmente por Microsoft, IBM y otros proveedores que trabajan juntos [13].

2.4 AXIS2

2.4.1 Definición

El proyecto Apache Axis2 es un framework basado en Java, tanto del cliente como del servidor de la ecuación de servicios Web. Diseñado para aprovechar las lecciones aprendidas de Apache Axis 1.0, Apache Axis2 proporciona un modelo de objetos completo y una arquitectura modular que hace fácil agregar funcionalidad y soporte para los nuevos servicios de la Web relacionados con las especificaciones y recomendaciones [3].

Axis2 permite realizar fácilmente las siguientes tareas [3]:

- Enviar mensajes SOAP.
- Recibir y procesar mensajes SOAP.
- Crear un servicio Web de una simple clase Java.
- Crear clases de implementación para el servidor y el cliente mediante WSDL.
- Fácilmente recuperar el WSDL para un servicio.
- Enviar y recibir mensajes SOAP con archivos adjuntos.
- Crear o utilizar un servicio web basado en REST.
- Crear o utilizar los servicios que se aprovechan de la WS-Security, WS-ReliableMessaging, WS-Addressing, WS-Coordinación y WS-Atomic Transaction recomendaciones.
- Utilizar la estructura modular Axis2 para añadir fácilmente soporte para las nuevas recomendaciones que van surgiendo.

2.4.2 Arquitectura

La arquitectura de Axis2 establece algunos principios para preservar la uniformidad. Estos son los siguientes [1].

La arquitectura Axis2 separa la lógica y los estados.- El código que realiza el procesamiento no tiene un estado dentro de Axis2. Esto permite que el código se ejecute libremente por hilos paralelos [1].

Toda la información se guarda en un modelo de información.- Lo que permite que el sistema se detenga y se reanude [1].

La arquitectura Axis2 es modular. Por lo tanto, el framework Axis2 está constituido por módulos básicos, que en conjunto conforman la arquitectura principal de Axis2. Los Módulos Non-core/other se acodan en la parte superior de estos módulos básicos [1].

El desempeño en términos de memoria y la velocidad es una consideración importante para Axis2. El núcleo se basa en tres especificaciones WSDL, SOAP y WS-Addressing . Otras especificaciones, como JAX -RPC, SAAJ y WS-Policy se colocan en capas en la parte superior de la arquitectura Core [1].

Módulos básicos:

Información del modelo.- Axis2 define un modelo para manejar la información y todos los estados se mantienen en este modelo. El modelo consiste en una jerarquía de la información. El sistema gestiona el ciclo de vida de los objetos en esta jerarquía [1].

El procesamiento del modelo XML.- Manipulación del mensaje SOAP es la tarea más importante y más compleja. La eficacia de este es el factor más importante que decide el rendimiento. Tiene sentido de delegar esta tarea a un sub - proyecto separado en el framework del proyecto de servicios Web, lo que permite que los sub - proyectos (axioma o eje Object Model) para proporcionar una API simple para SOAP y XML. Se esconde la complejidad de procesamiento XML eficiente dentro de su aplicación [1].

Modelo de procesamiento de SOAP.- Controla la ejecución del proceso. El modelo define las diferentes fases de la ejecución, y el usuario puede extender el modelo de procesamiento en lugares específicos [1].

Modelo de Despliegue - El modelo de implementación Axis2 permite al usuario desplegar servicios, configurar el transporte, y extender el modelo de procesamiento de SOAP por sistema, servicio o base funcionamiento [1].

API Client.- Esto proporciona una API conveniente para los usuarios comunicarse con servicios web usando Axis2. Hay un conjunto de clases para interactuar con IN -OUT y IN Sólo patrones de estilo de intercambio de mensajes (MEP) [1].

Transporte.- Axis2 define un marco de transporte que permite al usuario utilizar varios medios de transporte diferentes. Los transportes encajan en lugares específicos en el modelo de procesamiento de SOAP. La aplicación proporciona unos medios de transporte comunes y el usuario puede escribir o usar algún otro plug-in nuevo, siempre y cuando sea necesario [1].

2.4.3 Manejo de Mensajes SOAP

Axis2 puede manejar el procesamiento tanto para el emisor y el receptor en una transacción [3]. Desde la perspectiva de Axis2, la estructura se observa en la Figura II.2.

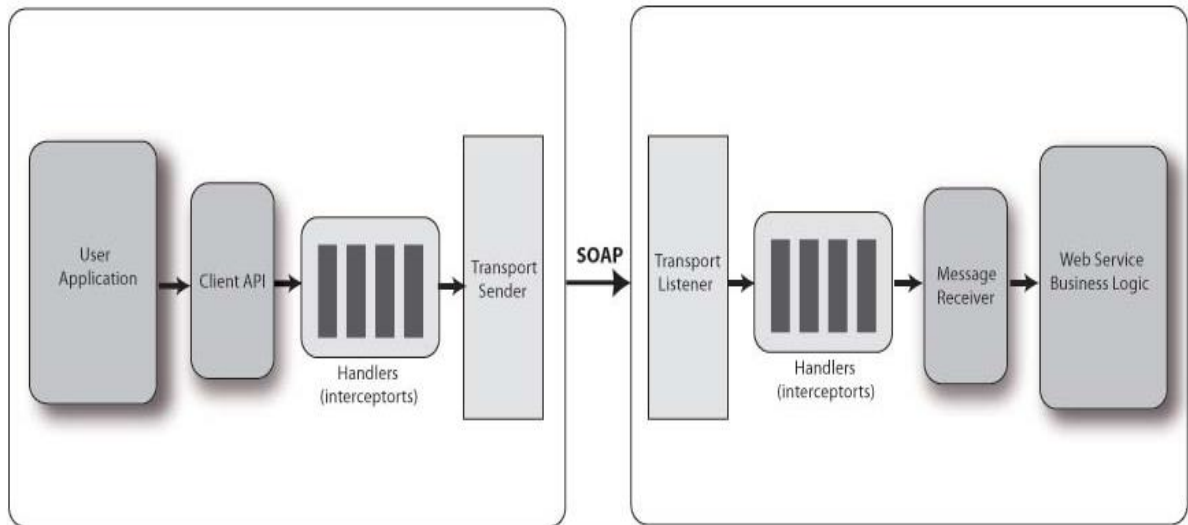


Figura II. 2 Manejo de Mensajes SOAP

Fuente: Apache Axis2 [3].

En cada extremo, usted tiene una aplicación diseñada para hacer frente a los mensajes (enviados o recibidos). En el medio, hay que Axis2, o más bien, puede hacer que Axis2. El valor de los servicios Web es que el remitente y el receptor (cada uno de los cuales puede ser o bien el servidor o el cliente) incluso no tienen que estar en la misma plataforma, y mucho menos ejecutando la misma aplicación [3].

2.4.4 Creación de Clientes

Cuando se trata de crear un cliente de servicios Web, se puede hacer de forma manual, pero en la mayoría de los casos, es mejor tener una definición de Web Service Description

Language (WSDL) que describe a los clientes que deben enviar mensajes y esperar recibir. Axis2 proporciona varias maneras de utilizar esta definición para generar automáticamente un cliente [2].

Selección del método de generación del cliente.

Axis2 provee varias opciones cuando se trata de mapeo WSDL a objetos para genera de clientes. Tres de estas opciones son los enlaces de datos Axis2 DataBinding, XMLBeans y JiBX. Todos estos métodos implican el uso de enlace de datos para crear objetos Java fuera de las estructuras XML que utiliza el servicio, y cada uno tiene sus pros y sus contras [2].

Framework Axis2 Databinding (ADB): ADB es probablemente el método más simple de generar un cliente de Axis2. En la mayoría de los casos, todas las clases pertinentes se crean como clases internas de una clase principal. ADB es muy fácil de usar, pero tiene limitaciones. No está destinado a ser una aplicación completa de enlace de esquema, y tiene dificultad con estructuras tales como extensiones y restricciones de los elementos del esquema XML [2].

XMLBeans: A diferencia de ADB, XMLBeans es un compilador de esquema completamente funcional, por lo que no tienen las mismas limitaciones que el ADB. Sin embargo, es un poco más complicado de usar que ADB. Se genera una enorme cantidad de archivos, y el modelo de programación que ciertamente es útil, no es tan sencillo como ADB [2].

JiBX: JiBX es un framework de enlace de datos que en realidad ofrece no sólo la conversión de WSDL a Java, sino también la conversión de Java a XML. En cierto modo,

JiBX ofrece lo mejor de ambos mundos. JiBX es extremadamente flexible, que le permite elegir las clases que representan las entidades, pero puede ser complicado de configurar. Por otro lado, una vez que se ha configurado, en realidad utilizando el código generado es tan fácil como usar ADB [2].

Al final, para estructuras simples, ADB probablemente será suficiente. Si, por el contrario se necesita más potencia o la flexibilidad, tanto si elige XMLBeans o JiBX depende de la cantidad de energía o la flexibilidad que necesita y la tolerancia de su complejidad [2].

Generación de clientes

El proceso para la generación y el uso de un cliente varía ligeramente en función del método de generación que se elija. Para la generación se hará uso del método de generación ABD bajo un sistema Windows [2].

Para crear un cliente usando el ADB, se debe ejecutar los siguientes pasos:

Descargar y descomprimir la distribución estándar de Apache Axis2. La Distribución Axis2 WAR no incluye las utilidades necesarias para la generación de código, como WSDL2Java [2].

En el método de ADB de generar clientes, todas las funcionalidades de los servicios se encuentran en una única clase llamada a stub. El stub contiene clases internas correspondientes a todos los objetos necesarios definidos en el archivo WSDL. Una vez que se tenga el stub, será capaz de crear un cliente simplemente refiriéndose a estas clases y sus métodos. Para generar el cliente, se debe ejecutar el siguiente comando [2]:

```
%AXIS2_HOME%\bin\WSDL2Java -uri Axis2UserGuide.wsdl -p  
org.apache.axis2.axis2userguide -d adb -s
```

Este comando analiza el archivo WSDL y crea el código auxiliar en el paquete *org.apache.axis2.axis2userguide*. Las opciones especifican [2]:

-uri.-Nombre del archivo WSDL.

-p.- Paquete auxiliar de generación de código Java.

-d abs.- Los métodos deben estar relacionados.

-s.- La comunicación se debe realizar en modo asincrónico.

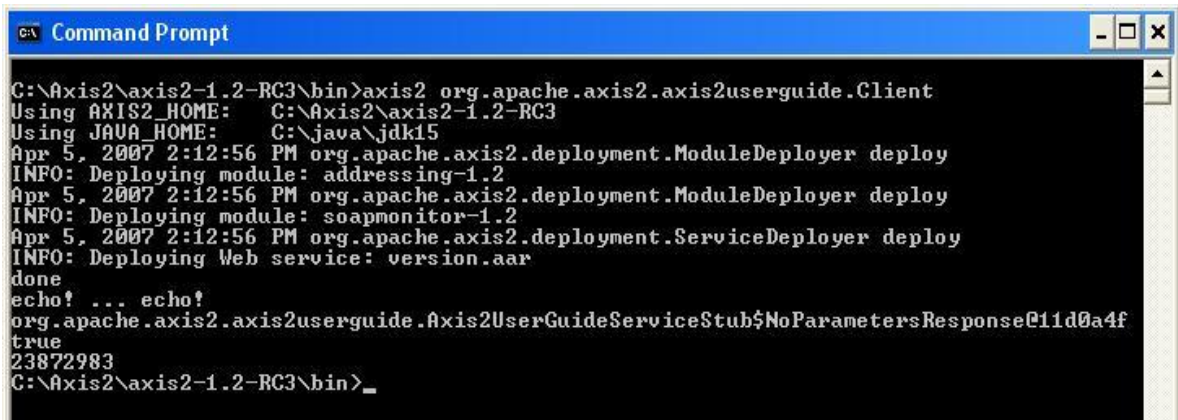
Una vez que se ejecuta este comando, se puede observar dos nuevos archivos en el directorio. El primero es el archivo *build.xml*, que contiene las instrucciones para Ant para compilar las clases generadas y el segundo es el directorio *src*, que contiene el archivo *Axis2UserServiceStub.java*. Al abrir el archivo, se podrá ver una colección de clases internas para cada uno de los elementos en el archivo WSDL. También se verá una serie de llamadas a la API del cliente Axis2, incluyendo aquellos que utilizan AXIOM para construir y analizar los mensajes entrantes y salientes. Ahora lo que se necesita es un cliente para hacer uso de este código. Para crear un cliente, se crea una nueva clase y se guarda como *Client.java* en el directorio *org/apache/axis2/axis2userguide* [2].

Se debe tener en cuenta que el uso del servicio es simplemente una cuestión de crear y llenar el tipo apropiado de solicitud utilizando los nombres definidos en el archivo WSDL, y luego usar el stub para enviar realmente la petición al método apropiado. Por ejemplo, para llamar a la operación *DoInOnly*, se crea un *DoInOnlyRequest*, utiliza su método

setMessageString () para establecer el contenido de su elemento messageString, y le pasa como argumento a stub.DoInOnly () [2].

Para construir el cliente, escribir “ant jar.client”: Esta acción crea dos nuevos directorios, src y test. El directorio de test estará vacío, pero el directorio de construcción contiene dos versiones del cliente. La primera versión, en el directorio lib, es un archivo .ja. que contiene la clase de cliente y el stub. El segundo, en el directorio de clases, se encuentra sólo clases primas. Asegúrese de que todos los archivos .jar en el directorio Axis2 lib se encuentran en la ruta de clases [2].

La respuesta en la ventana de consola luego de la ejecución satisfactoria debería ser semejante a la Figura No 2.3.2.



```
C:\Axis2\axis2-1.2-RC3\bin>axis2 org.apache.axis2.axis2userguide.Client
Using AXIS2_HOME: C:\Axis2\axis2-1.2-RC3
Using JAVA_HOME: C:\java\jdk15
Apr 5, 2007 2:12:56 PM org.apache.axis2.deployment.ModuleDeployer deploy
INFO: Deploying module: addressing-1.2
Apr 5, 2007 2:12:56 PM org.apache.axis2.deployment.ModuleDeployer deploy
INFO: Deploying module: soapmonitor-1.2
Apr 5, 2007 2:12:56 PM org.apache.axis2.deployment.ServiceDeployer deploy
INFO: Deploying Web service: version.aar
done
echo! ... echo!
org.apache.axis2.axis2userguide.Axis2UserGuideServiceStub$NoParametersResponse@11d0a4f
true
23872983
C:\Axis2\axis2-1.2-RC3\bin>
```

Figura II. 3 Resultado en Consola de la generación con Axis2

Fuente: Axis2 [2].

Como Paso final y para ver los resultados del consumo, se necesita realizar una aplicación web para que esta a su vez haga uso de las clases y métodos generados por el AXIS2.

2.5 METRO

2.5.1 Definición

Metro es un framework de servicios web, de alto rendimiento, extensible y fácil de usar. Se trata de una ventanilla única para todas sus necesidades de servicio web, desde el servicio web más sencillo hasta un servicio web fiable, que implica incluso servicios .Net. El framework de servicios web Metro es una parte de la comunidad *GlassFish*, pero puede también ser utilizado fuera de *GlassFish* [6].

Los componentes de metro incluyen WSIT, JAXB y JAX-WS, como pilares para la creación de servicios y clientes interoperables.

2.5.2 Web Service Interoperability Technologies (WSIT)

Durante varios años Sun actualmente Oracle ha trabajado estrechamente con Microsoft para garantizar la interoperabilidad de las tecnologías de servicios web de la empresa, tales como la seguridad, la mensajería fiable y transacciones atómicas. La parte de Metro se conoce como WSIT (Web Service Interoperability Technologies). Subsistema WSIT Metro es una implementación de una serie de especificaciones de servicios web abiertas para apoyar las funciones de empresa. Además de la seguridad, la mensajería fiable y transacciones atómicas, Metro incluye un programa previo y la tecnología de configuración. Figura 2.4.1 se puede apreciar las características de metro [12].

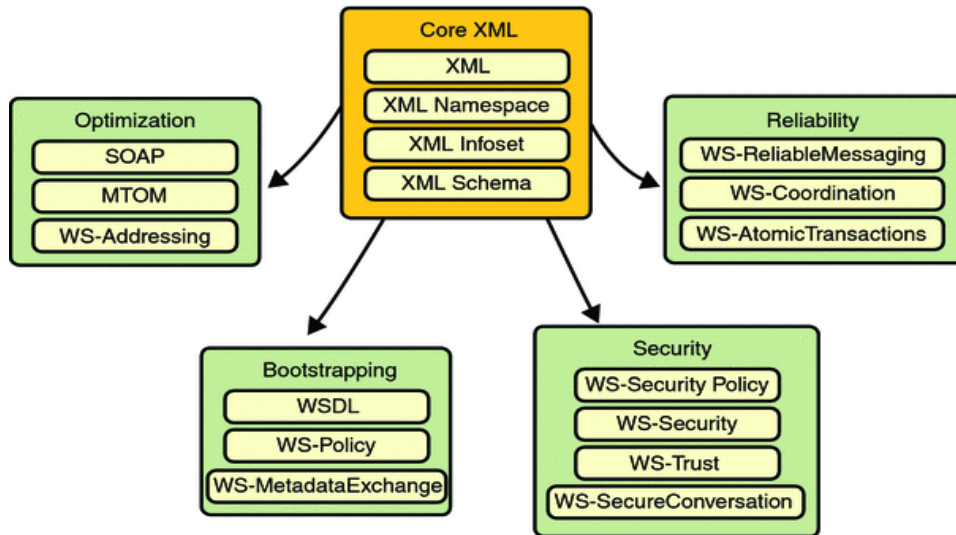


Figura II. 4 Características de los servicios web WSIT de Metro

Fuente: Metro [12].

Comenzando por la compatibilidad con XML integrada en la plataforma Java, Metro utiliza o amplía funciones existentes y añade compatibilidad adicional para servicios web interoperables. Las funciones son las siguientes [12]:

- Secuencia de arranque y configuración].
- Tecnología de optimización de mensajes.
- Tecnología de mensajería fiable.
- Tecnología de Seguridad.

2.5.3 Java Architecture XML Binding (JAXB)

Forma parte Metro, como proyecto para JAXB proporcionado implementación de referencia es desarrollado por JCP para JAXB [11].

JAXB es un acrónimo de la arquitectura Java para XML Binding. JAXB proporciona API para acceder y procesar un documento XML. JAXB es intermediario entre instancias de documentos XML y Java. Paso importante en el uso de JAXB es la creación de los POJOs Java. Esquema XML puede ser utilizado por un compilador de unión JAXB para generar clases Java. Esas clases java generados coinciden con el esquema XML y serán cargados en tiempo de ejecución de la aplicación. Si no se tiene el esquema XML, entonces podemos se puede codificar manualmente las clases POJO y anotar con anotaciones JAXB. Se puede crear instancias de esas clases luego leen o escriben en XML. XJC - `com.sun.tools.xjc.XJCTask` es el compilador de enlace proporcionado por el Metro. Las clases JAXB generados a partir de un esquema XML son POJOs con las anotaciones JAXB estándar. Ellos están destinados a ser compatible con otras implementaciones JAXB [11].

2.5.4 Relación Metro y .NET WCF

Interoperabilidad de servicios Web fue una iniciativa de Sun y Microsoft. Metro es el producto de la iniciativa de interoperabilidad de servicios Web de Sun. Windows Communication Foundation (WCF) en .NET es el modelo de programación unificado de Microsoft para la creación de sistemas conectados. WCF, que está disponible como parte del producto 3.x .NET Framework. Incluye interfaces de programación de aplicaciones (API) para la construcción, los servicios de seguros confiables y fueron contratados web que interactúan con las plataformas que no son de Microsoft [12].

Sun Microsystems y Microsoft probaron conjuntamente Metro contra WCF para asegurar que los clientes de servicios Web de Sun (consumidores) y servicios web (productores), de

hecho, interoperar con aplicaciones de servicios web de WCF y viceversa. Esta prueba asegura que los siguientes objetivos de interoperabilidad se realizan [12].

Los clientes Metro de servicios web pueden acceder y consumir servicios web de WCF. Los clientes WCF de servicios web pueden acceder y consumir servicios web de Metro. Sun proporciona Metro en la plataforma Java y Microsoft proporciona WCF en los .NET 3.0 y .NET 3.5 plataformas [12].

2.5.5 Usando Metro

Herramientas de metro

Metro provee las siguientes herramientas para el desarrollo de aplicaciones con servicios web:

Wsimport CLI.- Genera artefactos portables JAX-WS tales como: Interfaz de Extremo de Servicio (SEI), Servicio, Clases mapeadas de wsdl:fault (if any), Async Reponse Bean derived response wsdl:message (if any), tipos de valores JAXB. Estos artefactos pueden ser empaquetados en un archivo WAR con los documentos WSDL y de esquema junto con la implementación de punto final para ser desplegados. `wsimport -p stockquote http://stockquote.xyz/quote?wsdl` [12].

Wsimport Ant Task.- Esta Herramienta permite mediante un script tipo xml ejecutar comandos WSImport [23].

Wsgen CLI.- La herramienta lee una clase de servicio web de punto final y genera todos los artefactos necesarios para el despliegue de servicios web, y la invocació [12].

Wsgen Ant Task.- Esta Herramienta permite mediante un script tipo xml ejecutar comandos Wsgen CLI [12].

Apt.- Proporciona una instalación para el procesamiento de programáticamente las anotaciones añadidas a Java JSR 175, Facilidad de metadatos para el lenguaje de programación Java TM. En resumen, JSR 175 permite a los programadores declarar nuevos tipos de modificadores estructurados que pueden estar asociados con los elementos del programa, campos, métodos, clases, etc [12].

Usando Metro 2.0

Metro 2.0 esta suministrado con el API JAX- WS 2.2, mientras que Java SE 6 Update Release 4 y posteriores contienen la API JAX -WS 2.1 y versiones mayores Java SE 6 contienen JAX -WS 2.0. A menos que se haya tomado precauciones, las aplicaciones que utilizan la API de JAX -WS se ejecutarán con la versión de la API JAX -WS incorporado en Java SE 6 y no con la JAX- WS 2.2 API integrada en Metro en el classpath de la aplicación. Las aplicaciones que utilizan las nuevas funciones de la JAX -WS 2.2 API u otra funcionalidad Metro 2.0, por tanto, no se ejecutará. No se debe instalar cualquier otra biblioteca JAX- WS o metro que los que se mencionan a continuación en un directorio endosado al mismo tiempo, de lo contrario el código de metro puede no ser capaz de cargar las clases desde el classpath de la aplicación [12].

Tomcat y GlassFish V2.x.- Si se está ejecutando una aplicación en Tomcat o GlassFish V2.x, se debe asegurar de que se está utilizando el script de instalación metro-on-glassfish.xml o metro - on- tomcat.xml. Se copiarán todos los archivos en los directorios

específicos respaldados Tomcat/ GlassFish. Sin ese paso, las aplicaciones web y los EJB no será capaz de recoger la API JAX- WS 2.2 [12].

GlassFishV3.- Si se está ejecutando una aplicación en GlassFish V3, es necesario instalar Metro a través del Centro de Actualización de GlassFish , esta es la única forma de garantizar que Metro está instalado correctamente en GlassFish V3 y se encargará de copiar las bibliotecas de metro en la en los directorios de Glassfish [12].

Si desea ejecutar una aplicación o cliente de servicio Web fuera del Tomcat o contenedores de GlassFish, se tiene que instalar el archivo - api webservices.jar en el JRE respaldó directorio \$ JRE_HOME /lib/endorsed (o \$ JDK_HOME / jre / lib / endorsed). El directorio puede no existir todavía y en ese caso usted tendrá que crear usted mismo [12].

Generación del cliente.- En el modelo de programación JAX-WS, para desarrollar un cliente de servicios web se compila el WSDL desplegado usando wsimport y después en tiempo de ejecución el mismo WSDL se utiliza para determinar la información de enlace. El WSDL predeterminado utilizado puede determinarse mirando en la subclase javax.xml.ws.Service generada por wsimport [12].

La API de servicio para pasar la información

Catalogo Xml .- Crear un archivo de catálogo, META-INF/jax-ws-catalog.xml [12].

Usando el interruptor-wsdlLocation.- Ejecutar el comando proporcionar el valor de ubicación de WSDL que es relativo a la clase de servicio generado y proporcionar el valor

de ubicación de WSDL que es relativo a la clase de servicio generado y que necesita para poner este archivo WSDL en esta ubicación relativa [12].

2.5.6 Seguridad de Clientes mediante de SSL

Para configurar el sistema para SSL es necesario añadir el certificado del servidor al almacén de claves y archivo truststore con el fin de establecer una mutua confianza entre cliente y servidor. Esto hace innecesario establecer las propiedades del sistema para apuntar al store de los clientes, ya que tanto GlassFish y NetBeans IDE son conscientes de estos certificados y punto a ellos de forma predeterminada [12].

En general, para el lado del cliente de SSL que no se va a utilizar los mismos certificados para el cliente y el servicio. En ese caso, es necesario definir los almacenes de certificados de cliente estableciendo el sistema de propiedades *-Djavax.net.ssl.trustStore* , *-Djavax.net.ssl.keyStore* , *-Djavax.net.ssl.trustStorePassword* , and *-Djavax.net.ssl.keyStorePassword* en el contenedor de la aplicación cliente [12].

Se puede especificar las variables de entorno de almacén de claves y almacén de confianza estableciendo la variable de entorno VMARGS a través del entorno de shell o dentro de un script Ant, o pasando al iniciar NetBeans IDE desde la línea de comandos. Por ejemplo, en este último caso, se debe especificar los valores de propiedades de la siguiente manera [12].

-J-Djavax.net.ssl.trustStore=as-install/domains/domain1/config/cacerts.jks.

-J-Djavax.net.ssl.keyStore=as-install/domains/domain1/config/keystore.jks.

-J-Djavax.net.ssl.trustStorePassword=changeit.

CAPÍTULO III

ANÁLISIS COMPARATIVO DE METRO Y AXIS2 PARA EL DESARROLLO DE APLICACIONES QUE CONSUMAN SERVICIOS WEB WCF

3.1 Introducción

En este capítulo se realizará el análisis comparativo entre los frameworks de servicios web Metro y Axis2 orientados desde el lado cliente y tiene como objetivo demostrar los beneficios y debilidades de cada uno, en base a parámetros de comparación y prototipos de prueba, desarrollados para cada framework.

Al finalizar la comparación y evaluación de resultados, se realizará la demostración de la hipótesis previamente planteada y se seleccionará el framework de servicios web más adecuado para el desarrollo del sitio de información estudiantil de la ESPOCH.

3.2 Definición de los Parámetros de Comparación

Para realizar la comparación de los frameworks de servicios web Metro y Axis2 se establecen parámetros de comparación, los cuales contienen indicadores y que servirán para la realización del presente análisis comparativo, los mismos que se han sido seleccionados por el autor de la tesis, en base a la información obtenida.

Rendimiento.- Mediante este parámetro se pretende mostrar la eficacia con la que el framework de servicios web administra procesos de rendimientos tales como: tiempo de respuesta y la cantidad de requerimientos ejecutados, comúnmente llamado Throughput.

Requerimientos de Hardware.- Con este parámetro se pretende determinar los niveles de memoria RAM y procesador necesarios para la satisfactoria ejecución del framework de servicios web.

Simplicidad.- Mediante este parámetro se demuestra el nivel de sencillez que posee el framework de servicios web a la hora de instruirse e implementarlo.

En la Tabla III.I se puede observar en resumen los parámetros de comparación junto a sus respectivos indicadores y definiciones de estos, que se emplearán en el presente análisis.

Tabla III. I Parámetros e indicadores de comparación

Parámetro	Indicador	Definición Indicador
Rendimiento	I1. Throughput	Número de peticiones ejecutadas por el framework de servicios web en un segundo.
	I2. Tiempo de Respuesta	Tiempo que transcurre desde que se envía la petición hasta recibir una respuesta del framework de servicios web.

Requerimientos de Hardware	I3. Memoria RAM	Cantidad de espacio en Memoria RAM necesaria para almacenar un promedio de peticiones concurrentes al framework de servicios web.
	I4. Uso de Procesador	Porcentaje de uso del procesador por el framework de servicios web.
Simplicidad	I5. Disponibilidad de Información	Información disponible acerca de cada framework de servicios web.
	I6. Facilidad de Desarrollo	Facilidad de desarrollo para la integración de la aplicación.

Fuente: Autor

3.3 Método para la Evaluación de Resultados

Los frameworks de servicios web serán evaluados de forma comparativa mediante datos estadísticos calculados con los resultados de las pruebas realizadas a los prototipos en cada parámetro definido anteriormente. Para comparar el nivel de cumplimiento de cada framework de servicios web se creará tablas de calificación, con lo cual se pretende exponer la comparación entre los dos frameworks. Las tablas se emplearán para la realización de conclusiones de las pruebas.

La evaluación de los dos frameworks de servicios web, se lo hará en mediante a los parámetros descritos con anterioridad, con lo cual se obtendrán resultados cuantitativos y cualitativos que permitirán realizar una selección sustentada de uno de los dos frameworks de servicios web analizados.

En la Tabla III.II se ilustra los niveles de cumplimiento que serán utilizados en el desarrollo de la presente investigación, de esta forma se muestra de mejor manera el grado de desempeño del framework de servicios web, y la ventaja del uno sobre el otro.

Tabla III. II Niveles de Cumplimiento

VALORACIÓN	CALIFICACIÓN	FORMA GRÁFICA	VALOR	DESCRIPCIÓN
Excelente	>75% y <=100%		4	Todas las expectativas cumplidas
Buena	>50% y <=75%		3	Mayoría de las expectativas cumplidas
Regular	>25% y <=50%		2	Pocas de las expectativas cumplidas
Malo	>=0% y <=25%		1	Ninguna expectativa cumplida

Fuente: Autor

La calificación de cada parámetro se realiza mediante la suma de los puntajes obtenidos en el análisis, haciendo uso de las siguientes formulas:

$$C_m = \sum_{i=0}^n V_i \quad (1)$$

$$C_a = \sum_{i=0}^n V_i \quad (2)$$

$$C_{max} = \sum_{i=0}^n VM_i \quad (3)$$

De donde:

n: Número de indicadores del parámetro

V_i: Valor de calificación de cada indicador

VM_i : Valor máximo de calificación de cada indicador

C_m : Calificación para el framework de servicios web Metro en el parámetro

C_a : Calificación para el framework de servicios web Axis2 en el parámetro

C_{max} : Calificación máxima sobre el que se mide el parámetro.

Las formulas 4 y 5 sirven para calcular los porcentajes de cada uno de los parámetros de comparación en cada framework de servicios web.

$$P_m = \left(\frac{C_m}{C_{max}} \right) * 100\% \quad (4)$$

$$P_a = \left(\frac{C_a}{C_{max}} \right) * 100\% \quad (5)$$

De donde:

P_m : Porcentaje del framework de servicios web Metro.

P_a : Porcentaje del framework de servicios web Axis2.

3.4 Instrumentos de Medición

Los instrumentos de medición son herramientas que permiten medir los indicadores de los parámetros de comparación planteados anteriormente, sometiendo a prueba a los prototipos de cada framework de servicios web. Estos instrumentos se han seleccionado en base a los parámetros de comparación y según criterio del autor de la presente investigación. Las herramientas a emplearse son:

JMeter.- Es una herramienta Open Source que permiten realizar pruebas de funcionalidad y con ello poder medir la eficacia de una aplicación web. Mediante esta herramienta se obtendrá resultados para los indicadores del parámetro rendimiento.

Java Visual VM.- Es una herramienta incluida en el JDK y permite visualizar el comportamiento de las aplicaciones en relación al consumo y uso de la memoria RAM y procesador respectivamente. Con esta se pretende medir los recursos de hardware necesarios para cada framework.

Sitios Oficiales de los Frameworks.-Sitio web de información oficial del framework de servicios web Metro denominado Glassfish Metro y el sitio oficial de Axis2 llamado Apache Axis2/Java, que ofrecen documentación sobre guías de uso, administración, implementación, ejemplos, etc. Mediante la documentación oficial, se pretende demostrar los niveles de interoperabilidad y complejidad de cada framework de servicios web.

3.2. Descripción del Escenario de Pruebas

Se realizara un único escenario de prueba que contará con equipo de hardware y software, y servirá de ambiente para ejecutar los prototipos de cada framework de servicios web, a fin de obtener resultados que permitan en lo posterior evaluar a los mismos.

3.2.1 Equipo Utilizado

El escenario de pruebas está constituido por dos máquinas, un servidor y otra como cliente, en donde se han ejecutado todas las pruebas. Las máquinas son computadoras portátiles que tienen las especificaciones de hardware detalladas en la Tabla III. III.

Tabla III. III Especificaciones de hardware del equipo utilizado

Característica	Maquina Servidor	Maquina Cliente
Procesador	Intel Core i5 2450m 2.50GHz	Intel Core i5 M430 2.27GHz
Memoria RAM	8GB	4GB
Disco Duro	600GB	300GB

Fuente: Elaborada por el autor

3.2.2 Software Utilizado

El las especificaciones y componentes de software que han sido parte del escenario y se ha utilizado para la realización de las pruebas se detallan en la Tabla III. IV.

Tabla III. IV Especificaciones de software utilizado

Característica	Descripción
Sistema Operativo	Windows 7 Ultimate
IDE	Visual Studio 2010, Netbeans 7.3, Eclipse Indigo Release 2
Servidor Web	IIS7, GlassFish Server Open Source Edition 3.1.2.2
.Net Framwork	4.0
JDK	1.7.0_21
Pruebas de Carga	Jmeter 2.9
Administración de Recursos	Java Visual VM
BackEnd	Módulo contiene los servicios web WCF desplegado en el servidor IIS7

Fuente: Autor

Previo a la construcción de los prototipos y como requisito para estos se desarrolló el módulo BackEnd, que provee los servicios web WCF.

BackEnd.- Módulo Servidor, que contiene los servicios web. Este fué desarrollado en tecnología .Net con el lenguaje de programación c# y se constituyó de las siguientes capas:

Base de datos.- Lugar donde persisten la información del estudiante, para los prototipos se seleccionó el gestor de base de datos Microsoft SQL Server 2008, debido al caso aplicativo.

Acceso a datos.- Capa donde se relaciona la base de datos con los servicios web, para esto se hizo uso de Entity Framework 4.0, debido a que es compatible con WCF.

Servicios web.- Capa que expone los servicios web de información del estudiante, los mismos que fueron desarrollados mediante el Framework de servicios web para .Net WCF 4.0. Los servicios web WCF fueron desplegados en el servidor web IIS7.0, en un sistema operativo Windows 7.

3.2.2 Prototipos

La finalidad de los prototipos es proveer una fuente de información oportuna para con esta realizar una posterior comparación y verificación de las tecnologías con las que se han desarrollado los mismos.

Para el presente proyecto se construyó dos prototipos, uno para Metro y otro para Axis2 que sirvieron para realizar las pruebas de carga y con ello comparar los frameworks de servicios web Metro y Axis2. Estos fueron pequeñas aplicaciones web que tenían la funcionalidad de mostrar una lista de estudiantes y se desarrollaron bajo una arquitectura SOA.

3.2.2.1 Prototipo Metro

Es una aplicación web cliente que contiene el consumo de los servicios web WCF mediante el framework de servicios web Metro y la navegabilidad para el usuario. Está constituido de las siguientes capas:

Proxy.- Esta capa fue la encargada de la integración con el BackEnd, aquí se concentró el consumo de los servicios web WCF mediante el framework de servicios web Metro.

Presentación.- Esta capa fue la encargada de recibir las peticiones del usuario, invocar a la lógica de negocio del consumo de servicios web en la capa proxy y devolver resultados al mismo. Esta capa fue desarrollada con el framework JSF y Primefaces.

El prototipo fue desplegado en el servidor web GlassFish Server 3.1.2 Open Source Edition, en un sistema operativo Windows7.

3.2.2.1 Prototipo Axis2

Al igual que el prototipo Metro es una aplicación web cliente que contiene el consumo de los servicios web WCF mediante el framework de servicios web Axis2 y la navegabilidad para el usuario. Constituido de las siguientes capas:

Proxy.- Esta capa fue la encargada de la integración con el BackEnd, aquí se concentró el consumo de los servicios web WCF mediante el framework de servicios web Axis2.

Presentación.- Esta capa fue la encargada de recibir las peticiones del usuario, invocar a la lógica de negocio del consumo de servicios web en la capa proxy y devolver resultados al mismo. Esta capa fue desarrollada con el framework JSF y Primefaces.

El prototipo fue desplegado en el servidor web GlassFish Server 3.1.2 Open Source Edition, en un sistema operativo Windows7.

Una vez definidas el hardware, software y prototipos se dejó preparado el escenario para realizar las pruebas que proveerán los datos necesarios para la comparación de los frameworks de servicios web Metro y Axis2.

3.3. Desarrollo de las pruebas con los parámetros de comparación

Se inicia la realización de las pruebas en cada uno de los parámetros e indicadores de comparación establecidos para los frameworks de servicios web, en las que se detallan el proceso de obtención de resultados. Los mismos que serán evaluados, calificados, se los graficará y se realizará su respectiva interpretación.

Para los parámetros rendimiento y requerimientos de hardware, se empleará dos indicadores adicionales tales como la frecuencia y el número de usuarios concurrentes por prueba.

La frecuencia es el número de ejecuciones realizadas sobre las pruebas, que para el presente trabajo de investigación posee un valor de 10, el mismo que se ha estimado según criterio del autor.

El número de usuarios concurrentes, hace referencia a la cantidad promedio de usuarios conectados simultáneamente haciendo uso del prototipo. Este número de usuarios es calculado mediante la siguiente fórmula estadística que es aplicada cuando no se conoce la población universal.

$$n = \frac{Z_{\alpha}^2 \times p \times q}{d^2}$$

De donde:

n: Tamaño de la muestra.

Z_{α}^2 : Nivel de confianza elegido, determinado por el valor α . Para una confianza del 95% ($\alpha = 0.05$), este valor es de 1,96, según la tabla de valores Z más usados.

p: Proporción esperada, para el caso de estudio se considera una proporción del 50% para reducir el factor de error (0.5).

q: Proporción $1 - p$

d: Margen de error considerado para la prueba, considerado el 5%.

$$n = \frac{1.96^2 \times 0.5 \times 0.5}{0.05^2} = 384.16$$

Aproximando al inmediato superior la cantidad de usuarios concurrentes es 385.

3.3.1. Rendimiento

Mediante este parámetro se pretende mostrar la eficacia con la que el framework de servicios web administra procesos de rendimientos tales como: tiempo de respuesta y la cantidad de requerimientos ejecutados, comúnmente llamado Throughput.

Tabla III. V Indicadores del parámetro rendimiento

Indicador	Descripción
I1. Throughput	Número de peticiones ejecutadas por el framework de servicios web en un segundo.
I2. Tiempo de Respuesta	Tiempo que transcurre desde que se envía la petición hasta recibir una respuesta del framework de servicios web.

Fuente: Autor

A partir de las aplicaciones prototipo descritas anteriormente se ejecutaron un grupo de pruebas de carga de manera automatizada con la herramienta JMeter, dando de esta manera una mayor certificación a los resultados obtenidos.

Para la obtención de los resultados se realizó el siguiente procedimiento:

1. Ejecución de la herramienta JMeter, con el objetivo de realizar la prueba de carga.
2. Creación de un plan de Pruebas para cada uno de los prototipos
3. Adición de un grupo de hilos, para simular el número de peticiones http realizadas al prototipo desplegado en Glassfish.

4. Adicionar en el grupo de hilos un mostrador y realizar la configuración de la petición HTTP.
5. Agregar dentro del mismo grupo de hilos un receptor de tipo Resumen y Agregado.
6. Ejecutar la prueba.
7. Observar resultados en los receptores anteriormente añadidos.
8. Ir a 6.

La aplicación prototipo realiza una Request al servicio web WCF para obtener una lista de registros y este le responde con un Response con los datos en formato xml, para ser interpretados por el framework de servicios web que mediante el programa cliente los transforma en estructuras de tipo java, para que estos sean presentados finalmente al usuario mediante una combinación del framework JSF y Primefaces.

II. Throughput

Este indicador hace referencia a la cantidad de peticiones ejecutadas que atiende el framework de servicios web, en un período determinado de tiempo, que para este proyecto la unidad será en segundos.

Luego de ejecutar las pruebas se obtiene resultados individuales con los cuales se calcula la media o promedio, los errores relativos y porcentuales, desviación estándar e intervalos de confianza, con la finalidad de establecer datos estadísticos confiables que servirán a su vez en la evaluación del indicador. Las pruebas ejecutadas se pueden apreciar en el Anexo1 y sus resultados en resumen en la Tabla III.VI

Tabla III. VI Resultados de las pruebas para el throughput

Prototipo	Prueba	(x ₀)	(x)	(x ₀ - x)	$\frac{(x_0 - x)}{x_0}$	(\bar{x})	($\bar{x}-x$)	($\bar{x} - x$) ²
Metro	1	27.3	27	0.3	0.0110	275/10=27.5	0.5	0.25
	2	27.2	27	0.2	0.0074		0.5	0.25
	3	27.1	27	0.1	0.0037		0.5	0.25
	4	27.1	28	0.1	0.0037		-0.5	0.25
	5	27.3	27	0.3	0.0110		0.5	0.25
	6	28.1	28	0.1	0.0036		-0.5	0.25
	7	27.0	27	0	0		0.5	0.25
	8	27.6	28	0.4	0.0145		-0.5	0.25
	9	27.7	28	0.3	0.0108		-0.5	0.25
	10	27.7	28	0.3	0.0108		-0.5	0.25
Axis2	1	12.9	13	0.1	0.0078	130/10 = 13	0	0
	2	12.7	13	0.3	0.0236		0	0
	3	12.9	13	0.1	0.0078		0	0
	4	12.5	13	0.5	0.04		0	0
	5	12.7	13	0.3	0.0236		0	0
	6	12.6	13	0.4	0.0317		0	0
	7	12.9	13	0.1	0.0078		0	0
	8	13.1	13	0.1	0.0078		0	0
	9	12.7	13	0.3	0.0236		0	0
	10	12.6	13	0.4	0.0317		0	0

Fuente: Autor

Cálculo de errores para resultados de las pruebas en los prototipos

$$\epsilon_a = \frac{(\sum_{i=1}^n (|x_0 - x|))}{n}$$

$$\epsilon_r = \frac{\left(\sum_{i=1}^n \frac{(|x_0 - x|)}{x_0}\right)}{n}$$

$$\epsilon_p = \frac{\left(\sum_{i=1}^n \frac{(|x_0 - x|)}{x_0}\right)}{n} \times 100\%$$

De donde:

ϵ_a : Es el error absoluto de las pruebas.

ϵ_r : Es el error relativo de las pruebas

ϵ_p : Es el porcentaje de error relativo de las pruebas

$(\sum_{i=1}^n (|x_0 - x|))$: Es la sumatoria de la diferencia absoluta entre el valor verdadero y el valor tomado.

x_0 : Es el resultado inicial obtenido de la prueba.

x : Es el resultado obtenido de la prueba.

n : Es el número de pruebas realizadas.

Para la prueba con Metro el error absoluto, relativo y porcentual con respecto a los resultados son:

$$\epsilon_{am} = \frac{(2.1)}{10} = 0.21 \text{ peticiones/segundo}$$

$$\epsilon_{rm} = \frac{(0.0765)}{10} = 0.00765$$

$$\epsilon_{rm} = \frac{(0.0765)}{10} \times 100\% = 0.765\%$$

Para la prueba con Axis2 el error absoluto, relativo y relativo porcentual con respecto a los resultados son:

$$\varepsilon_{am} = \frac{(2.6)}{10} = 0.26 \text{ peticiones/segundo}$$

$$\varepsilon_{rm} = \frac{(0.2054)}{10} = 0.0205$$

$$\varepsilon_{rm} = \frac{(0.2054)}{10} \times 100\% = 2.054\%$$

Cálculo de la desviación estándar:

$$S = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}}$$

De donde:

S: Desviación estándar

\bar{x} : Es el valor promedio de las prueba.

n: Es el número de pruebas realizadas.

El valor de la desviación estándar para Metro es:

$$S_m = 2.5/9 = 0.2778$$

El valor de la desviación estándar para Axis2 es:

$$S_a = 0/9 = 0$$

Cálculo del intervalo de confianza:

$$IC = \bar{x} \pm Z \left(\frac{S}{\sqrt{n-1}} \right)$$

De donde:

IC : Intervalos de confianza.

\bar{x} : Media de los valores tomados

Z: Es el nivel de confianza elegido, determinado por el valor de α . Para una confianza del 95% ($\alpha=0,05$), este valor es de 1,96 según la tabla de valores *Z* más usados,

S: Desviación estándar

n: Es el número de pruebas realizadas.

Los intervalos de confianza para Metro son:

$$IC_m = 27.5 \pm 1.96 \left(\frac{0.2778}{\sqrt{9}} \right)$$

$$IC_m = 27.5 \pm 0.1815$$

Los intervalos de confianza para Axi2 son:

$$IC_a = 13 \pm 1.96 \left(\frac{0}{\sqrt{9}} \right)$$

$$IC_a = 13 \pm 0.0$$

De esta forma el resumen de los resultados en número de peticiones sobre segundo de las pruebas realizadas a los prototipos se aprecian en la Tabla Tabla III. VIII.

Tabla III. VII Resumen de los resultados de la medición del throughput

	Frecuencia	Promedio	Error Relativo	Porcentaje Error	$S_{\bar{x}}$	IC
Metro	10	27.5	0.00765	0.765%	0.2778	[27.32, 27.68]
Axis2	10	13	0.0205	2.054%	0	[13.0, 13.0]





Fuente: Autor

Interpretación de Resultados

Metro presenta un mayor número de peticiones o transacciones realizadas por segundo al responder a 27.5 peticiones /segundo en comparación a Axis2 que responde a 13 peticiones/segundo, indicando una relación de 47.27%.

Calificación

Tabla III. VIII Calificación del indicador throughput

Rango de valores	Valoración Cuantitativa	Valoración Cualitativa	Forma Gráfica
24 a 30	4	Excelente	
17 a 23	3	Bueno	
10 a 16	2	Regular	
Menor a 10	1	Malo	

Fuente: Autor

El framework Metro atiende a 27.5 solicitudes en un segundo, con lo cual y en base a la tabla de calificación para el presente indicador, se le asigna una calificación Excelente. Su equivalencia grafica será 4 estrellas.

Axis2 al atender 13 peticiones en un segundo, se le asigna al indicador una calificación con valor Regular. Su equivalencia grafica será 2 estrellas.

I2: Tiempo de Respuesta

Este indicador mide el tiempo que pasa desde que se envía la petición hasta recibir una respuesta por parte del framework de servicios web. Dicho tiempo tendrá como unidad de medida el segundo.

Al ejecutar las pruebas se obtiene resultados individuales, con los cuales se calcula la media o promedio, los errores absolutos, relativos y porcentuales, desviación estándar e intervalos de confianza, con la finalidad de establecer datos estadísticos confiables que servirán a su vez en la evaluación del indicador. Las pruebas ejecutadas sobre este indicador, se puede apreciar en el anexo1 y su respectivo resumen en segundos en la Tabla III.IX.

Tabla III. IX Resultados de las pruebas para el tiempo de respuesta

Prototipo	Prueba	x_0	(x)	$ x_0 - x $	$\frac{(x_0 - x)}{x_0}$	(\bar{x})	$(\bar{x}-x)$	$(\bar{x} - x)^2$
Metro	1	6.326	6.3	0.026	0.00411	64.8/10= 6.48	0.18	0.0324
	2	6.527	6.5	0.027	0.00413		-0.02	0.0004
	3	6.708	6.7	0.008	0.00119		-0.22	0.0484

Prototipo	Prueba	x_0	(x)	$ x_0 - x $	$\frac{(x_0 - x)}{x_0}$	(\bar{x})	$(\bar{x}-x)$	$(\bar{x} - x)^2$
	4	6.705	6.7	0.005	0.00075		-0.22	0.0484
	5	6.346	6.4	0.054	0.0085		0.08	0.0064
	6	6.458	6.5	0.042	0.0065		-0.02	0.0004
	7	6.759	6.8	0.041	0.00607		-0.32	0.1024
	8	6.379	6.4	0.021	0.00329		0.08	0.0064
	9	6.314	6.3	0.014	0.00222		0.18	0.0324
	10	6.222	6.2	0.022	0.00354		0.28	0.0784
Axis2	1	13.576	13.6	0.024	0.00177	141.2/10= 14.12	0.52	0.2704
	2	13.706	13.7	0.007	0.00051		0.42	0.1764
	3	14.296	14.3	0.004	0.00028		-0.08	0.0064
	4	14.485	14.5	0.015	0.00104		-0.38	0.1444
	5	14.218	14.2	0.018	0.00127		0.08	0.0064
	6	14.546	14.6	0.054	0.0037		-0.48	0.2304
	7	14.126	14.1	0.026	0.00184		0.02	0.0004
	8	13.581	13.6	0.019	0.00139		0.52	0.2704
	9	14.302	14.3	0.002	0.00014		-0.18	0.0324
	10	14.367	14.4	0.024	0.00167		0.28	0.0784

Fuente: Autor

Cálculo de errores para resultados de las pruebas en los prototipos

$$\varepsilon_a = \frac{(\sum_{i=1}^n (|x_0 - x|))}{n}$$

$$\varepsilon_r = \frac{(\sum_{i=1}^n \frac{(|x_0 - x|)}{x_0})}{n}$$

$$\varepsilon_p = \frac{(\sum_{i=1}^n \frac{(|x_0 - x|)}{x_0})}{n} \times 100\%$$

Para las pruebas con Metro el error absoluto, relativo y porcentual con respecto a los resultados son:

$$\varepsilon_{am} = \frac{(0.458)}{10} = 0.0458 \text{ segundos}$$

$$\varepsilon_{rm} = \frac{(0.04101)}{10} = 0.0041 \text{ segundos}$$

$$\varepsilon_{rm} = \frac{(0.04101)}{10} \times 100\% = 0.41\%$$

Para la prueba con Axis2 el error absoluto, relativo y relativo porcentual con respecto a los resultados son:

$$\varepsilon_{am} = \frac{(0.193)}{10} = 0.0193 \text{ segundos}$$

$$\varepsilon_{rm} = \frac{(0.01501)}{10} = 0.0015 \text{ segundos}$$

$$\varepsilon_{rm} = \frac{(0.01501)}{10} \times 100\% = 0.15\%$$

Cálculo de la desviación estándar:

$$S = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}}$$

El valor de la desviación estándar para Metro es:

$$S_m = 0.356/9 = 0.03956$$

El valor de la desviación estándar para Axis2 es:

$$S_a = 1.216/9 = 0.135$$

Cálculo del intervalo de confianza:

$$IC = \bar{x} \pm Z \left(\frac{S}{\sqrt{n - 1}} \right)$$

Los intervalos de confianza para Metro son:

$$IC_m = 6.48 \pm 1.96 \left(\frac{0.03956}{\sqrt{9}} \right)$$

$$IC_m = 6.48 \pm 0.03$$

Los intervalos de confianza para Axi2 son:

$$IC_a = 14.176 \pm 1.96 \left(\frac{0.135}{\sqrt{9}} \right)$$

$$IC_a = 14.12 \pm 0.09$$

De esta forma los resultados en la unidad segundos de las pruebas realizadas a los prototipos se presentan en la Tabla III.X.

Tabla III. X Resultados de la medición del tiempo de respuesta

	Frecuencia	Promedio	Error Relativo	Porcentaje Error	$S_{\bar{x}}$	IC
Metro	10	6.48	0.0041	0.41%	0.03956	[6.45, 6.51]
Axis2	10	14.12	0.0015	0.15%	0.135	[14.03, 14.21]

Fuente: Autor





Interpretación de Resultados

El framework de servicios web Metro presenta un menor tiempo de respuesta al responder 385 peticiones en 6.48 segundos en comparación al framework de servicios web Axis2 que responde en un tiempo de 14.12 segundos, indicado una relación de 45.63%.

Debido a que el indicador es un tiempo, los rangos de valores son inversos, ya que a menor tiempo mayor rendimiento.

Calificación

Tabla III. XI Calificación del indicador tiempo de respuesta

Rango de valores	Valoración Cuantitativa	Valoración Cualitativa	Forma Gráfica
Menor a 5	4	Excelente	
5 a 10	3	Bueno	
11 a 16	2	Regular	
Mayor a 16	1	Malo	

Fuente: Autor

El framework de servicios web Metro posee un tiempo de respuesta promedio de 6.48 segundos, con lo que en base a la tabla de calificación para el presente indicador se le asigna una calificación de Bueno. Su equivalencia grafica será de 3 estrellas.

El framework de servicios web Axis2 al tener un tiempo de respuesta promedio de 14.176 segundos, se le asigna al indicador una calificación con valor Regular. Su equivalencia gráfica será 2 estrellas.

3.3.1.3 Evaluación de resultados

Para evaluar el parámetro rendimiento se utiliza las calificaciones obtenidas por los indicadores en las pruebas realizadas y de las fórmulas (1, 2 y 3) previamente definidas en el método de evaluación.

Los resultados que se obtenga a partir del cálculo se presentarán en forma numérica y gráfica. Para después realizar una interpretación de los mismos

La calificación máxima del parámetro de comparación se establece mediante la suma de los valores máximos de calificación de cada indicador. Es así que:

$$C_{max} = \sum_{i=0}^n V_i = 4 + 4 = 8$$

La calificación numérica para el framework de servicios web Metro en el parámetro rendimiento, se calcula mediante la sumatoria de las calificaciones de los indicadores obtenidas en las pruebas. De esta forma:

$$C_m = \sum_{i=0}^n V_i = 4 + 3 = 7$$

La calificación numérica para el framework de servicios web Axis2 en el parámetro rendimiento, se calcula de igual forma a la del framework de servicios web Metro.

$$C_a = \sum_{i=0}^n V_i = 2 + 2 = 4$$

El nivel de cumplimiento del framework de servicios web Metro expresado en porcentajes es igual a la división de la calificación obtenida por este entre la calificación máxima del parámetro y todo esto multiplicado por el cien por ciento.

$$P_m = \left(\frac{C_m}{C_{max}} \right) \times 100\% = \left(\frac{7}{8} \right) \times 100\% = 87.5\%$$

De igual forma el nivel de cumplimiento del framework de servicios web Axis2 expresado en porcentajes es igual a la división de la calificación obtenida por este entre la calificación máxima del parámetro y todo esto multiplicado por el cien por ciento.

$$P_a = \left(\frac{C_a}{C_{max}} \right) \times 100\% = \left(\frac{4}{8} \right) \times 100\% = 50\%$$

Los resultados del cumplimiento de los frameworks de servicios web se expresan en el Figura III.1.

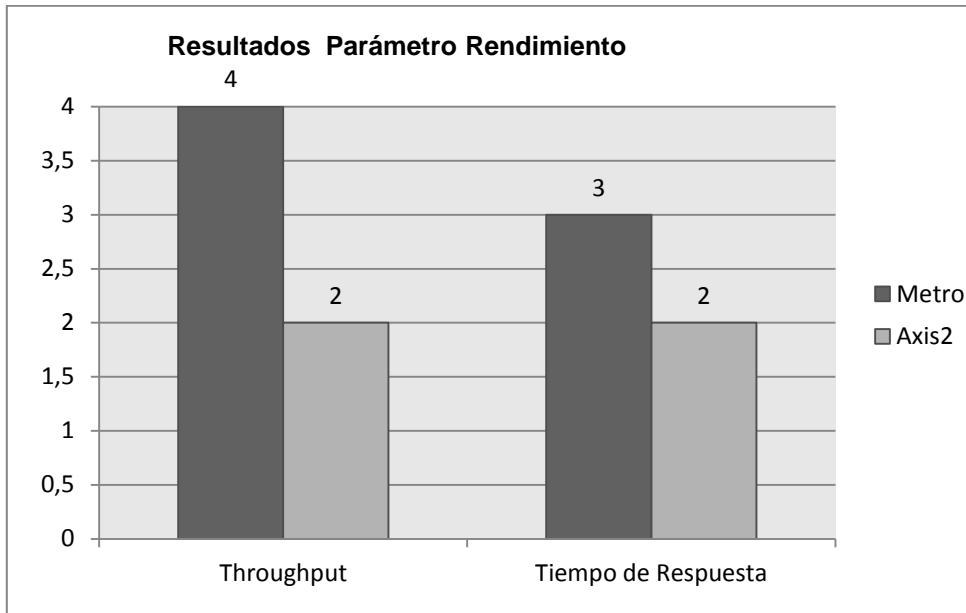


Figura III. 1 Resultados Parámetro Rendimiento

Fuente: Autor

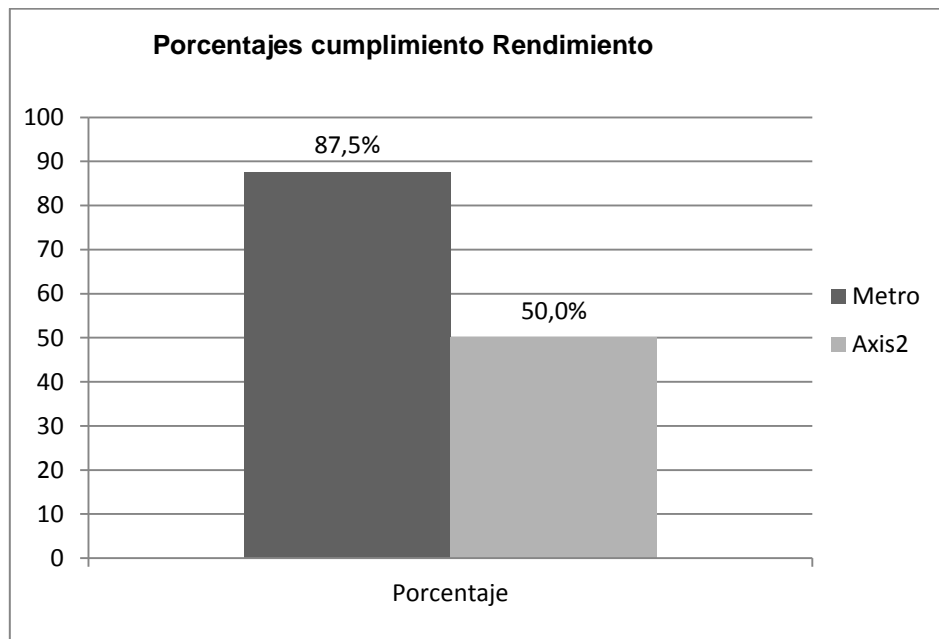


Figura III. 2 Porcentajes cumplimiento Rendimiento

Fuente: Autor

3.3.1.4 Interpretación de resultados

En base a la Tabla III.II que permite establecer el equivalente cualitativo de acuerdo a porcentajes, se valorará los resultados obtenidos por cada framework de servicios web.

Los resultados obtenidos en parámetro rendimiento revelan que el framework de servicios web Metro posee un nivel de cumplimiento del 87.5% de los indicadores establecidos, lo que equivale a **Excelente**. Mientras que el framework de servicios web Axis2 cumple con el 50% de los indicadores establecidos, lo que equivale a **Regular**.

3.3.2 Requerimientos de Hardware

Con este parámetro se pretende determinar los niveles de memoria RAM y procesador necesarios para la satisfactoria ejecución del framework de servicios web.

Tabla III. XII Indicadores del parámetro requerimientos de hardware

Indicador	Descripción
I3. Uso de Memoria RAM	Cantidad de espacio en Memoria RAM necesaria para almacenar un promedio de peticiones concurrentes al framework de servicios web.
I4. Uso de Procesador	Porcentaje de uso del procesador por el framework de servicios web.

Fuente: Autor

A partir de las aplicaciones prototipo descritas anteriormente se ejecutaron 10 pruebas de carga de manera automatizada con la herramienta JMeter, captando los resultados mediante la herramienta Java Visual VM.

Para la obtención de los resultados se realizó el siguiente procedimiento:

1. Ejecución de la herramienta JMeter, con el objetivo de realizar la prueba de carga.
2. Creación de un plan de Pruebas para cada uno de los prototipos
3. Adición de un grupo de hilos, para simular el número de peticiones http realizadas al prototipo desplegado en Glassfish.
4. Adicionar al grupo de hilos un mostrador y realizar la configuración de la petición HTTP.
5. Ejecutar la prueba.
6. Observar resultados en la herramienta de Java visual VM
7. Ir a 6.

I3: Uso de Memoria RAM

Este indicador se hace referencia a la cantidad de espacio en Memoria RAM necesaria para almacenar un promedio de peticiones concurrentes realizadas al framework de servicios web.

Luego de ejecutar las pruebas se obtiene resultados individuales con los cuales se calcula la media o promedio, los errores absolutos, relativos y porcentuales, desviación estándar e intervalos de confianza, con la finalidad de establecer datos estadísticos confiables que servirán a su vez en la evaluación del indicador. Las pruebas ejecutadas sobre el presente indicador se puede apreciar en el anexo2 y sus resultados en la Tabla III.XIII.

Tabla III. XIII Resultados de las pruebas para el uso de memoria RAM

Prototipo	Prueba	x_0	(x)	$ x_0 - x $	$\frac{(x_0 - x)}{x_0}$	(\bar{x})	$(\bar{x}-x)$	$(\bar{x} - x)^2$
Metro	1	75.208	75.2	0.008	0.00001	746.5/10= 74.65	-0.55	0.3025
	2	74.317	74.3	0.017	0.00023		0.35	0.1225
	3	70.785	70.8	0.015	0.00021		3.85	14.823
	4	70.997	71	0.003	0.00004		3.65	13.323
	5	72.951	73	0.49	0.00067		1.65	2.723
	6	79.648	79.7	0.052	0.00065		-5.05	25.503
	7	76.387	76.4	0.013	0.00017		-1.75	3.063
	8	79.487	79.5	0.013	0.00017		-4.85	23.523
	9	75.961	76	0.039	0.00513		-1.35	1.823
	10	70.576	70.6	0.024	0.00034		4.05	16.403
Axis2	1	110.762	110.8	0.038	0.00034	1151.4/10 = 115.14	4.34	18.836
	2	113.482	113.5	0.018	0.00016		1.64	2.689
	3	117.921	117.9	0.021	0.00018		-2.76	7.618
	4	115.256	115.3	0.044	0.00038		-0.16	0.026
	5	119.819	119.8	0.019	0.00016		-4.66	21.716
	6	108.138	108.1	0.038	0.00035		7.04	49.562
	7	119.288	119.3	0.012	0.0001		-4.16	17.306
	8	117.902	117.9	0.002	0.00002		-2.76	7.618
	9	116.486	116.5	0.014	0.00012		-1.36	1.850
	10	112.339	112.3	0.039	0.00035		2.84	8.066

Fuente: Autor

Cálculo de errores para resultados de las pruebas en los prototipos

$$\varepsilon_a = \frac{(\sum_{i=1}^n (|x_0 - x|))}{n}$$

$$\varepsilon_r = \frac{(\sum_{i=1}^n \frac{(|x_0 - x|)}{x_0})}{n}$$

$$\varepsilon_p = \frac{(\sum_{i=1}^n \frac{(|x_0 - x|)}{x_0})}{n} \times 100\%$$

Para las pruebas con Metro el error absoluto, relativo y relativo porcentual con respecto a los resultados son:

$$\varepsilon_{am} = \frac{(0.674)}{10} = 0.0674\text{MB}$$

$$\varepsilon_{rm} = \frac{(0.00762)}{10} = 0.00076\text{MB}$$

$$\varepsilon_{rp} = \frac{(0.00762)}{10} \times 100\% = 0.076\%$$

Para la prueba con Axis2 el error absoluto, relativo y relativo porcentual con respecto a los resultados son:

$$\varepsilon_{am} = \frac{(0.245)}{10} = 0.0245$$

$$\varepsilon_{rm} = \frac{(0.00216)}{10} = 0.000216\text{MB}$$

$$\varepsilon_{rp} = \frac{(0.00216)}{10} \times 100\% = 0.0216\%$$

Cálculo de la desviación estándar:

$$S = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}}$$

El valor de la desviación estándar para el framework de servicios web Metro es:

$$S_m = 101.609/9 = 11.29$$

El valor de la desviación estándar para el framework de servicios web Axis2 es:

$$S_a = 135.287/9 = 15.032$$

Cálculo del intervalo de confianza:

$$IC = \bar{x} \pm Z \left(\frac{S}{\sqrt{n - 1}} \right)$$

Los intervalos de confianza para Metro son:

$$IC_m = 74.65 \pm 1.96 \left(\frac{11.29}{\sqrt{9}} \right)$$

$$IC_m = 74.65 \pm 7.77$$

Los intervalos de confianza para Axi2 son:

$$IC_a = 115.14 \pm 1.96 \left(\frac{15.032}{\sqrt{9}} \right)$$

$$IC_a = 115.14 \pm 9.82$$

De esta forma el resumen de los resultados en MegaBytes de las pruebas realizadas a los prototipos se aprecian en la Tabla III. XIV.

Tabla III. XIV Resumen de los resultados de la medición del uso de memoria RAM

	Frecuencia	Promedio	Error Relativo	Porcentaje Error	$S_{\bar{x}}$	IC
Metro	10	74.65	0.00076	0.076%	11.29	[66.88, 82.42]
Axis2	10	115.14	0.000216	0.0216%	15.032	[105.32, 124.96]

Fuente: Autor





Interpretación de Resultados

El framework de servicios web Metro presenta una menor cantidad de uso de memoria RAM al ocupar 74.65MB en comparación al framework de servicios web Axis2 que hace uso de 115.14MB. Estableciendo una relación del 64.83%.

Debido a que el indicador es espacio en memoria, los rangos de valores son inversos, ya que el objetivo es reducir requerimientos de hardware.

Calificación

Tabla III. XV Calificación del indicador uso de memoria RAM

Rango de valores	Valoración Cuantitativa	Valoración Cualitativa	Forma Gráfica
Menor a 25 a 50	4	Excelente	
51 a 76	3	Bueno	
76 a 101	2	Regular	
Mayor a 101	1	Malo	

Fuente: Autor

El framework de servicios web Metro hace uso de 74.65MB de memoria RAM, con lo que en base a la tabla de calificación para el presente indicador, se le asigna una calificación de Bueno, su equivalencia grafica es de 3 estrellas. Mientras que el framework de servicios web Axis2 consume 115.14MB de memoria RAM, con lo cual se le asigna una calificación de Malo, su representación gráfica es de 1 estrellas.

I4: Uso de Procesador

Este indicador mide el porcentaje de uso del procesador por el framework de servicios web.

Al ejecutar las pruebas se obtiene resultados individuales, con los cuales se calcula la media o promedio, los errores relativos y porcentuales, desviación estándar e intervalos de confianza, con la finalidad de establecer datos estadísticos confiables que servirán a su vez en la evaluación del indicador. . Las capturas de las pruebas ejecutadas sobre el presente indicador se puede apreciar en el anexo2 y sus resultados en la Tabla III.XVI.

Tabla III. XVI Resultados de las pruebas para el uso de procesador

Prototipo	Prueba	x_0	(x)	$ x_0 - x $	$\frac{(x_0 - x)}{x_0}$	(\bar{x})	$(\bar{x}-x)$	$(\bar{x} - x)^2$
Metro	1	25.9	26	0.1	0.0039	263/10= 26.3	0.3	0.09
	2	25.8	26	0.2	0.0078		0.3	0.09
	3	27.6	28	0.4	0.0145		-1.7	1.369
	4	25.7	26	0.3	0.0117		0.3	0.09
	5	27.1	27	0.1	0.0037		-0.7	0.49
	6	22.2	22	0.2	0.009		4.3	18.49
	7	23.0	23	0	0.0		3.3	10.89

Prototipo	Prueba	x_0	(x)	$ x_0 - x $	$\frac{(x_0 - x)}{x_0}$	(\bar{x})	$(\bar{x}-x)$	$(\bar{x} - x)^2$
	8	29.0	29	0	0.0		-2.7	7.29
	9	27.7	28	0.3	0.0108		-1.7	2.89
	10	27.7	28	0.3	0.0108		-1.7	2.89
Axis2	1	37.0	37	0	0.0	382/10= 38.2	1.2	1.44
	2	37.3	37	0.3	0.008		1.2	1.44
	3	39.5	40	0.5	0.0127		-1.8	3.24
	4	39.0	39	0	0.0		-0.8	0.64
	5	39.5	40	0.5	0.0127		-1.8	3.24
	6	42.1	42	0.1	0.0024		-3.8	14.44
	7	34.7	35	0.3	0.0086		3.2	10.24
	8	37.1	37	0.1	0.0027		1.2	1.44
	9	37.4	37	0.4	0.0107		1.2	1.44
	10	38.4	38	0.4	0.0104		0.2	0.04

Fuente: Autor

Cálculo de errores para resultados de las pruebas en los prototipos

$$\epsilon_a = \frac{(\sum_{i=1}^n (|x_0 - x|))}{n}$$

$$\epsilon_r = \frac{(\sum_{i=1}^n \frac{(|x_0 - x|)}{x_0})}{n}$$

$$\epsilon_p = \frac{(\sum_{i=1}^n \frac{(|x_0 - x|)}{x_0})}{n} \times 100\%$$

Para las pruebas con Metro el error absoluto, relativo y relativo porcentual con respecto a los resultados son:

$$\varepsilon_{am} = \frac{(1.9)}{10} = 0.19\% \text{ CPU}$$

$$\varepsilon_{rm} = \frac{(0.0722)}{10} = 0.0072\% \text{ CPU}$$

$$\varepsilon_{rm} = \frac{(0.0722)}{10} \times 100\% = 0.72\%$$

Para la prueba con Axis2 el error absoluto, relativo y relativo porcentual con respecto a los resultados son:

$$\varepsilon_{aa} = \frac{(2.6)}{10} = 0.26\% \text{ CPU}$$

$$\varepsilon_{ra} = \frac{(0.0682)}{10} = 0.0068\% \text{ CPU}$$

$$\varepsilon_{ra} = \frac{(0.0682)}{10} \times 100\% = 0.68\%$$

Cálculo de la desviación estándar:

$$S = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}}$$

El valor de la desviación estándar para el framework de servicios web Metro es:

$$S_m = 44.579/9 = 4.953$$

El valor de la desviación estándar para el framework de servicios web Axis2 es:

$$S_a = 37.6/9 = 4.178$$

Cálculo del intervalo de confianza:

$$IC = \bar{x} \pm Z \left(\frac{S}{\sqrt{n-1}} \right)$$

Los intervalos de confianza para el framework de servicios web Metro son:

$$IC_m = 26.3 \pm 1.96 \left(\frac{4.953}{\sqrt{9}} \right)$$

$$IC_m = 26.3 \pm 3.24$$

Los intervalos de confianza para Axi2 son:

$$IC_a = 38.2 \pm 1.96 \left(\frac{4.178}{\sqrt{9}} \right)$$

$$IC_a = 38.2 \pm 2.73$$

De esta forma los resultados en %CPU de las pruebas realizadas a los prototipos se presentan en la Tabla III.XVII

Tabla III. XVII Resultados de la medición de uso del procesador

	Frecuencia	Promedio	Error Relativo	Porcentaje Error	S_{x̄}	IC
Metro	10	26.3	0.0072	0.72%	4.953	[23.06, 29.54]
Axis2	10	38.2	0.0068	0.000077%	4.178	[35.47, 40.93]

Fuente: Autor





Interpretación de Resultados

El framework de servicios web Metro presenta un menor uso del procesador al atender 385 peticiones con el 26.3% del CPU. A diferencia del framework de servicios web Axis2 que responde haciendo uso de 38.2% del CPU.

Debido a que el indicador es un recurso de hardware los rangos de valores son inversos.

Calificación

Tabla III. XVIII Calificación del indicador uso del procesador

Rango de valores	Valoración Cuantitativa	Valoración Cualitativa	Forma Gráfica
Menor a 25	4	Excelente	
25 a 50	3	Bueno	
51 a 75	2	Regular	
Mayor a 75	1	Malo	

Fuente: Autor

El framework de servicios web Metro consume un 26.3% del procesador con lo cual y en base a la tabla de calificación se le asigna una calificación de Bueno, su equivalencia gráfica es de 3 estrellas.

El framework de servicios web Axis2 al tener un consumo de 38.2% de consumo del procesador recibe una calificación de Bueno, su equivalencia gráfica de 3 estrellas.

3.3.2.3 Evaluación de resultados

Para evaluar el parámetro requerimientos de hardware se utiliza las calificaciones obtenidas por los indicadores en las pruebas realizadas y de las fórmulas (1, 2 y 3) previamente definidas en el método de evaluación.

Los resultados que se obtenga a partir del cálculo, se presentarán en forma numérica y gráfica, para después realizar la interpretación de los mismos

La calificación máxima del parámetro de comparación se establece mediante la suma de los valores máximos de calificación de cada indicador. Es así que:

$$C_{max} = \sum_{i=0}^n V_i = 4 + 4 = 8$$

La calificación numérica para el framework de servicios web Metro en el parámetro requerimientos de hardware, se calcula mediante la sumatoria de las calificaciones de los indicadores obtenidas en las pruebas. De esta forma:

$$C_m = \sum_{i=0}^n V_i = 3 + 3 = 6$$

La calificación numérica para el framework de servicios web Axis2 en el parámetro requerimientos de hardware, se calcula de igual forma a la del framework de servicios web Metro.

$$C_a = \sum_{i=0}^n V_i = 1 + 3 = 4$$

El nivel de cumplimiento del framework de servicios web Metro expresado en porcentajes se división de la calificación obtenida por este entre la calificación máxima del parámetro y todo esto multiplicado por el cien por ciento.

$$P_m = \left(\frac{C_m}{C_{max}} \right) \times 100\% = \left(\frac{6}{8} \right) \times 100\% = 75\%$$

De igual forma el nivel de cumplimiento del framework de servicios web Axis2 expresado en porcentajes se división de la calificación obtenida por este entre la calificación máxima del parámetro y todo esto multiplicado por el cien por ciento.

$$P_a = \left(\frac{C_a}{C_{max}} \right) \times 100\% = \left(\frac{4}{8} \right) \times 100\% = 50\%$$

Los resultados del cumplimiento de los frameworks de servicios web se expresan en el Figura III.3.

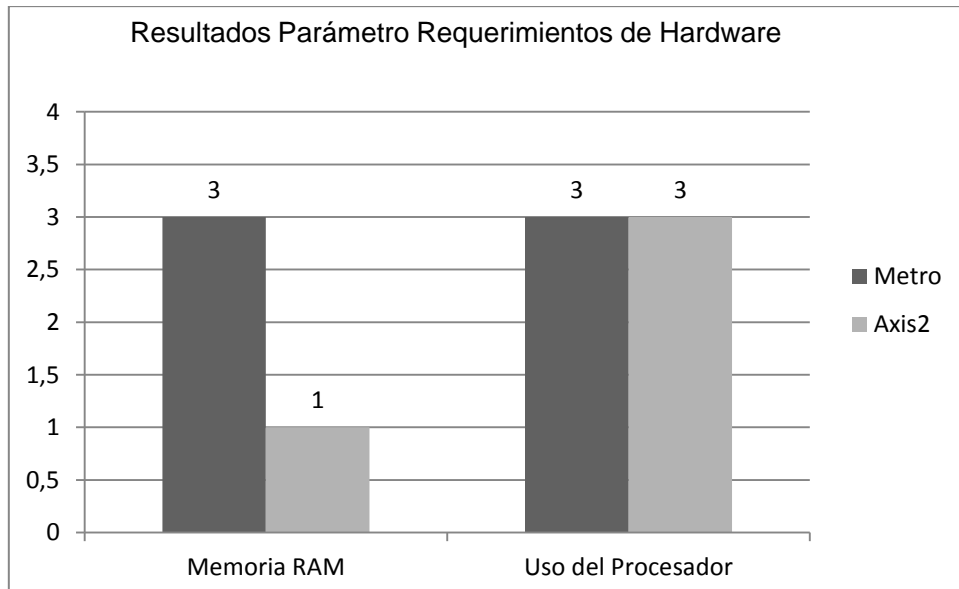


Figura III. 3 Resultados del parámetro requerimientos de hardware

Fuente: Autor

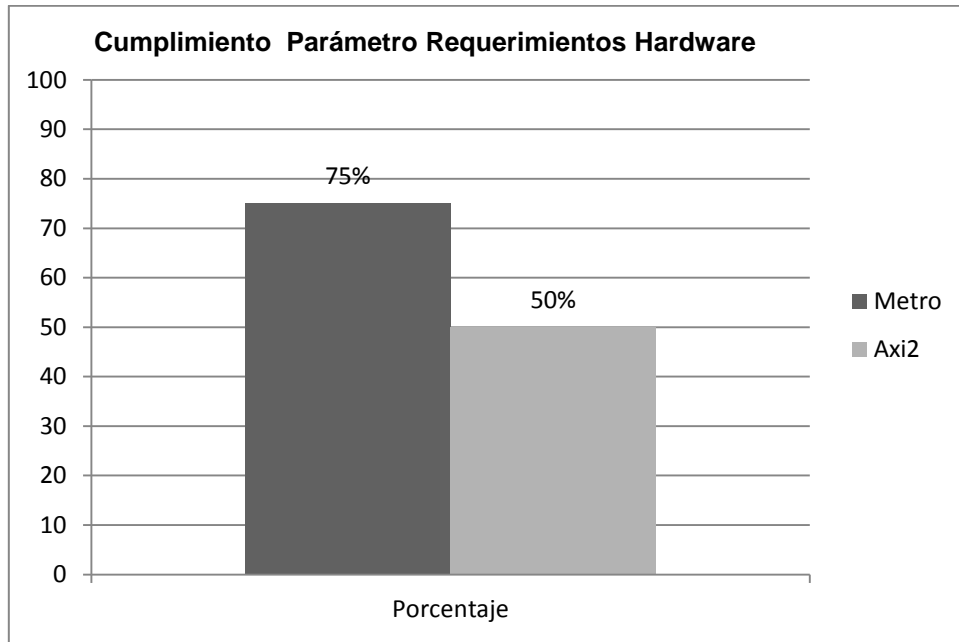


Figura III. 4 Porcentajes cumplimiento requerimientos de hardware

Fuente: Autor

3.3.2.4 Interpretación de resultados

En base a la Tabla III.II que permite establecer el equivalente cualitativo de acuerdo a porcentajes, se valorara los resultados obtenidos por cada framework de servicios web.

Los resultados obtenidos en el parámetro requerimientos de hardware revelan que el framework de servicios web Metro posee un nivel de cumplimiento del 75% de los indicadores establecidos, lo que equivale a Buena. Mientras que el framework de servicios web Axis2 cumple con el 50% de los indicadores establecidos, lo que equivale a Regular.

3.3.3 Simplicidad

Mediante este parámetro se demuestra el nivel de sencillez que posee el framework de servicios web a la hora de instruirse e implementarlo para el desarrollo de aplicaciones que consuman servicios web WCF.

Tabla III. XIX Indicadores del parámetro simplicidad

Indicador	Descripción
I5. Disponibilidad de Información	Información disponible acerca framework
I6. Facilidad de Desarrollo	Facilidad de desarrollo para la integración de la aplicación.

Fuente: Autor

I5. Disponibilidad de Información

Representa la disponibilidad de información que existe acerca del framework de servicios web, tomando en cuenta a los sitios oficiales de los proyectos, foros oficiales, tutoriales, código fuente.

Las fuentes a evaluar son los siguientes:

- Manuales
- Tutoriales/Ejemplos
- Artículos
- Foros

El cumplimiento de todas estas fuentes tendrá una valoración de Excelente,

Framework de servicios web Metro

Manuales.- La comunidad Glassfish a través del proyecto Metro proveen manuales de referencia tales como Getting Started, Users Guide, entre otros, en total suman 9.

Artículos.- Además de la información del sitio oficial, el framework de servicios web Metro posee muchos artículos escritos por otras empresas desarrolladoras de software, tales como IBM, Microsoft y Code Project.

Foros.- El framework de servicios web provee su propio foro para dudas, comentarios y preguntas de los usuarios llamado java.net Metro and JAXB.

Tutoriales/Ejemplos.- El sitio oficial provee 10 tutoriales incluyendo fragmentos de código, para desplegar y probar.

Framework de servicios web Axis2

Manuales.- El sitio oficial del framework de servicios web Axis2 provee varios manuales de implementación, guía entre otros, que en conjunto suman 8.

Artículos.- El sitio oficial, el framework de servicios web AXIS2 posee 25 artículos propios y otros escritos por empresas desarrolladoras de software, tales como IBM y Code Project.

Foros.- El framework de servicios web Axis2 no provee un foro oficial, pero existen otros en la web muy relacionados y con ello se pueden resolver las dudas.

Tutoriales/Ejemplos.- El sitio oficial provee 5 tutoriales incluyendo fragmentos de código, para desplegar y probar.

Tabla III. XX Valoración para el indicador disponibilidad de información

Rango	Valor
Mayor a 8	3
Entre 5 y 8	2
De 1 a 4	1
Ninguno	0

Fuente: Autor

Resultados de las fuentes de disponibilidad de información





Tabla III. XXI Fuentes de disponibilidad de información

	Manuales	Artículos	Foros	Tutoriales/Ejemplos	Total
Metro	3	2	3	3	11
Axis2	2	3	1	2	8

Fuente: Autor

Calificación

Tabla III. XXII Calificación del indicador disponibilidad de información

Rango de valores	Valoración Cuantitativa	Valoración Cualitativa	Forma Gráfica
10 a 14	4	Excelente	
5 a 9	3	Bueno	
1 a 4	2	Regular	
Menor a 1	1	Malo	

Fuente: Autor

El framework de servicios web Metro posee una disponibilidad equivalente a Excelente, mientras que para el framework de servicios web Axis2 la disponibilidad de información equivale a Bueno. Por tanto su representación es de 4 y 3 estrellas respectivamente.

I6. Facilidad de Desarrollo

Este indicador determina el grado de sencillez de desarrollo con el framework de servicios web se integra al aplicación web cliente.

Las fuentes a tomar en cuenta son:

Líneas de código.- Representa el número de líneas de código que empleadas para interactuar con el cliente generado por el framework de servicios web.

Número de archivos generados.-Representa el número de archivos generados por el framework de servicios web, pudiendo ser clases java, XML u otros.

Mediante los prototipos realizados para cada framework de servicios web, se pudo observar la cantidad de archivos generados para la creación del cliente y también se contabilizo las líneas de código que permiten interactuar con la aplicación web. Estos resultados se ilustran en la Tabla III.XXIII.

Tabla III. XXIII Resultados de la medición para el indicador facilidad de desarrollo

Fuente Framework	Líneas de código	Archivos generados
Metro	18	8
Axis2	40	10

Fuente: Autor

Calificación

Las siguientes tablas presentan el sistema de valoración para cada uno de las fuentes, los valores de mayor puntaje indican mayor simplicidad:

Tabla III. XXIV Valoración para la fuente líneas de código

Rango de valores	Valor
Menos de 10	3
De 10 a 20	2
Entre 21 y 31	1
Mayor a 31	0

Fuente: Autor

Tabla III. XXV Valoración para la fuente archivos generados

Rango de Valores	Valor
Menos de 5	3
De 5 a 10	2
Entre 11 y 16	1
Mayor a 16	0

Fuente: Autor





Resultados de las fuentes de facilidad de desarrollo

Tabla III. XXVI Resultados de las fuentes facilidad de desarrollo

	Líneas de Código	Archivos generados	Total
Metro	2	2	4
Axis2	0	2	2

Fuente: Autor

Tabla III. XXVII Valoración para el indicador facilidad de desarrollo

Rango de valores	Valoración cuantitativa	Valoración Cualitativa	Forma Gráfica.
Mayor a 5	4	Excelente	
De 4 a 5	3	Buena	
De 2 a 3	2	Regular	
De 0 a 1	1	Malo	

Fuente: Autor

Haciendo uso de la Tabla III.XXVII se asigna al framework de servicios web Metro la calificación de Buena con una equivalencia gráfica de 3 estrellas, mientras que para el framework de servicios web Axis2 se le asigna una calificación de Regular, cuya representación gráfica es de 2 estrellas.

3.3.3.3 Evaluación de resultados

Para evaluar el parámetro simplicidad se utiliza las calificaciones obtenidas por los indicadores y las fórmulas previamente definidas en el método de evaluación.

Los resultados que se obtenga a partir del cálculo se presentarán en forma numérica y gráfica, para después realizar la interpretación de los mismos

La calificación máxima del parámetro de comparación se establece mediante la suma de los valores máximos de calificación de cada indicador. Es así que:

$$C_{max} = \sum_{i=0}^n V_i = 4 + 4 = 8$$

La calificación numérica para el framework de servicios web Metro en el parámetro simplicidad, se calcula mediante la sumatoria de las calificaciones de los indicadores obtenidas en las pruebas. De esta forma:

$$C_m = \sum_{i=0}^n V_i = 4 + 3 = 7$$

La calificación numérica para el framework de servicios web Axis2 en el parámetro simplicidad, se calcula de igual forma a la del framework de servicios web Metro.

$$C_a = \sum_{i=0}^n V_i = 3 + 2 = 5$$

El nivel de cumplimiento en simplicidad del framework de servicios web Metro expresado en porcentajes es igual a la división de la calificación obtenida por este entre la calificación máxima del parámetro y todo esto multiplicado por el cien por ciento.

$$P_m = \left(\frac{C_m}{C_{max}} \right) \times 100\% = \left(\frac{7}{8} \right) \times 100\% = 87.5\%$$

De igual forma el nivel de cumplimiento en simplicidad del framework de servicios web Axis2 expresado en porcentajes es igual a la división de la calificación obtenida por este entre la calificación máxima del parámetro y todo esto multiplicado por el cien por ciento.

$$P_a = \left(\frac{C_a}{C_{max}} \right) \times 100\% = \left(\frac{5}{8} \right) \times 100\% = 62.5\%$$

Los resultados del cumplimiento de los frameworks de servicios web se expresan en el Figura III.5.

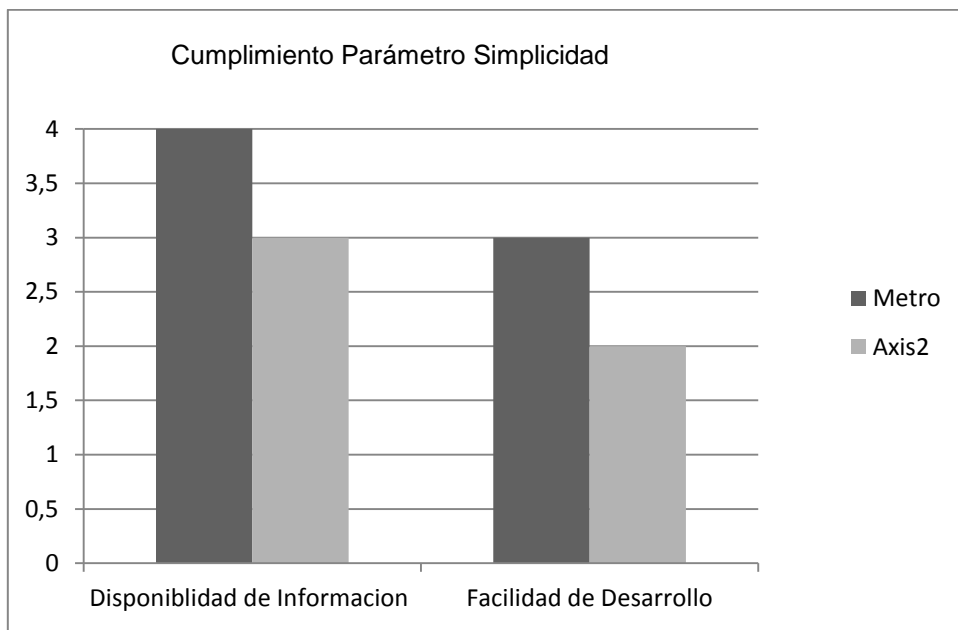


Figura III. 5 Resultados del parámetro simplicidad

Fuente: Autor

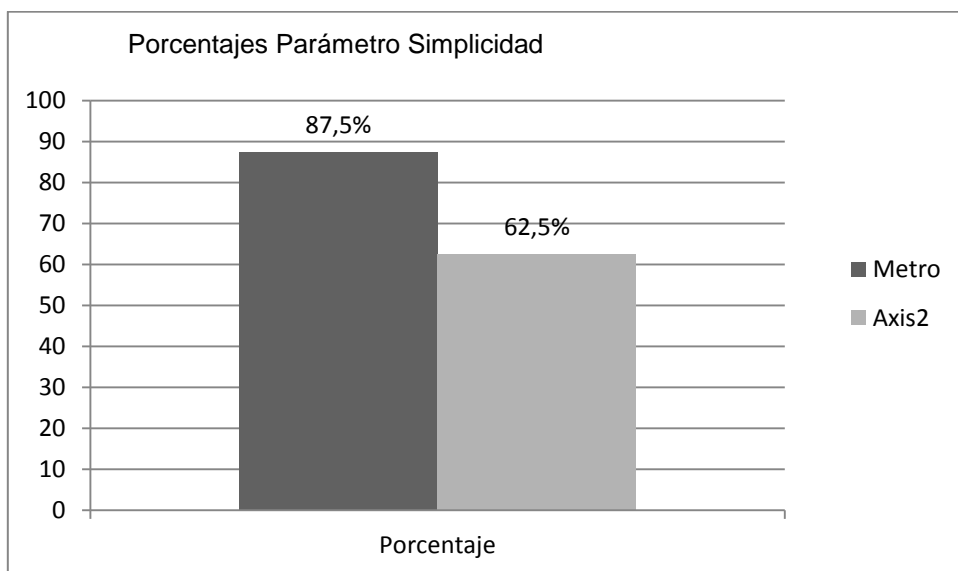


Figura III. 6 Porcentajes cumplimiento parámetro simplicidad

Fuente: Autor

3.3.3.4 Interpretación de resultados

En base a la Tabla III.II que permite establecer el equivalente cualitativo de acuerdo a porcentajes, se valorara los resultados obtenidos por cada framework de servicios web.

Los resultados obtenidos en el parámetro complejidad revelan que el framework de servicios web Metro posee un 87.5% de nivel de cumplimiento de complejidad para el desarrollo de aplicaciones, lo que equivale a Excelente. Mientras que el framework de servicios web Axis2 posee un 62.5% de nivel de cumplimiento de complejidad a la hora de desarrollar aplicaciones web, lo que equivale a Bueno.

3.4 Demostración de la Hipótesis

Las calificaciones individuales para los indicadores de los parámetros de comparación se detallan en la Tabla III.XXVIII.

Tabla III. XXVIII Resumen calificación de indicador en cada parámetro

Parámetro e Indicador		Metro	Axis2	Calificación Máxima	%Metro	%Axis2
Rendimiento	I1. Throughput	4	2	4	87.5%	50%
	I2. Tiempo de Respuesta	3	2	4		
Requerimientos de hardware	I3. Memoria RAM	3	1	4	75%	50%
	I4. Uso de Procesador	3	3	4		
Simplicidad	I5. Disponibilidad de Información	4	3	4	87.5%	62.5%
	I6. Facilidad de Desarrollo	3	2	4		

Fuente: Autor

Las calificaciones en porcentajes de cada parámetro de comparación empleado en la presente investigación se observa en la Figura III.7.

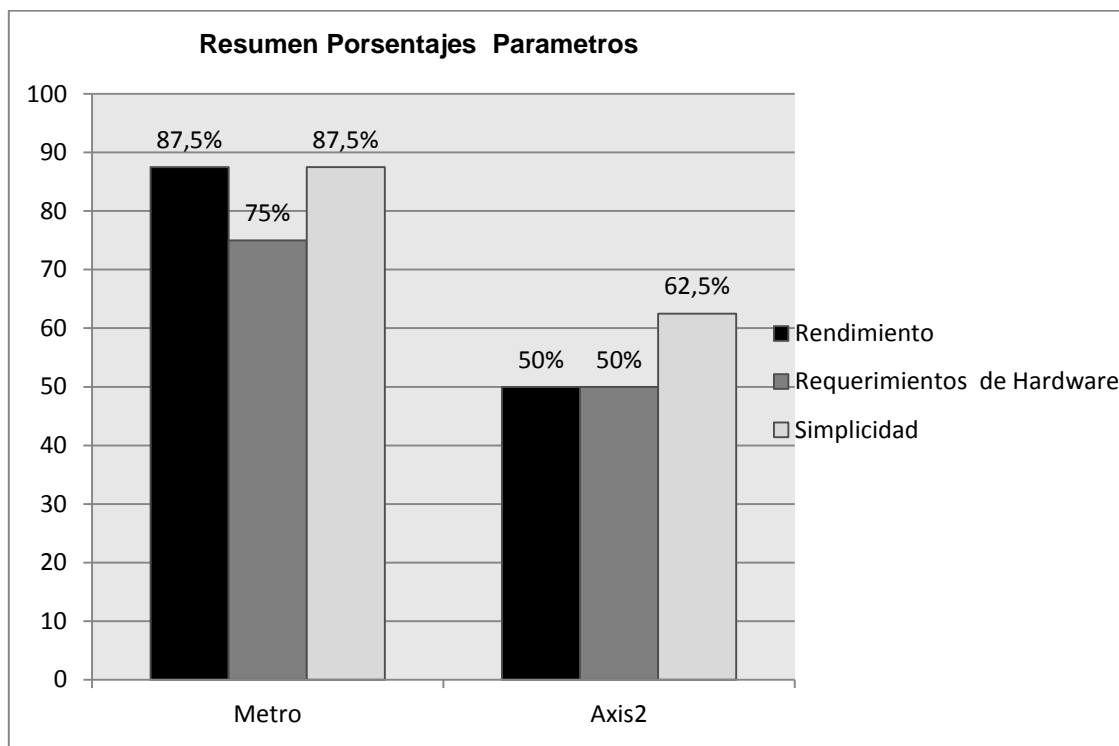


Figura III. 7 Resumen porcentajes parámetros

Fuente: Autor

En base a resultados obtenidos y a la Figura III.7 se interpreta:

- Metro provee una ventaja en rendimiento del 37.5% sobre Axis2. Debido a que este cumple con el 87.5% de las expectativas en los indicadores comparación, frente al 50% de cumplimiento de Axis2
- Metro requiere una menor cantidad de recursos hardware cumpliendo con el 75% en comparación a Axis2 que cumple con el 50% de las expectativas en los indicadores asignados.

- Metro al cumplir con el 87.5% de los indicadores asignados pose una mayor simplicidad de desarrollo de software en comparación a 62.5% de cumplimiento de Axis2

Considerando los parámetros establecidos en el análisis, se presenta en la Tabla III.XXIX el resumen de la calificación obtenidos por cada framework de servicios web.

Tabla III. XXIX Resumen de calificaciones de los frameworks en cada parámetro

	Rendimiento	Requerimientos de Hardware	Simplicidad	Total	%
Metro	4	3	4	11	91.67%
Axis2	2	2	3	7	58.33%
Calificación Máxima	4	4	4	12	100%

Fuente: Autor

Las calificaciones de los framework de servicios web en porcentajes se ilustran en la Figura III.8.

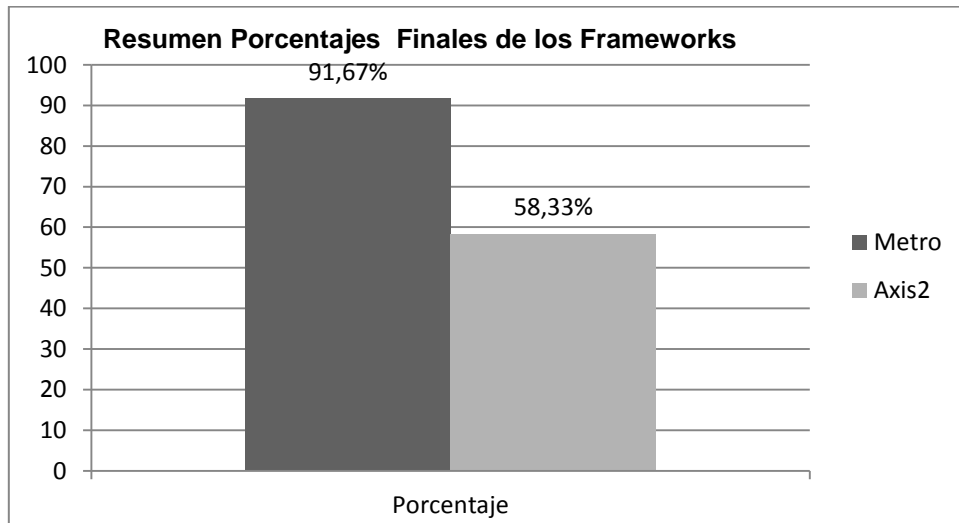


Figura III. 8 Resumen porcentajes finales de los frameworks

Fuente: Autor

3.5 Comprobación de la Hipótesis

En base a los resultados del análisis realizado, su interpretación y haciendo uso de la estadística descriptiva se comprueba la hipótesis planteada para la presente tesis, mediante la cual se puede afirmar que la tecnología Metro mejora el desempeño de las aplicaciones que consumen servicios web WCF cumpliendo con el 91.67% de los parámetros establecidos para el análisis en comparación al 58.33% de cumplimiento del Axis2.

CAPÍTULO IV

SITIO WEB DE INFORMACIÓN PARA ESTUDIANTES DE LA ESPOCH.

4.1 Introducción

En el presente capítulo se detallará el desarrollo del sitio de información para estudiantes de la ESPOCH, haciendo uso de la metodología de desarrollo de software ágil XP (Programación Extrema). La misma que ha sido seleccionada debido a que es adaptable a las necesidades que surgen, posee un enfoque grupal, permite ver al proyecto desde un punto de vista del usuario y promueve valores que todo programador XP debe tener tal como la simplicidad, la comunicación, y la realimentación

La metodología de desarrollo de software XP cuenta con cuatro fases, que son: Planificación, Diseño, Codificación y Pruebas. Estas serán descritas durante el presente capítulo. En la Figura IV.1 se puede apreciar la relación de las fases de la metodología.

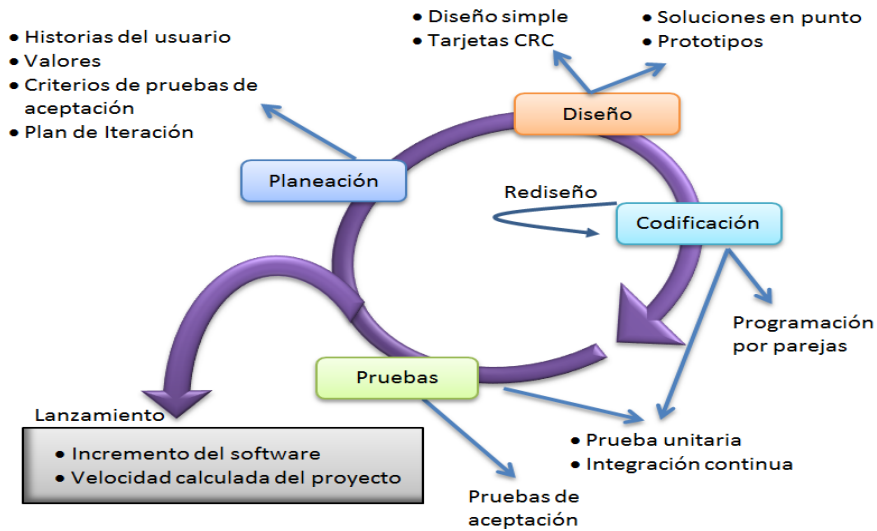


Figura IV. 1 Fases de metodología XP

Fuente: CODEJBOS [4].

4.2 Fase I. Planeación

En esta fase se empieza a interactuar con el cliente y el resto del grupo de desarrollo para descubrir los requerimientos del sistema. En este punto se identifica el número y tamaño de interacciones, así como la prioridad de cada historia de usuario y se llega a un acuerdo con el cliente en el cronograma de entrega.

4.2.1 Descripción del sistema

El sitio web de información para estudiantes de la ESPOCH, permite visualizar la información personal y académica obtenida por los mismos en las diferentes unidades académicas de la institución. Entre sus módulos están información personal y de

discapacidad del estudiante, horarios de clase y exámenes, notas académicas y realizar la pre-matrícula.

El sistema posee una arquitectura distribuida, interoperable, la cual está constituida de dos módulos, cada uno de estos con sus propias capas y tecnologías de desarrollo. Los módulos del sistema son: el módulo servidor BackEnd que contiene los servicios web WCF de información estudiantil y el módulo FrontEnd que es una aplicación web cliente JAVA que consume los servicios web y los presenta al usuario final.

4.2.2 Definición del flujo del proceso de visualización de información estudiantil

Establecer el flujo de procesos permite evitar pasar por alto procesos necesarios y que son fundamentales para el sitio web de información estudiantil. Estos procesos fueron definidos en conjunto con el administrador del sistema OASIS y representados gráficamente mediante diagramas de procesos. El formato para los diagramas fue establecido por la DTIC de la ESPOCH. Este está constituido por los siguientes componentes:

Actividad.- Representa el requerimiento o necesidad que se pretende realizar en cada fase.

Diagrama.- Indica principalmente el diagrama de flujo de la actividad excluyendo los proceso de entrada y salida.

IN.- Representa los procesos de entrada de datos.

OUT.- Representa la emisión de información

Responsable.- Es el actor responsable de realizar la actividad.

Observación.- Comentarios y/o aclaraciones sobre la actividad o responsable.

A manera de ejemplo en la Figura IV.2 se ilustra el diagrama de procesos para la autenticación e inicio en el sistema, los demás diagramas de procesos se sitúa en el Anexo3.

Proceso de autenticación

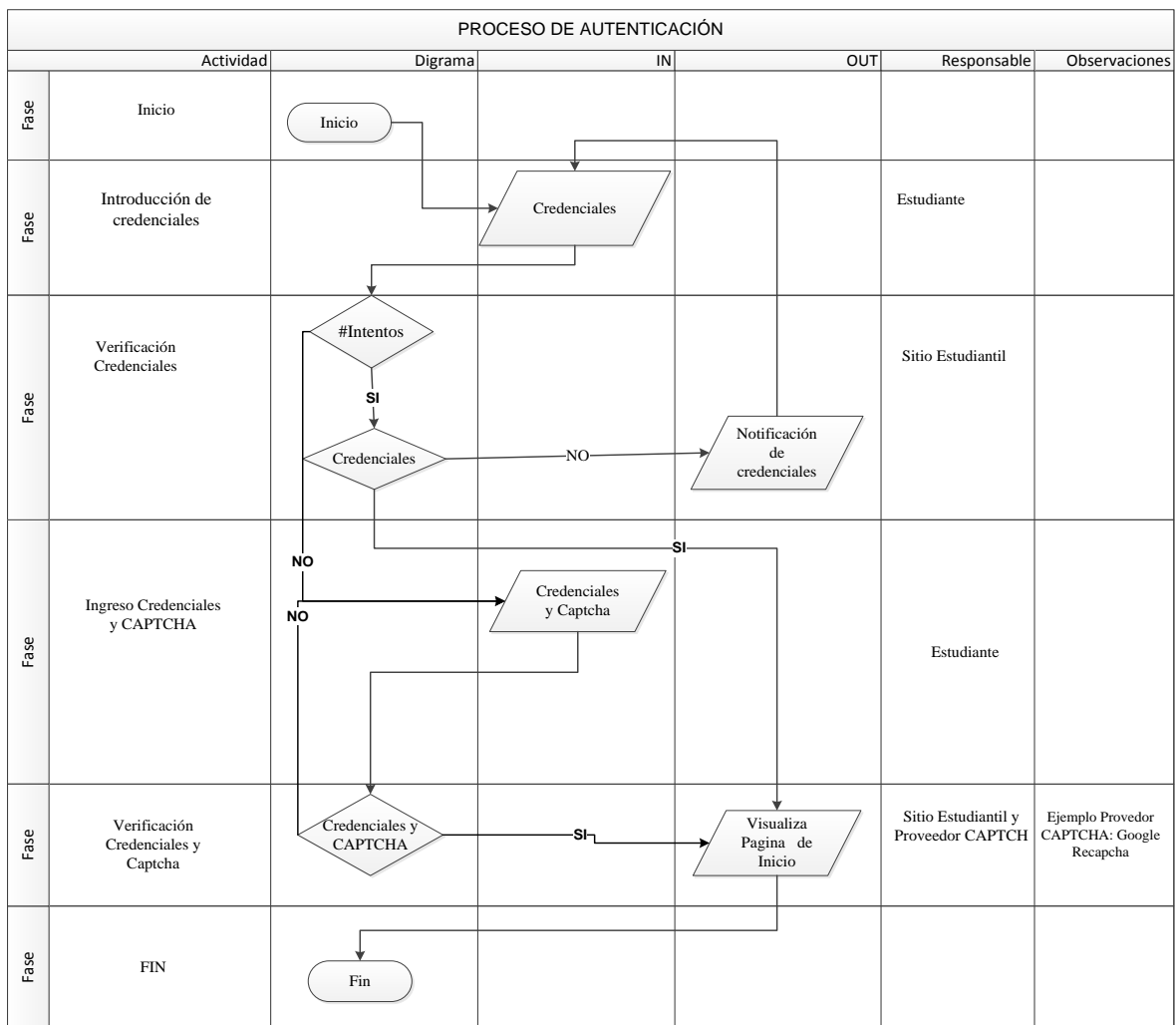


Figura IV. 2 Proceso de autenticación

Fuente: Autor

4.2.2 Especificación de Requerimientos

Se enlista los requerimientos de comportamiento funcional y no funcional del sistema que a su vez permitirán conocer los componentes que integran el sitio de información para estudiantes de la ESPOCH.

Requerimientos funcionales

Se hace uso del prefijo “Req” que representa el requerimiento, esto con la finalidad de establecer un identificador y poder referenciar a los mismos. Los requerimientos establecidos en concordancia con el administrador del OASIS son:

Req1. El sistema permitirá autenticar a estudiantes previamente registrados.

Req2. El sistema permitirá visualizar el perfil personal del estudiante.

Req3. El sistema permitirá actualizar la información de contacto del estudiante.

Req4. El sistema permitirá visualizar el perfil de discapacidad del estudiante.

Req5. El sistema permitirá visualizar los datos de las inscripciones realizadas por el estudiante.

Req6. El sistema permitirá visualizar la lista de períodos optados por el estudiante por unidad académica.

Req7. El sistema permitirá visualizar las asignaturas optadas al seleccionar un período determinado por unidad académica.

Req8. El sistema permitirá visualizar los títulos de Pregrado.

Req9. El sistema permitirá visualizar las notas desglosadas con su equivalencia de cada asignatura optada en determinado periodo académico.

Req10. El sistema permitirá visualizar las notas de los parciales de cada asignatura optada en determinado periodo académico.

Req11. El sistema permitirá visualizar el horario definido para el período actual del estudiante.

Req12. El sistema permitirá visualizar una lista de horarios disponibles por nivel y paralelo de cada Unidad académica optada por el estudiante.

Req13. El sistema permitirá visualizar una lista de horarios por materia de cada Unidad Académica.

Req14. El sistema permitirá visualizar el horario de exámenes definido según el período académico vigente.

Req15. El sistema permitirá inscribirse en los centros de apoyo en caso no pertenecer a estos.

Req16. El sistema permitirá visualizar noticias al inicio de este.

Req17. El sistema permitirá solicitar una matrícula de materias obligatorias y optativas que el estudiante haya seleccionado.

Requerimientos No funcionales

Escalabilidad.- El sistema poseerá una arquitectura distribuida en n-capas.

Interoperabilidad.- El sistema tendrá la capacidad de exponer y consumir servicios web.

Usabilidad.- El sistema estará constituido por una interfaz fácil y navegable.

Seguridad.- El sistema tendrá autenticación y el consumo como exposición de servicios web será encriptado.

Rendimiento.- El sistema poseerá un tiempo de respuesta inferior al aceptable.

4.2.3 Definición de usuarios

Es la persona o agente que hace uso del sitio web de información para estudiantes de las ESPOCH, así como el rol que cumple en el sistema. Los usuarios dependerán de los roles que se definan.

4.2.4 Definición de roles

El sistema cuenta únicamente con un único tipo de usuario:

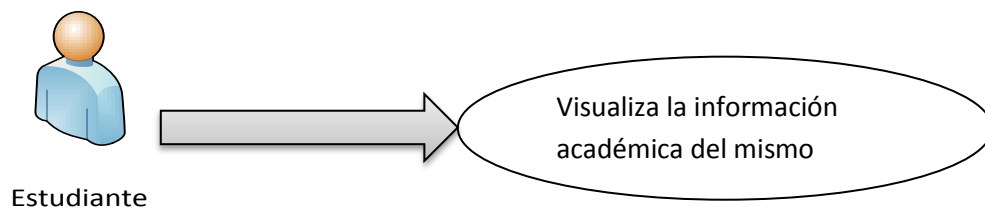


Figura IV. 3 Roles para el sitio de información para estudiantes

4.2.5 Historias de Usuario

Representa una descripción específica del comportamiento funcional del sistema. Se realiza uno por cada requerimiento funcional del sitio web, se emplean para reemplazar una gran cantidad de documentación y el tratamiento de las mismas es dinámico y flexible.

Las historias de usuario constan de los siguientes campos:

Número.-Valor Indica el identificador de la historia de usuario.

Nombre de la historia.- Nombre de la historia de usuario.

Usuario.- Nombre del usuario al que pertenece la historia.

Iteración.- Interacción en la que se va a entregar.

Prioridad de negocio.- Nivel de importancia para el usuario.

Programador responsable.- Programador encargado de programar la historia.

Riesgo de desarrollo.- Nivel de riesgo a la hora de desarrollar la historia.

La tabla IV.I ilustra la historia de usuario para el requerimiento 1 “**Req1**”, las historias faltantes se encuentran en el anexo4.

Tabla IV. I Historia de Usuario 1

HISTORIA DE USUARIO	
Numero: 1	Usuario: Estudiante
Nombre Historia: Autenticación del estudiante	Iteración Asignada: 1
Prioridad en Negocio: Alto	Riesgo en Desarrollo: Alto
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante procede a autenticarse, ingresando sus credenciales para iniciar en el sistema.	
Observaciones: Requerimiento esencial.	

4.2.6 Plan de iteraciones

Este plan permite medir la velocidad de avance del sistema, solo puede realizar por tiempo o alcance. Para el presente proyecto se ha decidido emplear el método por fechas, con lo cual cada iteración posee un tiempo determinado por días y semanas.

Iteración 1

En esta iteración se propone realizar la autenticación de usuarios previamente registrados y la visualización del perfil personal

Tabla IV. II Iteración 1

N°	Historias de usuario	Duración en días
1	Autenticar a estudiantes previamente registrados	7
2	Perfil personal del estudiante	3
	Total días	10
	Total semanas	2

Fuente: Autor

Iteración 2

En la presente iteración se propone actualizar los datos de contacto de perfil, visualizar el perfil de discapacidad y desplegar la lista de inscripciones realizadas en las unidades académicas.

Tabla IV. III Iteración 2

N°	Historias de Usuario	Duración en días
3	Actualizar datos contacto perfil	5
4	Perfil de discapacidad del estudiante	6
5	Inscripciones del estudiante.	4
	Total días	15
	Total semanas	3

Fuente: Autor

Iteración 3

En esta iteración se realizará el proceso de visualización de la lista de períodos optados por unidad académica, las asignaturas optadas en el periodo seleccionado y los títulos de pre-grado.

Tabla IV. IV Iteración 3

N°	Historias de Usuario	Duración en días
6	Periodos optados en una unidad académica.	5
7	Asignaturas optadas en un periodo optado.	7
8	Títulos pre-grado	3
	Total días	15
	Total semanas	3

Fuente: Autor

Iteración 4

En esta iteración pretende visualizar notas académicas totales y parciales de cada asignatura en una unidad academia seleccionada

Tabla IV. V Iteración 4

N°	Historias de Usuario	Duración en días
9	Notas de cada asignatura	8
10	Notas parciales	7
	Total días	15
	Total semanas	3

Fuente: Autor

Iteración 5

En esta iteración pretende visualizar los horarios de clase del estudiante en el período vigente, horarios por paralelo y materia.

Tabla IV. VI Iteración 5

N°	Historias de Usuario	Duración en días
11	Horario de clase del estudiante	5
12	Horarios de clase por paralelo	5
13	Horarios de clase por materia	5
	Total días	15
	Total semanas	3

Fuente: Autor

Iteración 6

En esta iteración pretende visualizar los horarios de exámenes, inscribirse en los centros de apoyo e informarse mediante el centro de noticias estudiantiles.

Tabla IV. VII Iteración 6

N°	Historias de Usuario	Duración en días
14	Horarios de exámenes	3
15	Inscribirse centros de apoyo	2
16	Noticias estudiantiles	10
	Total días	15
	Total semanas	3

Fuente: Autor

Iteración 7

En esta iteración pretende visualizar los horarios de clase del estudiante en el período vigente, horarios por paralelo y materia.

Tabla IV. VIII Iteración 7

N°	Historias de Usuario	Duración en días
7	Pre-matricula en materias	20
	Total días	20
	Total semanas	4

Fuente: Autor

4.3 Fase II. Diseño de Software

Esta etapa se desarrolla en concordancia al ciclo de desarrollo de software, dando así una flexibilidad para posibles cambios.

Los elementos más importantes del diseño son el diagrama de base de datos, diccionario de datos, diagrama de componentes, diagrama de despliegue y las interfaces.

4.3.1 Diseño de la Base de Datos

El diseño de base de datos permite tener una base de la forma de almacenamiento de los datos provenientes de la aplicación. El sitio de información para estudiantes hizo uso de tablas de dos bases de datos diseñadas por la DTIC de la ESPOCH, como son la base de datos OASIS y la segunda es la base de datos centralizada. Las tablas utilizadas en el sistema se pueden apreciar en las figuras Figura IV. 4 y Figura IV. 5.

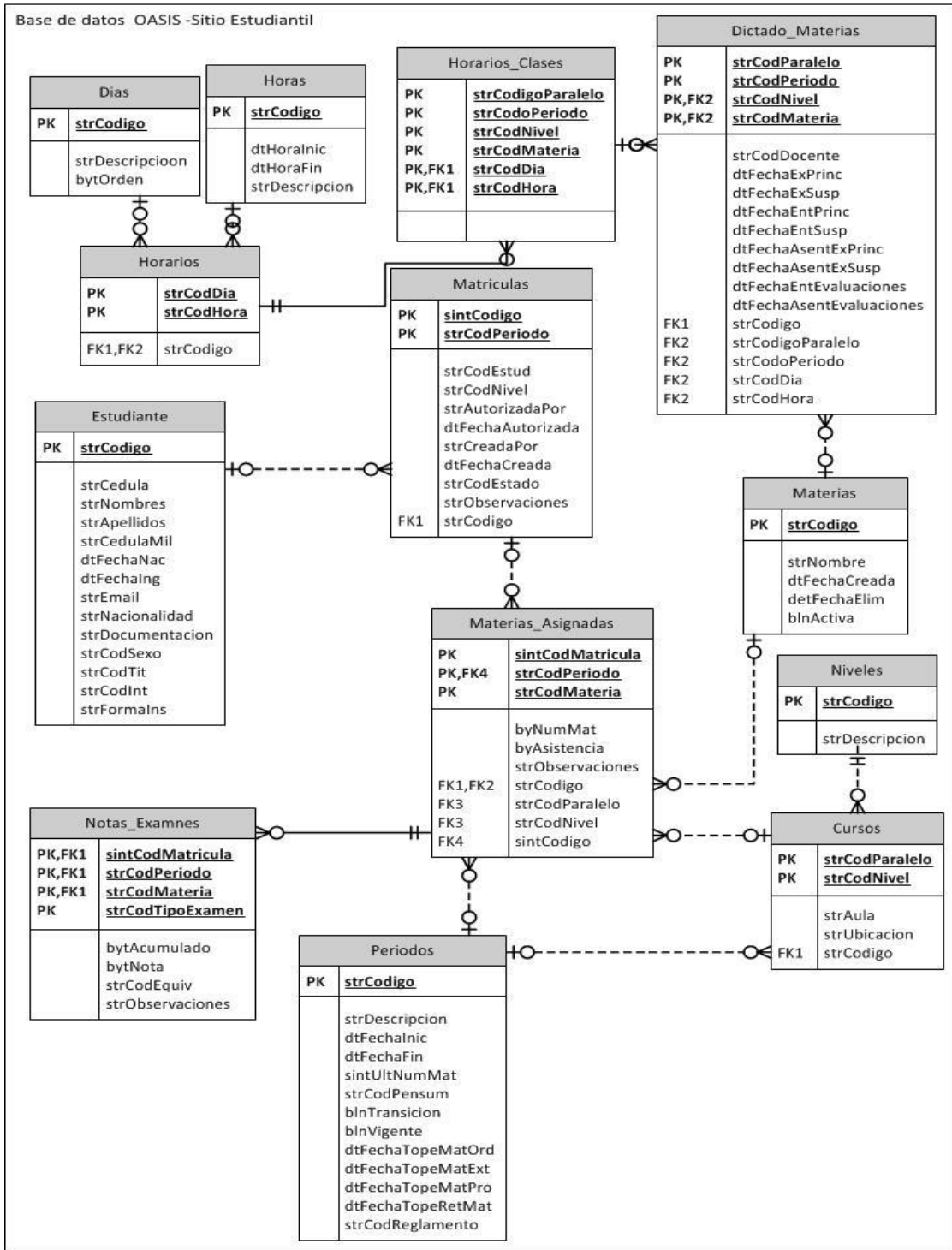


Figura IV. 4 Diagrama de base de datos OASIS-Sitio Estudiantil

Fuente: DTIC-ESPOCH

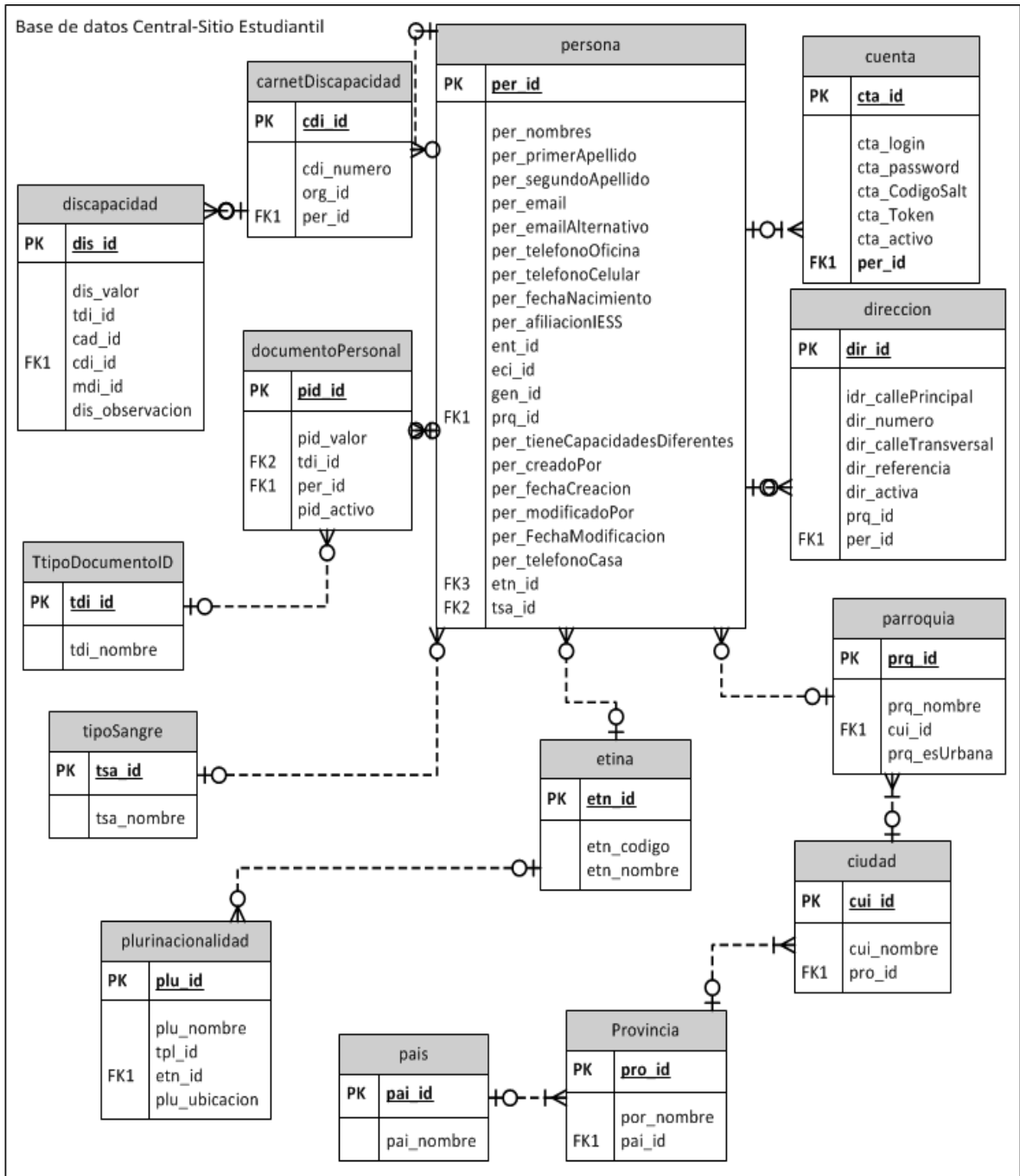


Figura IV. 5 Diagrama la de base de datos Central-Sitio Estudiantil

Fuente: DTIC-ESPOCH

4.3.3 Diccionario de datos

Es una agrupación de metadatos que define la estructura lógica de almacenamiento de la base de dato. Se realiza la descripción de cada tabla de las dos bases de datos empleadas en el desarrollo del sistema, los campos a tener en cuenta son:

Atributo.- Campo de la base de una tabla de la base de datos

Descripción.- significado del campo para la base de datos.

Tipo.- Tipo de datos para la base de datos.

PK.- Si el campo en cuestión es clave primaria.

FK.- Si el campo en cuestión es clave foránea.

NULL.- Admite o no valores nulos.

Esta información permite determinar la forma de acceso desde la aplicación. En la Tabla IV. IX se ilustra un ejemplo de la descripción del meta data en cada tabla, las demás descripciones se puede apreciar en el anexo5.

Tabla periodos

Tabla IV. IX Descripción tabla Periodos Base de datos OASIS -Sitio Estudiantil

Atributo	Descripción	Tipo	PK	NULL	FK
strCodigo	Identifica Periodos	varchar(10)	Si	NOT NULL	
strDescripcion	Descripción de Periodos	varchar(10)			

Atributo	Descripción	Tipo	PK	NULL	FK
dtFechaInic	Fecha de inicio del Periodos	datetime			
dtFechaFin	Fecha de fin de Periodos				
strCodPensum	Código del pensum Periodos	varchar(10)			
blnTransicion	Periodo en transición	binary(10)			
blnVigente	Si el periodo es vigente	binary(10)			
dtFechaTopeMatOrd	Fecha tope para matricularse ordinariamente	datetime			
dtFechaTopeMatExt	Fecha tope para matricularse extra ordinariamente	datetime			
dtFechaTopeMatPro	Fecha tope matricula provisional	datetime			
dtFechaTopeRetMat	Fecha tope para retirarse	datetime			
strCodReglamento	Código del reglamento del Periodo	varchar(10)			

4.3.4 Diseño de arquitectura

La arquitectura de una aplicación es la estructura base para su desarrollo e implementación, es así que se la misma se divide en dos, que son: arquitectura lógica y arquitectura física.

La arquitectura lógica

Hace referencia a un modelo de componentes para el desarrollo de la aplicación. De esta forma en este tipo de arquitecturas es muy importante establecer un diagrama de

componentes que permitan tener una visión sobre la organización y las dependencias del sistema.

En la Figura IV. 6 se puede apreciar el diagrama de componentes, el mismo que está compuesto de dos soluciones un BackEnd y un FrontEnd.

BackEnd

Modulo servidor o servicios web, está constituido por las siguientes capas:

Base de datos.- Componente de almacenamiento masivo, que para el presente proyecto será parte de la base de datos OASIS en el gestor Microsoft SQL Server 2008.

Acceso a datos.- Este componente se en carga de relacionar la aplicación con la base de datos. Para facilitar el desarrollo se hecho uso de Entity Framework que provee de un mapeo de objetos a tablas de la base de datos, con lo cual se ofrece una serie de métodos y clases para la manipulación de datos.

Servicios Web Estudiantil.- Contiene a los servicios web de información estudiantil desarrollados en .NET WCF haciendo uso de la capa de acceso a datos. Estos servicios web se los exponen con una seguridad en encriptación y autenticación personalizada mediante el uso de un certificado digital, dando así una forma segura de comunicación entre el cliente y servidor.

FrontEnd

Modulo cliente o aplicación web, está constituido de las siguientes capas:

Base de datos Centralizada.- Base de datos que pretende contener la información de todas las personas de las ESPCOH y como la información del estudiante.

Componente centralizado.- Proporciona todos los métodos y calses de acceso a los datos de la base de datos centralizada.

Proxy.- Esta capa provee de métodos para el acceso a la información académica del estudiante. Es la capa más importante del sistema, ya que en esta se realiza el consumo de los servicios web mediante el framework de servicios web METRO. Además aquí se establece las configuraciones necesarias para lograr una comunicación segura con el servidor.

Presentación.-Esta capa fue la encargada de recibir las peticiones del usuario, invocar a la lógica de negocio del consumo de servicios web en la capa proxy y devolver resultados al mismo. Esta capa fue desarrollada con el framework JSF y Primefaces.

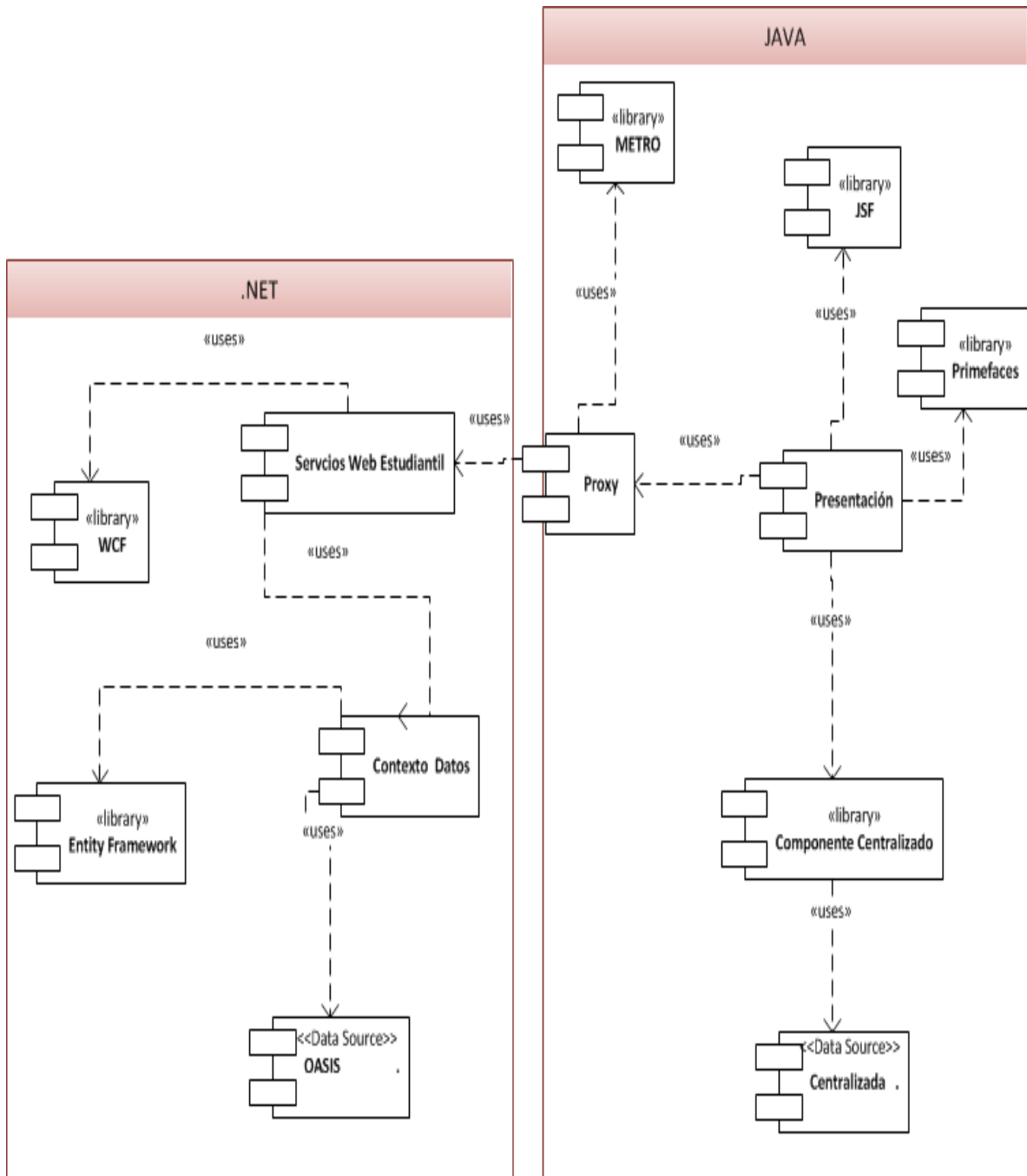


Figura IV. 6 Diagrama de componentes del Sitio de Información para Estudiantes

Fuente: Autor

La arquitectura física o de implementación

Hace referencia a un modelo de despliegue, con el cual la aplicación será implementada en los servidores físicos de la ESPOCH. Para detallar el proceso se realiza un diagrama de despliegue. En la Figura IV. 7 se ilustra el los nodos involucrados en el despliegue del sistema. Los mismos que cumple con funciones específicas.

Servidor de base de datos Windows Server 2008.-Tiene instalado un servidor de base de datos Microsoft SQL Server 2008 que a su vez administra la base de datos OASIS.

Servidor aplicaciones Windows Server 2008.-Tiene instalado un servidor de Internet Information Server 7, que a su vez contendrá a los servicios web de información estudiantil WCF. Este servidor tiene una conexión directa con el servidor de base datos OASIS.

Servidor de base de datos Linux Centos 5.5.- Tiene instalado un servidor de base de datos Postgres SQL 9.1 para administrar la base de datos centralizada.

Servidor de aplicaciones Linux Centos 6.4.- Tiene instalado un servidor de aplicaciones Glassfish 4 que incluye al framework de servicios web METRO, desplegado de esta forma a la aplicación web. Este servidor se conecta con el servidor de aplicaciones Windows Server 2008, con el propósito de consumir los servicios web de información estudiantil WCF y además se conecta con el servidor de base de datos Linux Centos 5,4 para recuperar los datos del estudiante.

PC Estudiante.- Tiene instalado un explorador web, con el el cual se conecta al sitio de información para estudiantes desplegado en el servidor de aplicaciones Linux Cento6.4.

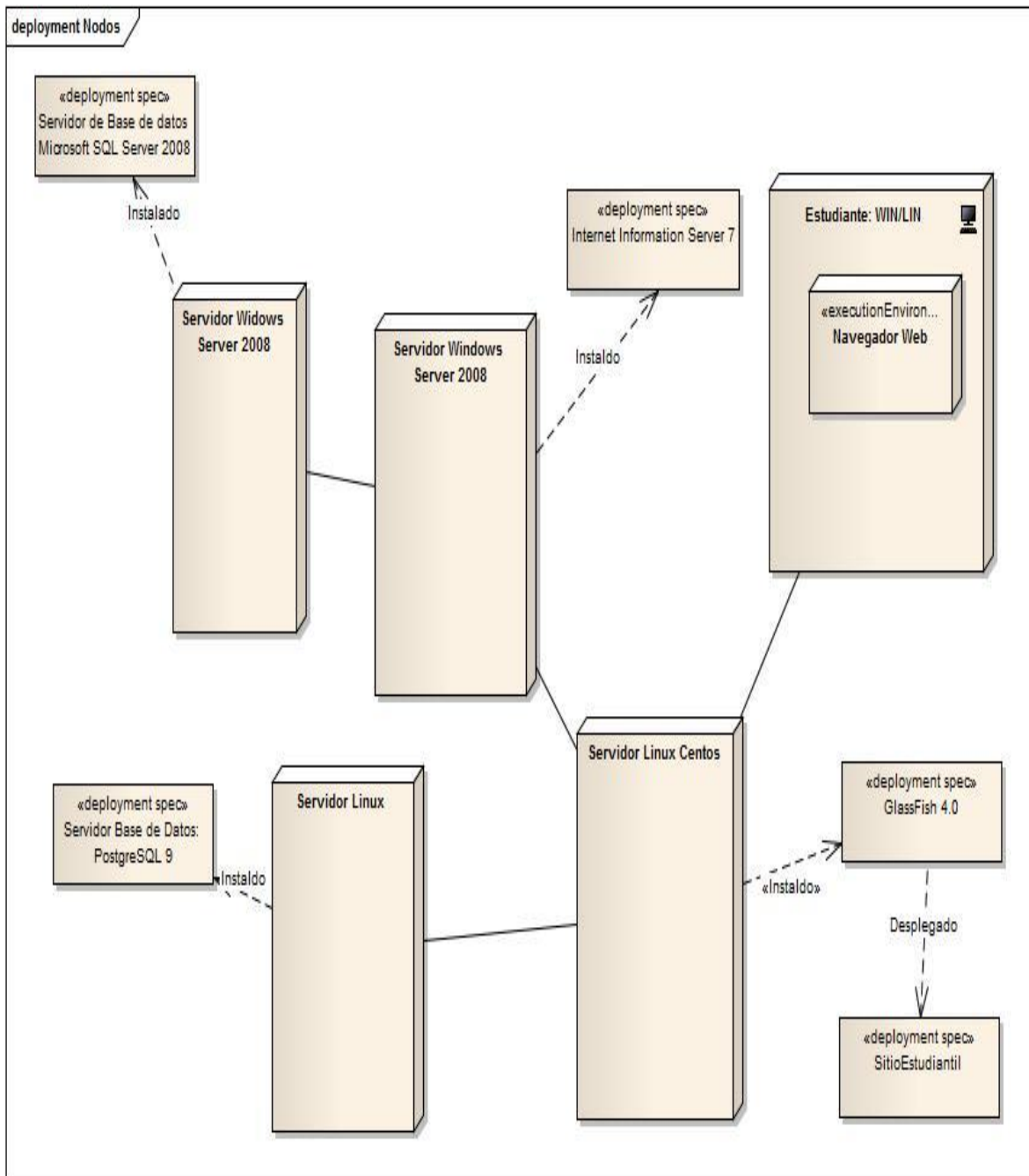


Figura IV. 7 Diagrama de despliegue del sitio de información para estudiantes

Fuente: Autor

4.3.5 Diseño de Interfaces

Para el desarrollo de las interfaces de usuario o pantallas se determinó un esquema gráfico que posee las siguientes características:

Banner.- Como norma o estándar para todas las pantallas se coloca una imagen en la parte superior izquierda, la misma que cuenta con el logo del sistema académico OASIS, en el centro se coloca el nombre del sitio de información para estudiantes que es “Sitio Estudiantil”.

Pie de página.- Contiene el nombre de la institución y dirección en el que se desarrolló el sistema.

Fuente.- Afín de mantener hegemonía en las páginas web se ha establecido la fuente denominada “Lucida Grande - 12”.

Menú.- Permiten agrupar opciones de navegación para el sitio web. Se realizó mediante un conjunto de botones que agrupan opciones horizontalmente, incluyendo la opción de despliegue automático de estas opciones. Además en la parte derecha se puede localizar un combo con las unidades en la cuales está inscrito, el nombre del usuario actual y un botón para cerrar la sesión.

Tablas.- Disponen de título contratado, distinción de títulos en las filas y columnas.

Botones de Comando.- Permiten realizar tareas tales como consultar, actualizar y visualizar. Su texto está centrado y poseen un oscurecimiento al estar sobre estos.

Todas estas características se ven reflejadas en las vistas o páginas web del sitio de información para estudiantes, con excepción de los reportes. En la Figura 8 se puede observar un ejemplo de la vista del sitio web.

Horario Niveles

Horario de Clase: SEGUNDO A

PRIMEROA	Hora_____	Lunes	Martes	Miercoles	Jueves	Viernes	Sabado	Domingo
PRIMEROB	PRIMERA HORA: 7h: 0min - 9h: 0min	PROGRAMACION E	FISICA II	PROGRAMACION E	FISICA II	PROGRAMACION E		
SEGUNDOA		DR. JULIO ROBERT	DR. JOSE RIGOBEN	DR. JULIO ROBERT	DR. JOSE RIGOBEN	DR. JULIO ROBERT		
SEGUNDOB	SEGUNDA HORA: 9h: 0min - 11h: 0min	MATEMATICA II	ELECTRONICA	MATEMATICA II	ELECTRONICA	MATEMATICA II		
TERCEROA		DR. MARIO HUMBE	ING. PAULINA ANDR	DR. MARIO HUMBE	ING. PAULINA ANDR	DR. MARIO HUMBE		
TERCEROB	TERCERA HORA: 11h: 0min - 13h: 0m	METODOLOGIA DE		ECOLOGIA Y MEDIC	METODOLOGIA DE			
		ING. GLADYS LORE		PATRICIO ADOLFO	ING. GLADYS LORE			
CUARTOA	CUARTA HORA: 13h: 0min - 15h: 0m							
QUINTOA	QUINTA HORA EXT: 14h: 0min - 16h: 0m							
SEXTOA								
SEXTOB								
SEPTIMOA								
SEPTIMOB								

ESPOCH-DTIC © 2014

Figura IV. 8 Vista del sitio web: Horio por paralelo


Reportes

El sitio dispone de la funcionalidad de exportar los datos de las tablas a un formato PDF, para que los mismos sean impresos. Los reportes poseen las siguientes características:

Encabezado.- Posee un banner que identifica al a institución, debajo de este se muestra la unidad académica, cédula de identidad, nombres, apellidos del estudiante y la fecha de la generación.

Cuerpo.- Consta generalmente de una tabla, en la cual las filas se distinguen mediante colores, para una mejor lectura del estudiante.

La fuente utilizada es “SansSerif-9” tanto para el encabezado y cuerpo de la página de impresión. Las características mencionadas se pueden apreciar en la Figura V. 9.



INGENIERÍA EN SISTEMAS

Cédula de Identidad: 030240840-6 Riobamba, lunes 16 septiembre 2013

Nombres y Apellidos: LUIS PATIÑO

Horario de Clase: SEPTIEMBRE 2007 - FEBRERO

Hora_____	Lunes	Martes	Miercoles	Jueves	Viernes	Sabado	Domingo
PRIMERA HORA 7:0 8:40	MATEMATICA BASICA (PRIMERO4) ING. DIEGO GUILLERMO BARBA MAGGI	GEOMETRIA Y TRIGONOMETRIA (PRIMERO4) ING. DIEGO GUILLERMO BARBA MAGGI	MATEMATICA BASICA PRIMERO4 ING. DIEGO GUILLERMO BARBA MAGGI	GEOMETRIA Y TRIGONOMETRIA (PRIMERO4) ING. DIEGO GUILLERMO BARBA MAGGI	GEOMETRIA Y TRIGONOMETRIA A (PRIMERO4) ING. DIEGO GUILLERMO BARBA MAGGI		
SEGUNDA HORA 8:40 10:20	FUNDAMENTOS DE PROGRAMACION (PRIMERO4) ING. VERONICA ELIZABETH MORA CHUNLLO	FUNDAMENTOS DE PROGRAMACION (PRIMERO4) ING. VERONICA ELIZABETH MORA CHUNLLO	FUNDAMENTOS DE PROGRAMACION PRIMERO4 ING. VERONICA ELIZABETH MORA CHUNLLO	FUNDAMENTOS DE PROGRAMACION (PRIMERO4) ING. VERONICA ELIZABETH MORA CHUNLLO	MATEMATICA BASICA (PRIMERO4) ING. DIEGO GUILLERMO BARBA MAGGI		
TERCERA HORA 10:30 12:10	FISICA BASICA (PRIMERO4) DR. JORGE MILTON LARA SINALUISA	FISICA BASICA (PRIMERO4) DR. JORGE MILTON LARA SINALUISA		FISICA BASICA (PRIMERO4) DR. JORGE MILTON LARA SINALUISA			

Figura IV. 9 Reporte horario de clase del estudiante

Fuente: Autor

El diseño de interfaces de usuario y la definición de los reportes permiten especificar la interacción del usuario con el sitio, así como dar una idea clara para el desarrollo de las mismas. Las páginas restantes que explican la funcionalidad total de la aplicación se muestra en el anexo 6.

4.4 Fase III. Codificación

Para la codificación del sitio web se empleó dos lenguajes de programación del lado servidor como los son C# .NET y Java, ya que son lenguajes altamente estables, interoperables y permiten la generación dinámica de contenidos HTML. Entre sus principales características cabe destacar su potencia, su alto rendimiento, su facilidad de aprendizaje y de desarrollo.

El sistema se elaboró bajo una arquitectura distribuida interoperable a fin de que esta sea modular y escalable. Se desarrolló en dos módulos:

4.4.1 BackEnd

Este módulo contiene los servicios web WCF fue desarrollado en .NET C# y está constituido por las siguientes capas:

Acceso a datos.- Para facilitar el desarrollo se hecho uso de Entity Framework que es compatible con WCF. Este framework genera dos archivos llamados “EstudiantilOAS_Carrera.edmx” y “EstudiantilOAS_Master.edmx” que contienen a su vez una entidad(Clases) por cada tabla seleccionada en la base de datos y métodos para el acceso y administración de los datos.

Servicios Web Estudiantil.- Contiene a los servicios web de información estudiantil desarrollados en .NET WCF haciendo uso de la capa de acceso a datos. Estos servicios web están constituidos por dos archivos, la interfaz y la clase de implementación, en la interfaz se definen la firma de los métodos y en la clase de implementación se desarrolla el método definido.

Servicio web en WCF

Interfaz IInfoCarrera.cs

```
[OperationContract]
List<Entidades.Nivel> getNiles(string strCodCarrera, string strCodPeriodo);
```

Clase implemtadora InfoCarrera

```
public List<Entidades.Nivel> getNiles(string strCodCarrera, string strCodPeriodo)
{
    List<Entidades.Nivel> lstNiveles = new List<Entidades.Nivel>();
    try
    {
        EntityConnection conexion = new
EntityConnection(FuncionesGenereales.getCadenenaConexionCarrera(this.GetBaseDatos(s
trCodCarrera)));
        OAS_CarreraEntities contexto = new OAS_CarreraEntities(conexion);
        var entidadesNiveles = (from entidadNivel in contexto.Niveles
                                join entidadCurso in contexto.Cursos on entidadNivel.strCodigo
equals entidadCurso.strCodNivel
                                where entidadCurso.strCodPeriodo.Equals(strCodPeriodo)
                                select new { entidadNivel.strCodigo, entidadNivel.strDescripcion
}).Distinct();
        if (entidadesNiveles != null)
        {
            //Obtengo Lista de Niveles a partir de la consulta.
            foreach (var entidadNivel in entidadesNiveles)
            {
                Entidades.Nivel nivel = new Nivel();
                nivel.IntCodNivel = Convert.ToInt32(entidadNivel.strCodigo);
                nivel.StrDescripcion = entidadNivel.strDescripcion;
                lstNiveles.Add(nivel);
            }
        }
    }
}
```

```
    }  
  
    lstNiveles = (from nivel in lstNiveles  
                 orderby nivel.IntCodNivel  
                 select nivel).ToList();  
    }  
    conexion.Dispose();  
    contexto.Dispose();  
    }  
    catch (Exception ex)  
    {  
        throw new Exception("Error al Obtener Niveles: " + ex.Message);  
    }  
    return lstNiveles;  
}
```

Luego del desarrollo el servicio web queda listo para ser publicado y en lo posterior ser consumido por algún cliente.

4.4.2 FrontEnd

Modulo cliente o aplicación web fue desarrollado en Java y está constituido de las siguientes capas:

Componente centralizado.- Proporciona todos los métodos y clases de acceso a los datos de la base de datos centralizada, desarrollada en Java. Este componente fue desarrollado por la DTIC y los métodos aceptan objetos como parámetros y realizan operaciones con la base de datos.

Función componente centralizado

```
public Persona encontrar(Object unSecuencial) throws Exception {  
    Persona persona = null;  
    try {  
        Persona _persona = new Persona();  
        _persona.setId((Long) unSecuencial);
```

```
List<Parametro> parametros =
this.getParametrosEntrada(PersonaParametros.OPERACION_SELECT_ID, _persona);
    DataBaseReader dbr =
AccesoDatos.ejecutaQuery(PersonaParametros.NOMBRE_PROCEDIMIENTO,
parametros);
    if (dbr.next()) {
        persona = this.cargaDatos(dbr);
    }
    dbr.close();
} catch (Exception e) {
    throw e;
}
return persona;
}
```

Mediante este componente se puede realizar visualización del perfil personal y de discapacidad del estudiante.

Proxy.- En esta capa se desarrolló haciendo uso de METRO y se realizó las configuraciones necesarias para lograr una comunicación segura con el servidor. Consta de dos elementos básicos el método de consumo y un intérprete de host.

Interprete de Host

```
static {
    javax.net.ssl.HttpURLConnection.setDefaultHostnameVerifier(
        new javax.net.ssl.HostnameVerifier() {
            @Override
            public boolean verify(String hostname,
                javax.net.ssl.SSLSession sslSession) {
                ResourceBundle recurso =
ResourceBundle.getBundle("ec.edu.esepoch.sitioestudiantil.Configuracion.General");
                String strExternalHostname = recurso.getString("hostname");
                return hostname.equals(strExternalHostname);
            }
        });
}
```

Método de consumo del servicio web

```
public ArrayOfNivel getNiles(java.lang.String strCodCarrera, java.lang.String
strCodPeriodo) throws Exception {
    try {
        ec.edu.esepoch.sitioestudiantil.ProxyWCF.Carrera.InfoCarrera service = new
ec.edu.esepoch.sitioestudiantil.ProxyWCF.Carrera.InfoCarrera();
        ec.edu.esepoch.sitioestudiantil.ProxyWCF.Carrera.IInfoCarrera port =
service.getBasicHttpBindingIInfoCarrera();
        return port.getNiles(strCodCarrera, strCodPeriodo);
    } catch (Exception e) {
        throw new Exception("Error en el Consumo del Servicio: " + e.getMessage());
    }
}
```

Mediante el uso de estos dos métodos se provee los datos necesarios a la capa de presentación.

Presentación.- Para esta última se utilizó el patrón de diseño arquitectónico MVC que permite separar los datos de la lógica de negocio, para esto se utilizó tres componentes internos tales como el modelo (clase), vista (XHTML) y controlador (lógica). Estos tres componentes fueron desarrollados con JSF y Primefaces.

Modelo.-Clase que contiene atributos y métodos descriptores de cada uno de estos.

```
package ec.edu.esepoch.sitioestudiantil.Entidades;
import java.io.Serializable;
public class MateriaEstudiante implements Serializable{
    private String strCodMateria;
    private String strDescripcion;
    public String getStrCodMateria() {
        return strCodMateria;
    }
    public void setStrCodMateria(String strCodMateria) {
```

```
        this.strCodMateria = strCodMateria;
    }
    public String getStrDescripcion() {
        return strDescripcion;
    }
    public void setStrDescripcion(String strDescripcion) {
        this.strDescripcion = strDescripcion;
    }
    //</editor-fold>
}
```

Vista.- Archivo XHTML que recibe las peticiones del usuario y las comunica al controlador.

```
<f:view xmlns="http://www.w3.org/1999/xhtml"
        xmlns:h="http://xmlns.jcp.org/jsf/html"
        xmlns:p="http://primefaces.org/ui"
    <html>
        <h:head>
            <title style="font-size: 4px; font-family: serif">Detalle Noticia - Sitio
Estudiantil</title>
            <link rel="stylesheet" type="text/css"
href="/SitioEstudiantil/faces/css/f/carousel.css" />
        </h:head>
        <h:body>
            <p:layout fullPage="true" >
                <p:layoutUnit position="north" >
                    <ui:include src="../plantilla/header.xhtml"/>
                </p:layoutUnit>
                <p:layoutUnit position="center" styleClass="centerCenter" >
                    <p:panel>
                        <h:form id="frmVerImagenCompleta">
                            <p:dialog id="dlgImagenCompleta" widgetVar="dlgImagenCompleta">
                                <p:graphicImage value="#{imagenStreamer.fileContentShow}" >
                                    <f:param id="idNoticia" value="#{detalle.strIdImagenCompleta}"/>
                                </p:graphicImage><br/>
                                <p:commandButton value="Aceptar" />
                            </p:dialog>
                        </h:form>
                    </p:panel>
                </p:layoutUnit>
                <p:layoutUnit position="south" >
            </h:body>
        </html>
    </f:view>
```

Controlador.- Clase que posee los métodos para recibir las peticiones enviadas por la vista y mediante el modelo genera otra vista.

```
@ManagedBean
@RequestScoped
public class TituloAcademico {
    private List<TitulosPregrado> lstTitulosPregrados;
    public List<TitulosPregrado> getLstTitulosPregrados() {
        return lstTitulosPregrados;
    }
    public void setLstTitulosPregrados(List<TitulosPregrado> lstTitulosPregrados) {
        this.lstTitulosPregrados = lstTitulosPregrados;
    }
    public TituloAcademico() {
        try {
            lstTitulosPregrados=new ArrayList<>();
            ExternalContext contexto =
FacesContext.getCurrentInstance().getExternalContext();
            SesionUsuario sesionUsuario = (SesionUsuario)
contexto.getSessionMap().get("sesionUsuario");
            InfoCarreraWS infoCarreraWS=new InfoCarreraWS();
            String strCedula = sesionUsuario.getRolAcademico().getStrCedula();
            List<TituloPreGradoEstudiante> listaPregrados =
infoCarreraWS.geTiutulosPregradoEstudiante(strCedula).getTituloPreGradoEstudiante();
            for (TituloPreGradoEstudiante tituloPreGradoEstudiante : listaPregrados) {
                TitulosPregrado titulosPregrado = new TitulosPregrado();

titulosPregrado.setStrNombreInstitucion(tituloPreGradoEstudiante.getStrNombreInstitucion().getValue());
                titulosPregrado.setStrFecha(tituloPreGradoEstudiante.getStrFecha().getValue());

titulosPregrado.setStrNombreTitulo(tituloPreGradoEstudiante.getStrNombreTitulo().getValue());

titulosPregrado.setStrRefrendacion(tituloPreGradoEstudiante.getStrRefrendacion().getValue());
                lstTitulosPregrados.add(titulosPregrado);
            }
        } catch (Exception e) {
            System.err.println("Error al Cargar Titulos Academicos:" + e.getMessage());
        }
    }
}
```

4.5 Fase IV. Pruebas

Uno de los pilares de la eXtreme Programming(XP) es el proceso de pruebas. XP anima a probar constantemente tanto como sea posible. Esto permite aumentar la calidad de los sistemas reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección. También permite aumentar la seguridad de evitar efectos colaterales no deseados a la hora de realizar modificaciones y refactorizaciones [17].

En base a lo anteriormente dicho se ha desarrollado un total de 17 pruebas funcionales una por cada historia de usuario. Cada prueba se constituye de los siguientes elementos:

Descripción.- Relato breve de lo que se desea realizar.

Entrada.- Secuencia de pasos para llevar a cabo la prueba

Resultado deseado.- Es el resultado con el cual se desea concluir.

Cumplimiento de la prueba.- Si la prueba cumple o no con el resultado deseado.

Tabla resumen.- La tabla resumen a las actividades realizadas, el resultado deseado y el cumplimiento de la prueba.

A manera de ejemplo se presenta a continuación un ejemplo de la prueba realiza a la historia de usuario 1, las demás pruebas se pueden apreciar en el anexo7.

Prueba para la historia de usuario 1: Autenticación del estudiante

Descripción

El usuario estudiante podrá autenticarse, ingresando sus credenciales para iniciar sesión en el sistema.

Entrada

- 1.- El usuario estudiante una vez cargada la página de inicio del sitio, procede a situarse en los campos de ingreso de credenciales.
- 2.- Ingresar el número de cédula y su clave.
- 3.- Ingresar al sistema.

Resultado deseado

Ingresado las credenciales del usuario estudiante, el sistema inicia sesión con el usuario actual.

Cumplimiento de la Prueba

Prueba superada con éxito.

Tabla IV. X Prueba de la historia de usuario 1

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Física y Matemática de la ESPOCH	Autenticación del estudiante	Ingresado los credenciales del usuario estudiante, el sistema inicia sesión con el usuario actual	Si
Modificaciones	Luis Patiño			

Fuente: Autor

De las pruebas realizadas se logró identificar que el sistema presentaba errores de un 3% en las salidos de la aplicación, los mismos que fueron corregidos, llegando de esta manera al cumplimiento de los requerimientos específicos definidos en la primera fase, por lo cual el sistema fue aceptado y recibido por la Dirección de Tecnologías de la Información y Comunicación.

CONCLUSIONES

1. El estudio de las características de los frameworks de servicios web ha permitido determinar que Metro posee gran portabilidad y compatibilidad con .NET WCF, mientras que Axis2 posee una arquitectura modular con varios enlaces de datos permitiendo adecuar a las necesidades de las aplicaciones interoperables.
2. El análisis comparativo basado en parámetros de desempeño permitió determinar que Metro es mejor para el desarrollo de aplicaciones que consuman servicios web WCF con un porcentaje de 91.67% de cumplimiento de expectativas frente a un 58.33% de Axis2.
3. Se seleccionó Metro para desarrollar el sitio de información para estudiantes, debido a que mejora el desempeño de las aplicaciones que consumen servicios web WCF en relación a Axis2, con lo que se logró integrar la investigación con un caso aplicativo real, en beneficio de la ESPOCH.
4. Se desarrollaron los nuevos servicios web de información estudiantil mediante el framework de servicios web.NET WCF, necesarios para la construcción del sitio de información para estudiantes.
5. El desarrollo del sitio de información para estudiantes de la ESPOCH permitió elevar la funcionalidad del sitio académico OASIS actual.
6. Metro propone un desarrollo simple y eficiente de aplicaciones que consuman servicios web WCF de un 25% en comparación a Axis2, haciendo de la interoperabilidad una herramienta integradora para alcanzar una única solución.

RECOMENDACIONES

1. Se recomienda al personal del área de desarrollo de la Dirección de Tecnologías de la Información y Comunicación de la ESPOCH haga uso de la documentación del sitio de información para estudiantes, para realizar una adecuada administración y mantenimiento al sistema.
2. Se recomienda la creación de servicios web con Metro y el consumo de los mismos mediante WCF, para lograr así una interoperabilidad entre Java y .NET absoluta.
3. Se recomienda la creación de una metodología de desarrollo de software basada en proceso ágiles que ayuden al desarrollo de aplicaciones en equipos reducidos.
4. Analizar el funcionamiento de los frameworks en otros contenedores web, tal como Apache Tomcat, JBoss y WebSphere para optar un posible cambio de servidor web.
5. Que el resultado de la tesis “ANÁLISIS COMPARATIVO DE METRO Y AXIS2 PARA EL DESARROLLO DE APLICACIONES QUE CONSUMAN SERVICIOS WEB WCF EN LA ESPOCH” sirva como base para estados posteriores de interoperabilidad, escalabilidad y portabilidad de aplicaciones en la ESPOCH.
6. Se recomienda investigar sobre los mecanismos de seguridad en autenticación y firmado mediante certificados digitales que poseen cada uno de los frameworks

RESUMEN

Esta Investigación pretende comparar Metro y Axis2 orientando a desarrollar aplicaciones que consuman servicios web WCF (Windows Communication Foundation) en la Escuela Superior Politécnica de Chimborazo. Se aplicó el método científico inductivo para registrar, clasificar y analizar resultados de pruebas realizadas a prototipos desarrollados en parámetros comparativos establecidos, demostrando así que Metro provee mejores prestaciones en comparación a Axis2. Además emplee pruebas de carga, estrés, observaciones como técnicas para recolectar datos.

Para analizar y desarrollar la presente investigación se manejó los siguientes materiales en hardware: dos computadoras portátiles y software: JMeter, Java Visual Virtual Machine, framework de servicios web Metro, Axis2, Netbeans 7.4, Prime faces, Entity Framework, JSF (Java Server Faces) , eclipse Indigo, Visual Studio 2010, como servidores de base de datos SQL SERVER 2008 y Postgres SQL 9.1, Jasper Report, servidores de aplicaciones Internet Information Server 7, Glassfish 4.

Mediante resultados obtenidos y de acuerdo a parámetros de comparación: Rendimiento, Requerimientos Hardware, Simplicidad, se determinó un 91.7% para Metro equivalente a Excelente y 58.33% para Axis2.equivalente a Bueno.

Se ha logrado crear un sitio estudiantil para estudiantes de la Escuela superior Politécnica de Chimborazo empleando Metro, puesto que este es más adecuado para desarrollar aplicaciones que consuman servicios web, brindando beneficios: robustez, estabilidad, accesibilidad, reducción en tiempos y usabilidad al sitio desarrollado

Se recomienda a los estudiantes hacer uso del sito estudiantil, debido a que este presenta mejores condiciones que el sistema académico OASIS (Orion Academic System with Internet Services).

SUMMARY

This research aims to compare Metro and Axis2 orientated to develop applications that consume Web services WCF (Windows Communication Foundation) at Escuela Superior Politécnica de Chimborazo. Scientific inductive method was applied in order to register, classify, and analyze test results performed to developed prototypes in established comparative parameters, so it shows that Metro provides better performance in comparison to Axis2. Besides, Carrying, stress, and observations tests; and techniques were used in order to collect data.

Hardware like two laptops and software like JMeter, and Java Visual Virtual Machine, framework of services web Metro, Axis2, Netbeans 7,4, Prime faces, Entity Framework, JSF (Java Server Faces), Indigo eclipse, Visual Studio 2010, as data base servers SQL SERVER 2008 and Postgres SQL 9.1, Jasper Report, servers of information internet applications Server 7, Glassfish 4.

Performance, hardware requirements, and simplicity were determined at 91.7% for Metro from results obtained and according to comparison parameters which means an excellent and 58.33% for Axis2 which means a good.

Creating a student site for pupils at Escuela Superior Politecnica de Chimborazo by using Metro, because it is the best to develop applications that consume web services, giving benefits such as: hardiness, stability, accessibility, and reduction in times and usefulness of the site developed.

It is recommended to students to use this student site because this has better conditions than OASIS academic system (Orion Academic System with Internet Services)

GLOSARIO

ADB	Enlace de datos de Axis2, convierte el XML a estructuras de código fuente Java
Axis2	Framework que permite la creación y generación de servicios y cliente web respectivamente en Java y C++
BackEnd	Módulo servidor de un sistema, que contiene los servicios web.
FrontEnd	Modulo cliente de un sistema, contiene el consumo de los servicios web y la interfaz de usuario
HTTP	Protocolo de transferencia de Hipertexto
HTTPS	Protocolo de transferencia de Hipertexto de manera seguro, vía SSL
IIS	Servidor de aplicaciones de Microsoft, posee la funcionalidad de desplegar los servicios web y asegúralos.
JAXB	Enlace de Datos para Metro, es intermediario entre instancias de documentos XML y Java
JSF	Framework para aplicaciones Java basadas en web que simplifica el desarrollo de interfaces de usuario
MVC	Patrón de diseño arquitectónico, que contempla un modelo, vista y controlador para la capa de presentación de una aplicación web.
Metro	Framework que permite la creación y generación de servicios y cliente web respectivamente en Java
SSL	Mecanismo que permite establecer un canal seguro, por el cual viajan los datos desde el cliente hacia el servidor.

WCF Framework de servicios que integran modos de comunicación tales como los web y datos.

XP Metodología ágil de desarrollo de software, orientada a la programación.

ANEXOS

Anexo1. Pruebas de Rendimiento

Pruebas de carga con JMeter

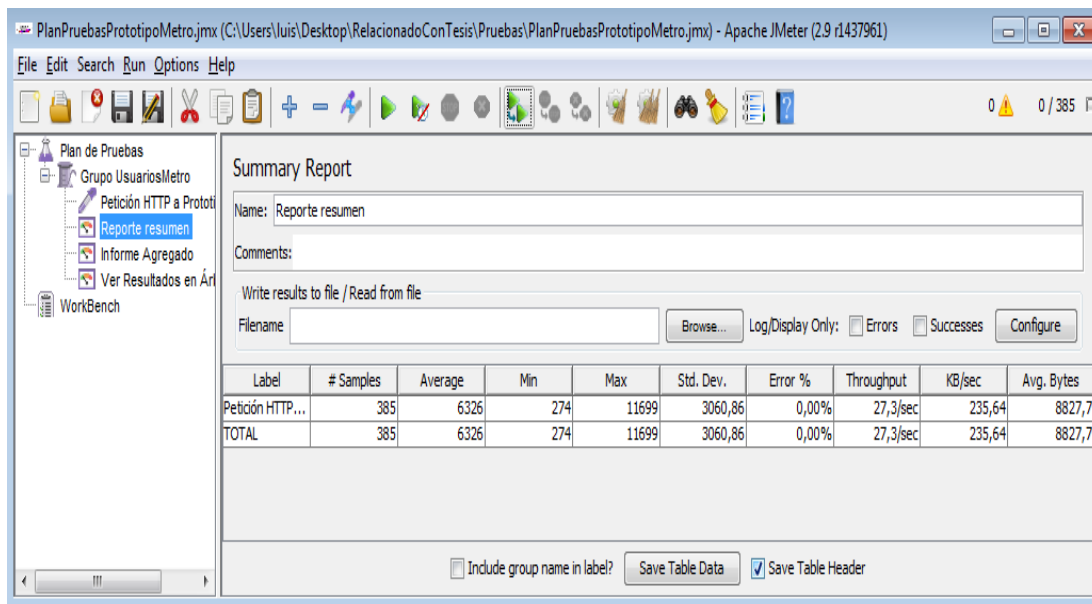
Se realizan las pruebas, según el procedimiento explicado en el capítulo III. Las presentes pruebas proveen resultados tanto para el indicador throughput como para el tiempo de respuesta.

El throughput tiene su propio campo dentro de la herramienta, mientras que para el tiempo de respuesta es el average que se toma como resultado.

Las Pruebas fueron ejecutadas para Metro y Axs2.

METRO

Prueba1



The screenshot shows the Apache JMeter Summary Report window. The report title is 'Summary Report' and the name is 'Reporte resumen'. The test results are as follows:

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg. Bytes
Petición HTTP...	385	6326	274	11699	3060,86	0,00%	27,3/sec	235,64	8827,7
TOTAL	385	6326	274	11699	3060,86	0,00%	27,3/sec	235,64	8827,7

Additional options at the bottom include 'Include group name in label?' (unchecked), 'Save Table Data' (button), and 'Save Table Header' (checked).

Prueba2

PlanPruebasPrototipoMetro.jmx (C:\Users\luis\Desktop\RelacionadoConTesis\Pruebas\PlanPruebasPrototipoMetro.jmx) - Apache JMeter (2.9 r1437961)

File Edit Search Run Options Help

Summary Report

Name: Reporte resumen

Comments:

Write results to file / Read from file

Filename Browse... Log/Display Only: Errors Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg. Bytes
Petición HTTP ...	385	6527	256	11657	3138,61	0,00%	27,2/sec	234,57	8827,6
TOTAL	385	6527	256	11657	3138,61	0,00%	27,2/sec	234,57	8827,6

Include group name in label? Save Table Header

Prueba3

PlanPruebasPrototipoMetro.jmx (C:\Users\luis\Desktop\RelacionadoConTesis\Pruebas\PlanPruebasPrototipoMetro.jmx) - Apache JMeter (2.9 r1437961)

File Edit Search Run Options Help

Summary Report

Name: Reporte resumen

Comments:

Write results to file / Read from file

Filename Browse... Log/Display Only: Errors Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg. Bytes
Petición HTTP ...	385	6708	399	11961	3059,02	0,00%	27,1/sec	233,45	8827,8
TOTAL	385	6708	399	11961	3059,02	0,00%	27,1/sec	233,45	8827,8

Include group name in label? Save Table Header

Prueba4

Summary Report

Name: Reporte resumen

Comments:

Write results to file / Read from file

Filename: Browse... Log/Display Only: Errors Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg. Bytes
Petición HTTP ...	385	6705	434	12047	3290,65	0,00%	27,1/sec	233,34	8827,2
TOTAL	385	6705	434	12047	3290,65	0,00%	27,1/sec	233,34	8827,2

Include group name in label? Save Table Header

Prueba5

Summary Report

Name: Reporte resumen

Comments:

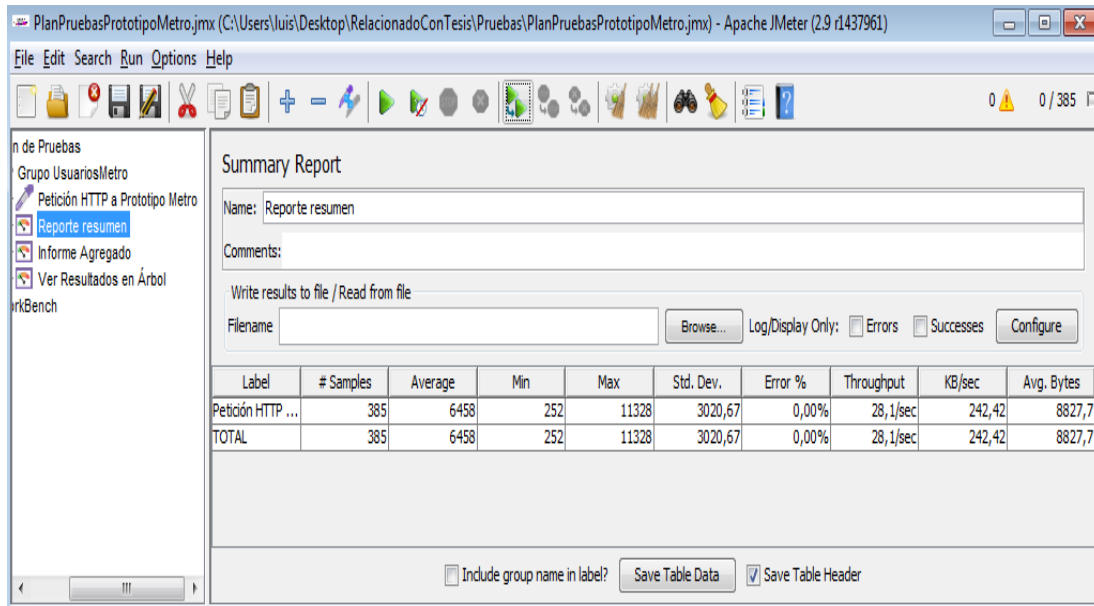
Write results to file / Read from file

Filename: Browse... Log/Display Only: Errors Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg. Bytes
Petición HTTP ...	385	6346	356	11809	3050,59	0,00%	27,3/sec	235,75	8826,6
TOTAL	385	6346	356	11809	3050,59	0,00%	27,3/sec	235,75	8826,6

Include group name in label? Save Table Header

Prueba6

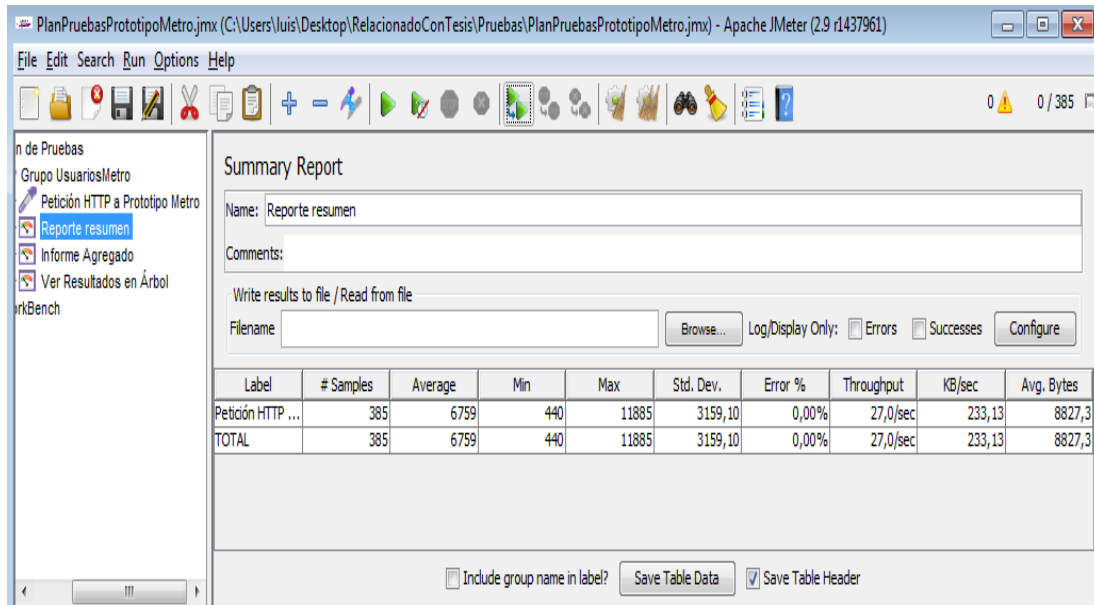


The screenshot shows the Apache JMeter Summary Report window. The title bar indicates the file path and version: "PlanPruebasPrototipoMetro.jmx (C:\Users\luis\Desktop\RelacionadoConTesis\Pruebas\PlanPruebasPrototipoMetro.jmx) - Apache JMeter (2.9 r1437961)". The menu bar includes File, Edit, Search, Run, Options, and Help. The toolbar contains various icons for file operations and test execution. On the left, a tree view shows the test plan structure: "n de Pruebas", "Grupo UsuariosMetro", "Petición HTTP a Prototipo Metro", "Reporte resumen", "Informe Agregado", and "Ver Resultados en Árbol". The main area displays the "Summary Report" for "Reporte resumen". It includes a "Name" field with the value "Reporte resumen" and a "Comments" field. Below these is a section for "Write results to file / Read from file" with a "Filename" field and a "Browse..." button. There are also checkboxes for "Log/Display Only: Errors" and "Successes", and a "Configure" button. A table summarizes the test results:

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg. Bytes
Petición HTTP ...	385	6458	252	11328	3020,67	0,00%	28,1/sec	242,42	8827,7
TOTAL	385	6458	252	11328	3020,67	0,00%	28,1/sec	242,42	8827,7

At the bottom, there are checkboxes for "Include group name in label?" and "Save Table Data", and a checked checkbox for "Save Table Header".

Prueba7



The screenshot shows the Apache JMeter Summary Report window for Prueba7. The title bar and menu bar are identical to the previous screenshot. The tree view on the left is also the same. The main area displays the "Summary Report" for "Reporte resumen". It includes a "Name" field with the value "Reporte resumen" and a "Comments" field. Below these is a section for "Write results to file / Read from file" with a "Filename" field and a "Browse..." button. There are also checkboxes for "Log/Display Only: Errors" and "Successes", and a "Configure" button. A table summarizes the test results:

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg. Bytes
Petición HTTP ...	385	6759	440	11885	3159,10	0,00%	27,0/sec	233,13	8827,3
TOTAL	385	6759	440	11885	3159,10	0,00%	27,0/sec	233,13	8827,3

At the bottom, there are checkboxes for "Include group name in label?" and "Save Table Data", and a checked checkbox for "Save Table Header".

Prueba8

PlanPruebasPrototipoMetro.jmx (C:\Users\luis\Desktop\RelacionadoConTesis\Pruebas\PlanPruebasPrototipoMetro.jmx) - Apache JMeter (2.9 r1437961)

Archivo Editar Search Lanzar Opciones Ayuda

Plan de Pruebas

- Grupo UsuariosMetro
 - Petición HTTP a Prototipo Metro
 - Reporte resumen
 - Informe Agregado
 - Ver Resultados en Árbol
- Banco de Trabajo

Reporte resumen

Nombre: Reporte resumen

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo Navegar... Log/Mostrar sólo: Escribir en Log Sólo Errores Éxitos

Etiqueta	# Muestras	Media	Min	Máx	Desv. Están...	% Error	Rendimiento	Kb/sec	Media de By...
Petición HTT...	385	6379	390	11605	3003,25	0,00%	27,6/sec	237,48	8823,1
Total	385	6379	390	11605	3003,25	0,00%	27,6/sec	237,48	8823,1

¿Incluir el nombre del grupo en la etiqueta? Guardar la cabecera de la tabla

Prueba9

PlanPruebasPrototipoMetro.jmx (C:\Users\luis\Desktop\RelacionadoConTesis\Pruebas\PlanPruebasPrototipoMetro.jmx) - Apache JMeter (2.9 r1437961)

File Edit Search Run Options Help

Plan de Pruebas

- Grupo UsuariosMetro
 - Petición HTTP a Prototipo Metro
 - Reporte resumen
 - Informe Agregado
 - Ver Resultados en Árbol
- WorkBench

Summary Report

Name: Reporte resumen

Comments:

Write results to file / Read from file

Filename Browse... Log/Display Only: Errors Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg. Bytes
Petición HTTP ...	385	6314	424	11763	3031,50	0,00%	27,7/sec	238,93	8827,8
TOTAL	385	6314	424	11763	3031,50	0,00%	27,7/sec	238,93	8827,8

Include group name in label? Save Table Header

Prueba10

PlanPruebasPrototipoMetro.jmx (C:\Users\luis\Desktop\RelacionadoConTesis\Pruebas\PlanPruebasPrototipoMetro.jmx) - Apache JMeter (2.9 r1437961)

Archivo Editar Search Lanzar Opciones Ayuda

Plan de Pruebas

- Grupo UsuariosMetro
- Petición HTTP a Prototipo Metro
- Reporte resumen
- Informe Agregado
- Ver Resultados en Árbol

Banco de Trabajo

Reporte resumen

Nombre: Reporte resumen

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo Log/Mostrar sólo: Escribir en Log Sólo Errores Éxitos

Etiqueta	# Muestras	Media	Min	Máx	Desv. Están...	% Error	Rendimiento	Kb/sec	Media de By...
Petición HTT...	385	6222	475	11159	2805,99	0,00%	27,7/sec	239,15	8825,7
Total	385	6222	475	11159	2805,99	0,00%	27,7/sec	239,15	8825,7

¿Incluir el nombre del grupo en la etiqueta? Guardar la cabecera de la tabla

AXIS2

Prueba1

PlanPruebasPrototipoAxis2.jmx (C:\Users\luis\Desktop\RelacionadoConTesis\Pruebas\PlanPruebasPrototipoAxis2.jmx) - Apache JMeter (2.9 r1437961)

File Edit Search Run Options Help

Plan de Pruebas Prototipo Axis2

- Grupo de Hilos Prototipo Axis2
- Petición HTTP a Prototipo Axis2
- Informe Agregado
- Reporte resumen
- Ver Resultados en Árbol
- jp@gc - Transactions per Second

Banco de Trabajo

Aggregate Report

Name: Informe Agregado

Comments:

Write results to file / Read from file

Filename Log/Display Only: Errors Successes

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
Petición HTTP...	385	13576	13397	25038	376	27690	0,00%	12,9/sec	122,4
TOTAL	385	13576	13397	25038	376	27690	0,00%	12,9/sec	122,4

Include group name in label? Save Table Header

Prueba2

The screenshot shows the Apache JMeter 2.9 interface displaying an Aggregate Report for a test plan named 'Plan de Pruebas Prototipo Axis2'. The report is titled 'Informe Agregado' and shows the following data:

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
Petición HTTP...	385	14296	14284	25461	404	28099	0,00%	12,7/sec	120,9
TOTAL	385	14296	14284	25461	404	28099	0,00%	12,7/sec	120,9

Additional details from the screenshot include the test name 'Informe Agregado', a comment field, and options to 'Write results to file / Read from file'. The 'Log/Display Only' section has 'Errors' and 'Successes' checkboxes, both of which are unchecked. At the bottom, there are checkboxes for 'Include group name in label?' (unchecked), 'Save Table Data' (checked), and 'Save Table Header' (checked).

Prueba3

The screenshot shows the Apache JMeter 2.9 interface displaying an Aggregate Report for a test plan named 'Plan de Pruebas Prototipo Axis2'. The report is titled 'Informe Agregado' and shows the following data:

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
Petición HTTP...	385	14080	14218	24880	452	27669	0,00%	12,9/sec	122,6
TOTAL	385	14080	14218	24880	452	27669	0,00%	12,9/sec	122,6

Additional details from the screenshot include the test name 'Informe Agregado', a comment field, and options to 'Write results to file / Read from file'. The 'Log/Display Only' section has 'Errors' and 'Successes' checkboxes, both of which are unchecked. At the bottom, there are checkboxes for 'Include group name in label?' (unchecked), 'Save Table Data' (checked), and 'Save Table Header' (checked).

Prueba4

Plan de Pruebas Prototipo Axis2
Grupo de Hilos Prototipo Axis2
Petición HTTP a Prototipo Axis2
Informe Agregado
Reporte resumen
Ver Resultados en Árbol
jp@gc - Transactions per Second
Banco de Trabajo

Aggregate Report

Name: Informe Agregado
Comments:

Write results to file / Read from file
Filename: Log/Display Only: Errors Successes

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
Petición HTTP...	385	14485	14406	25801	432	28712	0,00%	12,5/sec	118,5
TOTAL	385	14485	14406	25801	432	28712	0,00%	12,5/sec	118,5

Include group name in label? Save Table Header

Prueba5

Plan de Pruebas Prototipo Axis2
Grupo de Hilos Prototipo Axis2
Petición HTTP a Prototipo Axis2
Informe Agregado
Reporte resumen
Ver Resultados en Árbol
jp@gc - Transactions per Second
Banco de Trabajo

Aggregate Report

Name: Informe Agregado
Comments:

Write results to file / Read from file
Filename: Log/Display Only: Errors Successes

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
Petición HTTP...	385	14218	14245	25403	392	28096	0,00%	12,7/sec	121,0
TOTAL	385	14218	14245	25403	392	28096	0,00%	12,7/sec	121,0

Include group name in label? Save Table Header

Prueba6

Plan de Pruebas Prototipo Axis2
Grupo de Hilos Prototipo Axis2
Petición HTTP a Prototipo Axis2
Informe Agregado
Reporte resumen
Ver Resultados en Árbol
jp@gc - Transactions per Second
Banco de Trabajo

Aggregate Report

Name: Informe Agregado

Comments:

Write results to file / Read from file

Filename: Browse... Log/Display Only: Errors Successes

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
Petición HTTP...	385	14546	14809	25865	446	28429	0,00%	12,6/sec	119,5
TOTAL	385	14546	14809	25865	446	28429	0,00%	12,6/sec	119,5

Include group name in label? Save Table Header

Prueba7

Plan de Pruebas Prototipo Axis2
Grupo de Hilos Prototipo Axis2
Petición HTTP a Prototipo Axis2
Informe Agregado
Reporte resumen
Ver Resultados en Árbol
jp@gc - Transactions per Second
Banco de Trabajo

Aggregate Report

Name: Informe Agregado

Comments:

Write results to file / Read from file

Filename: Browse... Log/Display Only: Errors Successes

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
Petición HTTP...	385	14126	14112	25192	383	27761	0,00%	12,9/sec	122,3
TOTAL	385	14126	14112	25192	383	27761	0,00%	12,9/sec	122,3

Include group name in label? Save Table Header

Prueba8

PlanPruebasPrototipoAxis2.jmx (C:\Users\luis\Desktop\RelacionadoConTesis\Pruebas\PlanPruebasPrototipoAxis2.jmx) - Apache JMeter (2.9 r1437961)

File Edit Search Run Options Help

Plan de Pruebas Prototipo Axis2

- Grupo de Hilos Prototipo Axis2
 - Petición HTTP a Prototipo Axis2
 - Informe Agregado
 - Reporte resumen
 - Ver Resultados en Árbol
 - jp@gc - Transactions per Second
- Banco de Trabajo

Aggregate Report

Name: Informe Agregado

Comments:

Write results to file / Read from file

Filename: Browse... Log/Display Only: Errors Successes

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
Petición HTTP...	385	13581	13584	24383	510	27220	0,00%	13,1/sec	124,3
TOTAL	385	13581	13584	24383	510	27220	0,00%	13,1/sec	124,3

Include group name in label? Save Table Header

Prueba9

PlanPruebasPrototipoAxis2.jmx (C:\Users\luis\Desktop\RelacionadoConTesis\Pruebas\PlanPruebasPrototipoAxis2.jmx) - Apache JMeter (2.9 r1437961)

File Edit Search Run Options Help

Plan de Pruebas Prototipo Axis2

- Grupo de Hilos Prototipo Axis2
 - Petición HTTP a Prototipo Axis2
 - Informe Agregado
 - Reporte resumen
 - Ver Resultados en Árbol
 - jp@gc - Transactions per Second
- Banco de Trabajo

Aggregate Report

Name: Informe Agregado

Comments:

Write results to file / Read from file

Filename: Browse... Log/Display Only: Errors Successes

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
Petición HTTP...	385	14302	14311	25489	426	28047	0,00%	12,7/sec	121,2
TOTAL	385	14302	14311	25489	426	28047	0,00%	12,7/sec	121,2

Include group name in label? Save Table Header

Prueba10

PlanPruebasPrototipoAxis2.jmx (C:\Users\luis\Desktop\RelacionadoConTesis\Pruebas\PlanPruebasPrototipoAxis2.jmx) - Apache JMeter (2.9 r1437961)

File Edit Search Run Options Help

Informe Agregado

Name: Informe Agregado

Comments:

Write results to file / Read from file

Filename Browse... Log/Display Only: Errors Successes

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
Petición HTTP...	385	14367	14122	25874	418	28402	0,00%	12,6/sec	119,8
TOTAL	385	14367	14122	25874	418	28402	0,00%	12,6/sec	119,8

Include group name in label? Save Table Header

Anexo2. Pruebas de Requerimientos de Hardware

Pruebas JVisual VM

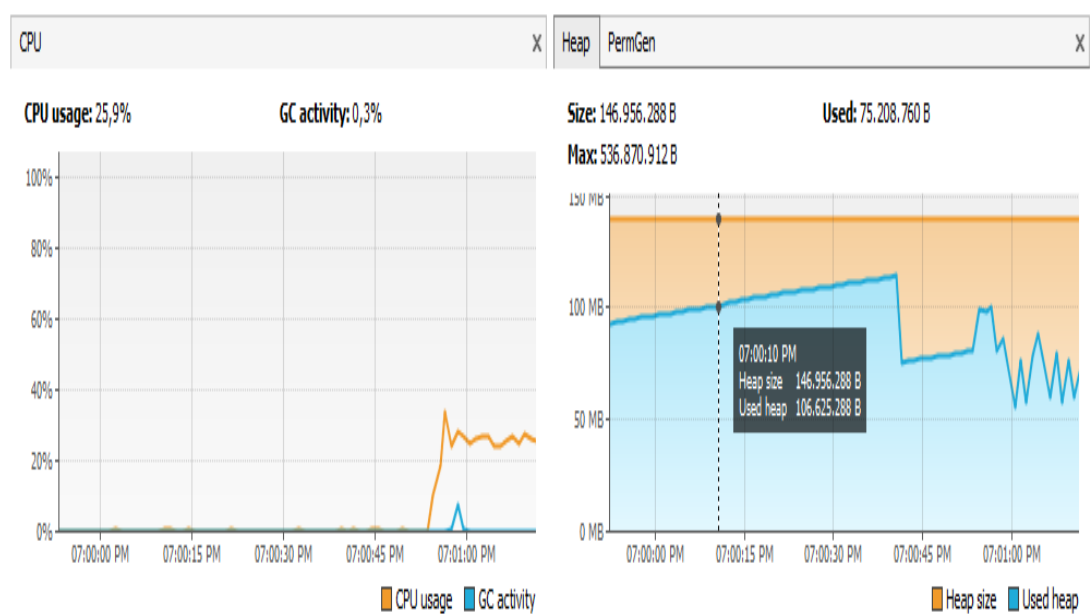
Se realizan las pruebas, según el procedimiento explicado en el capítulo III. Las presentes pruebas proveen resultados tanto para el indicador uso memoria RAM como para el uso del procesador.

El Uso de la memoria RAM se mide en MB y el uso del procesador en %CPU.

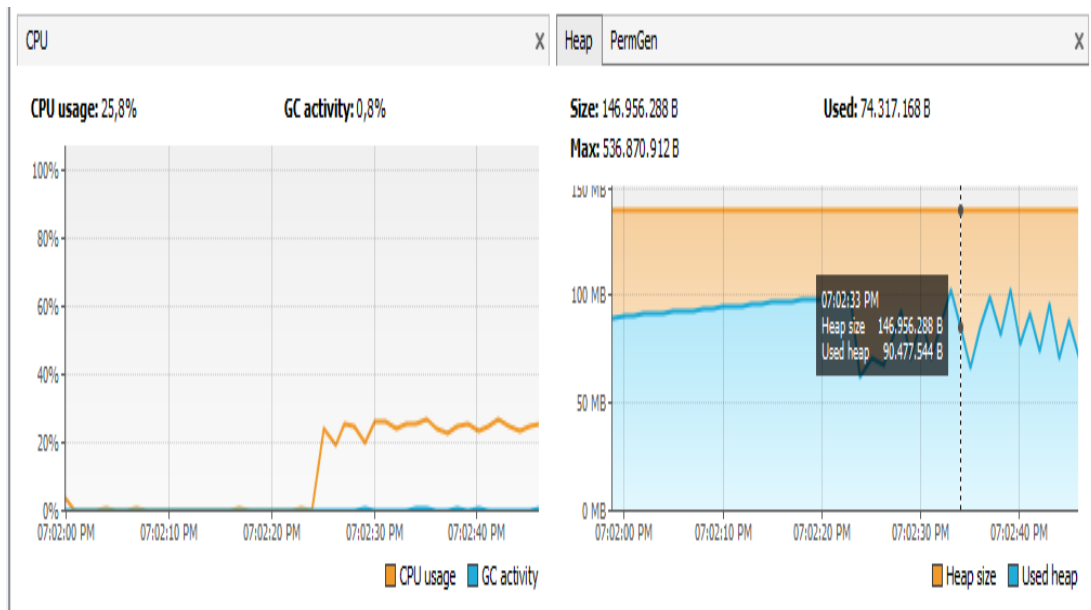
Las Pruebas fueron ejecutadas para Metro y Axs2.

Metro

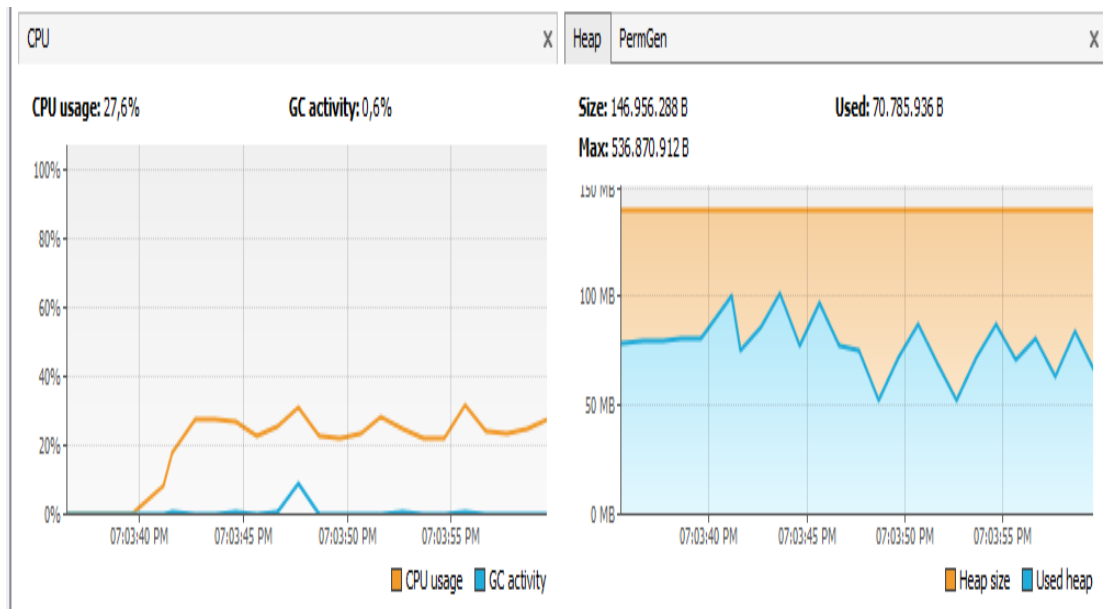
Prueba1



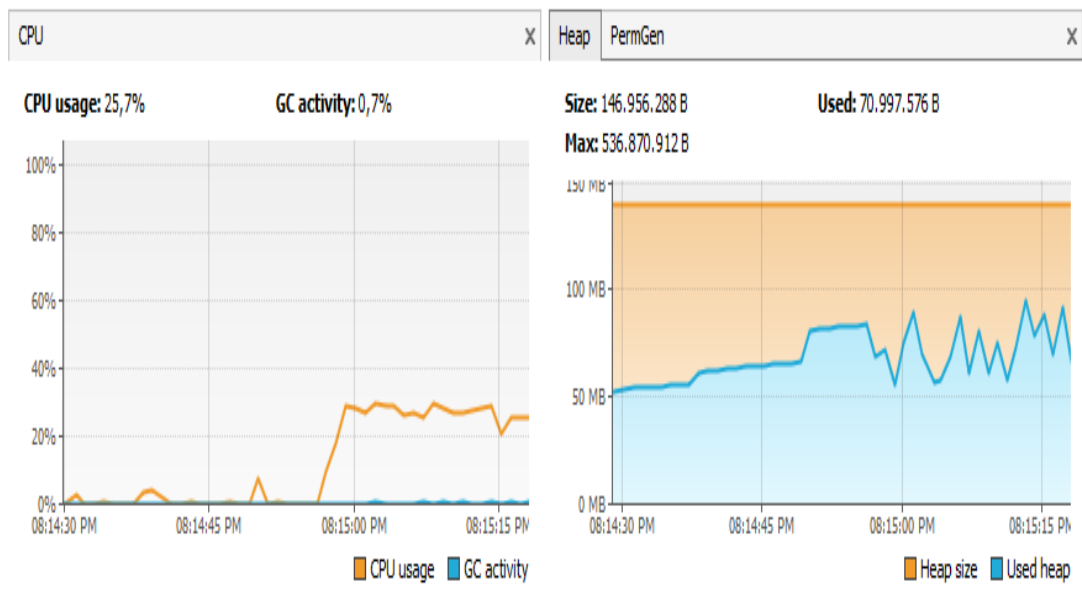
Prueba2



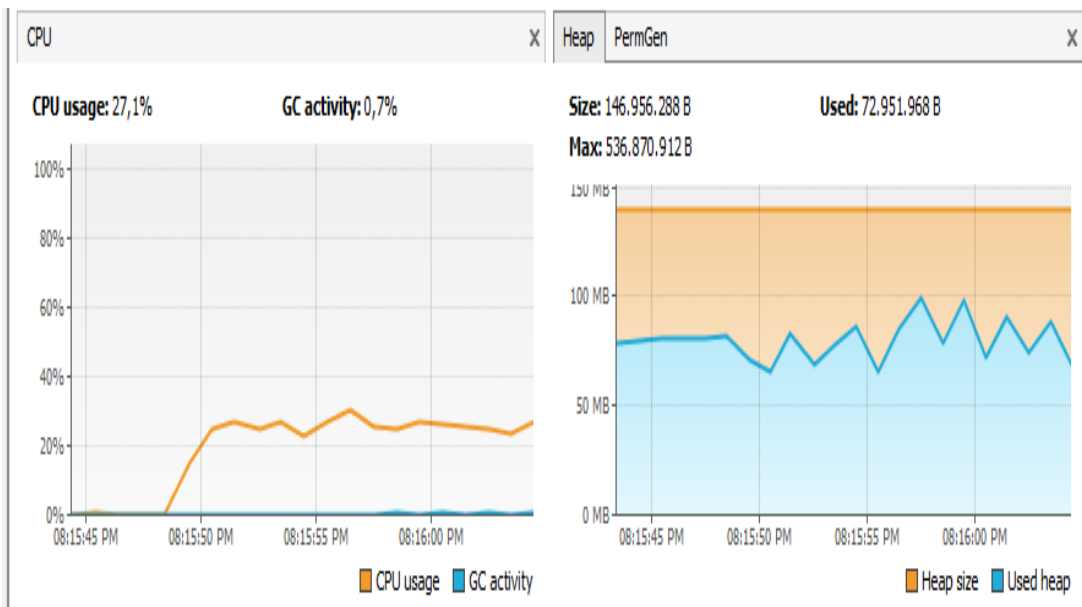
Prueba3



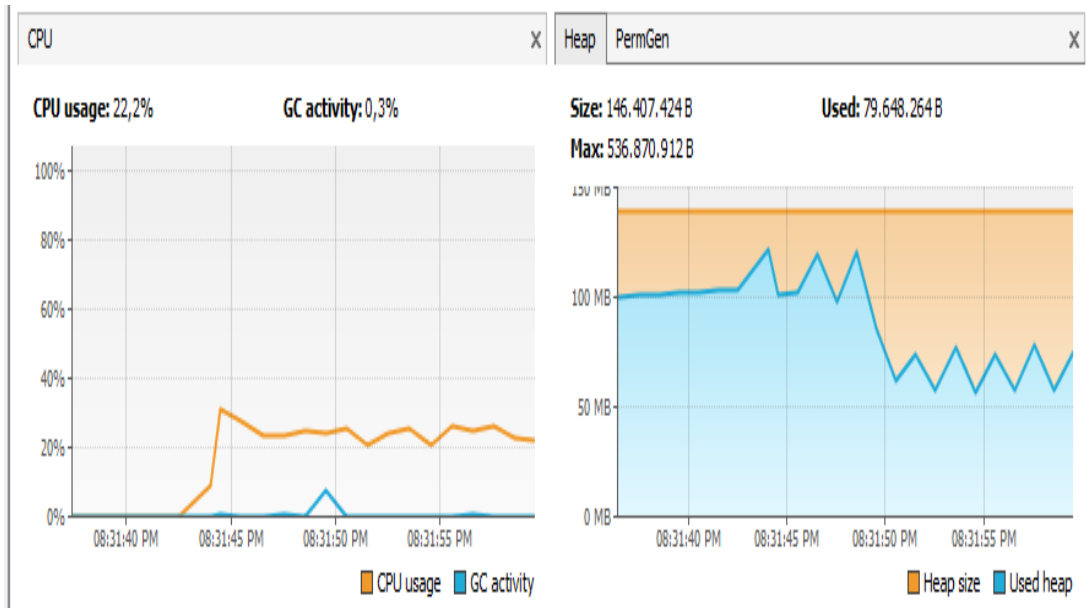
Prueba4



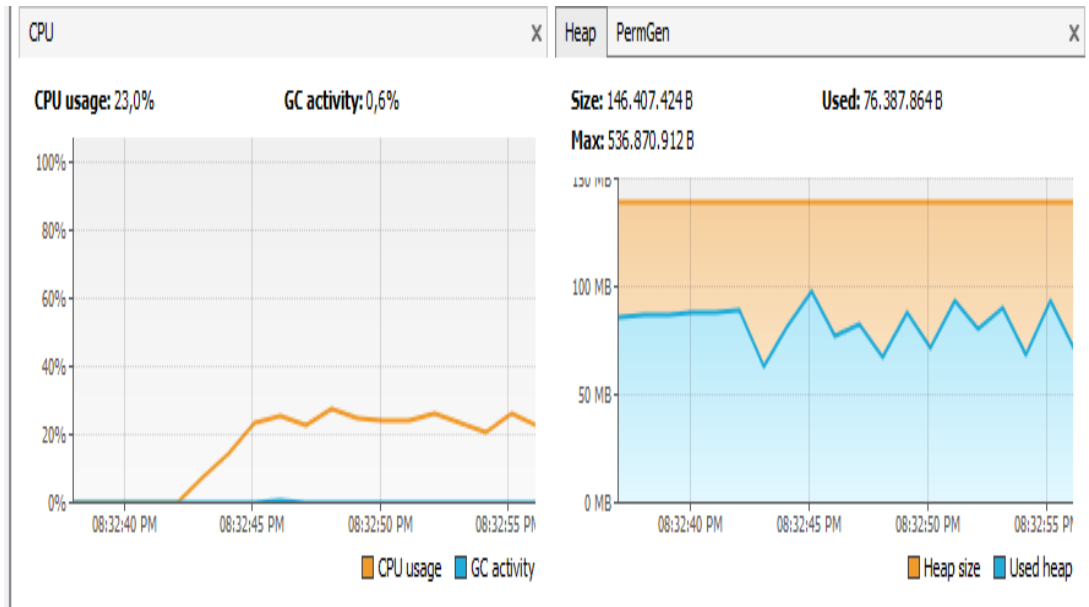
Prueba5



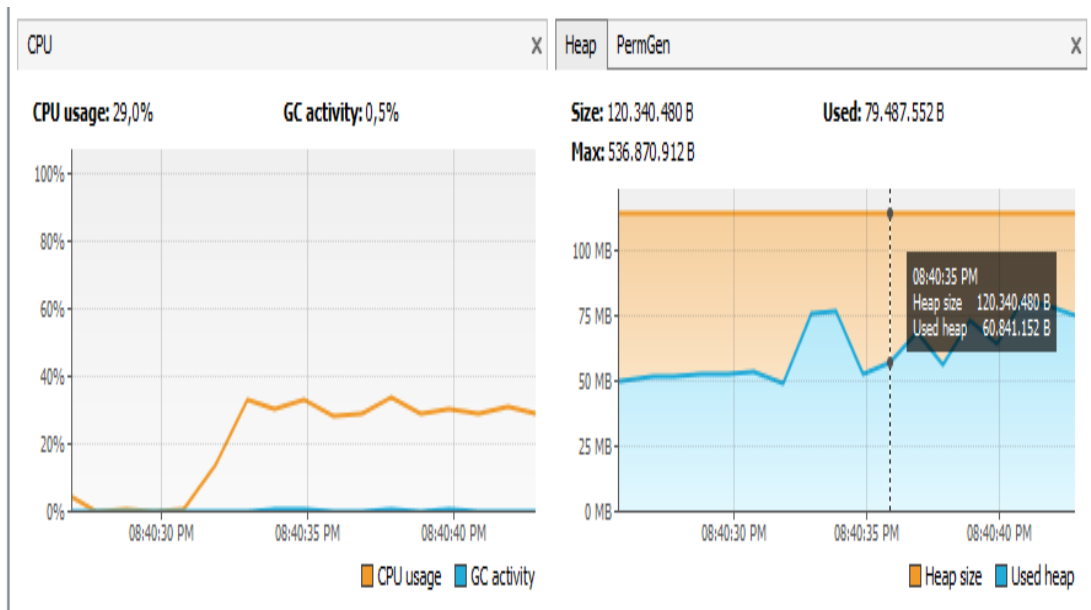
Prueba6



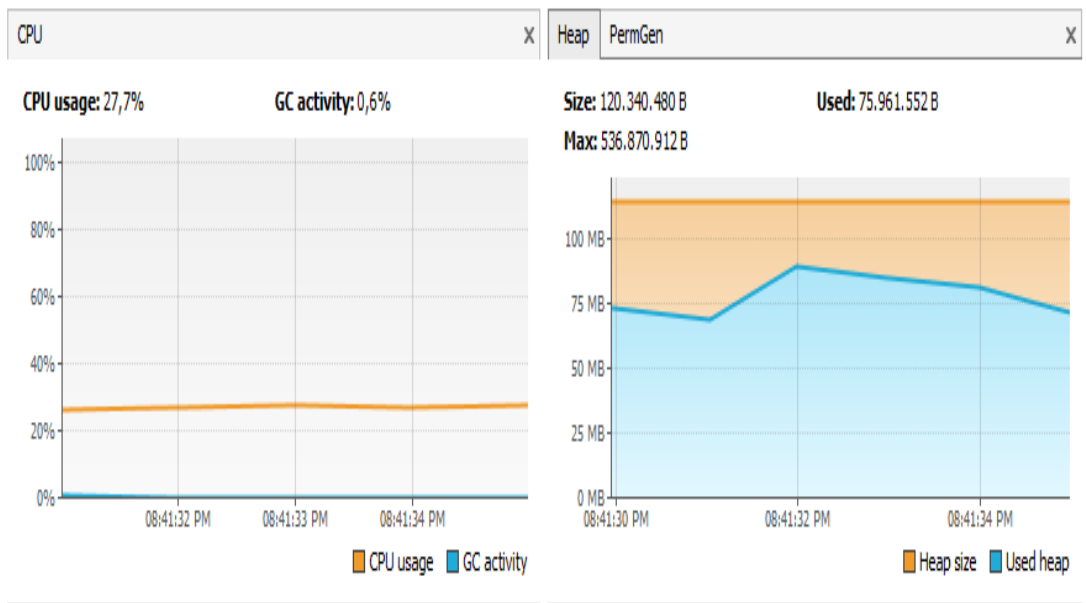
Prueba7



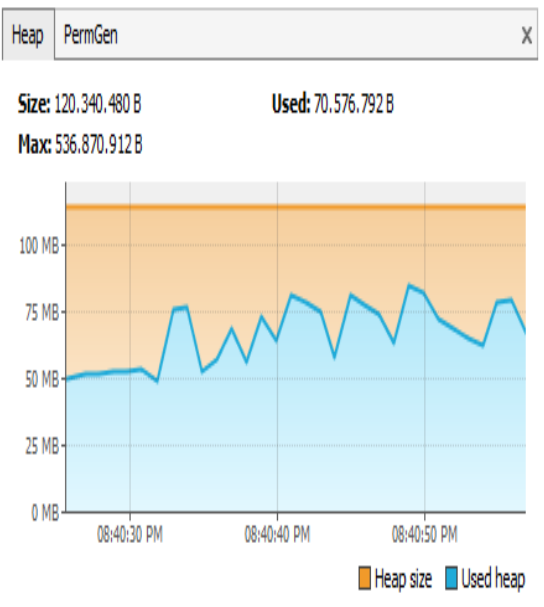
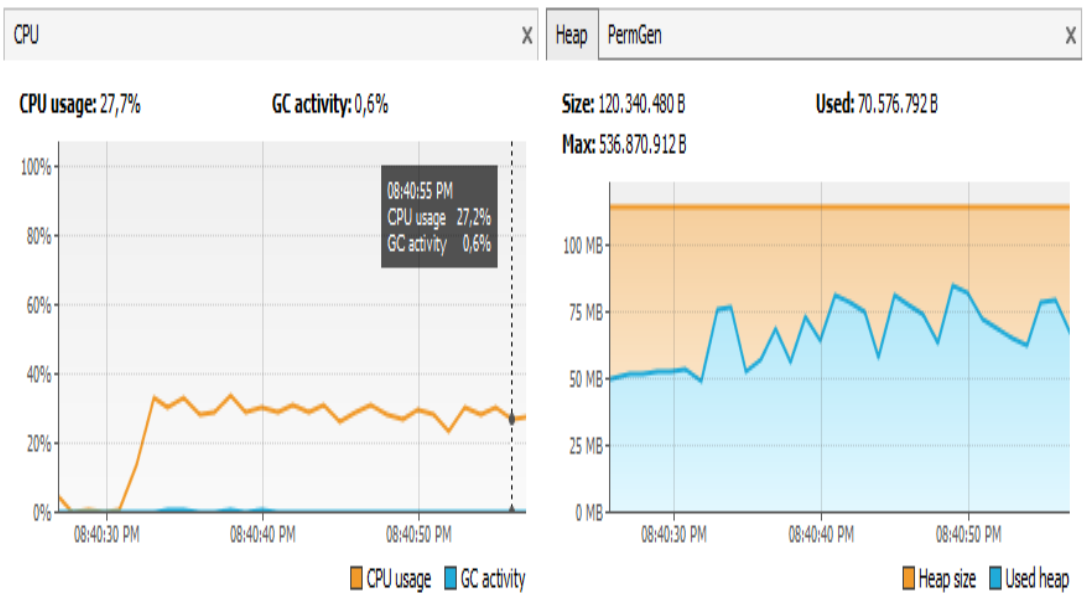
Prueba8



Prueba9

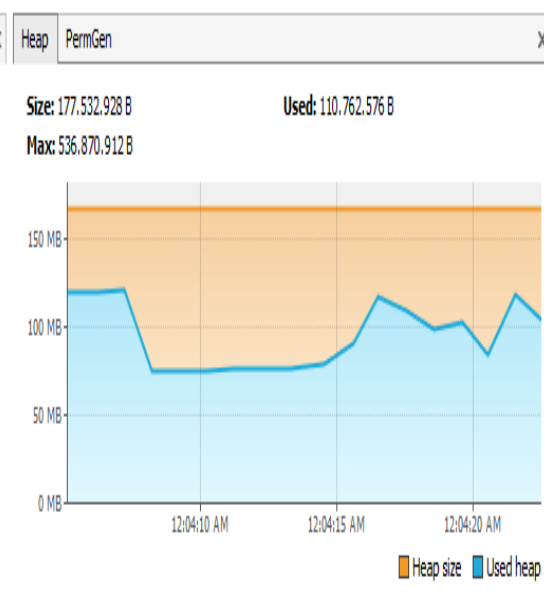
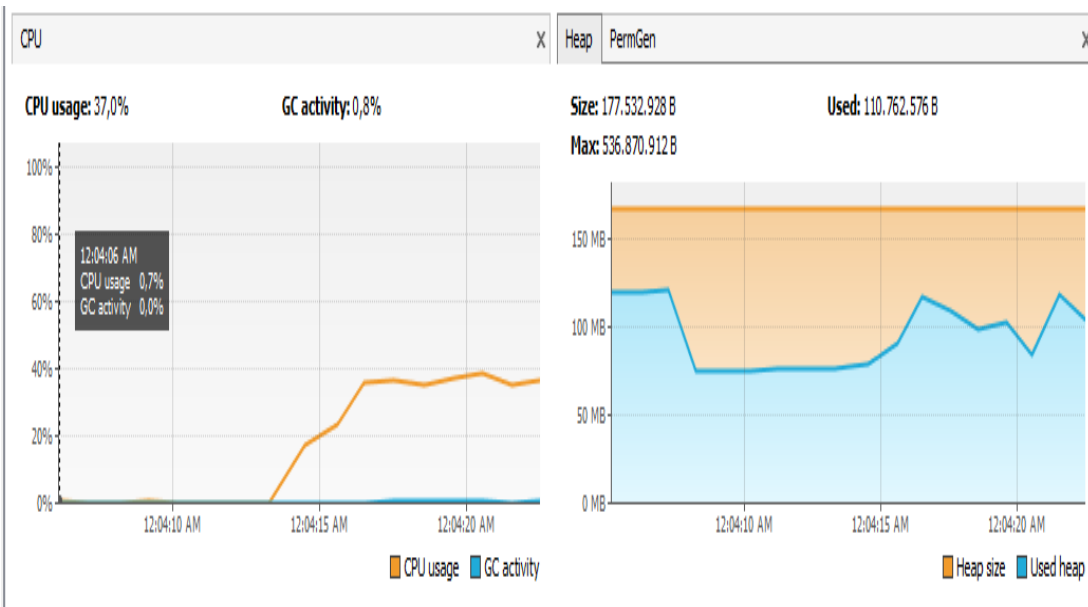


Prueba10

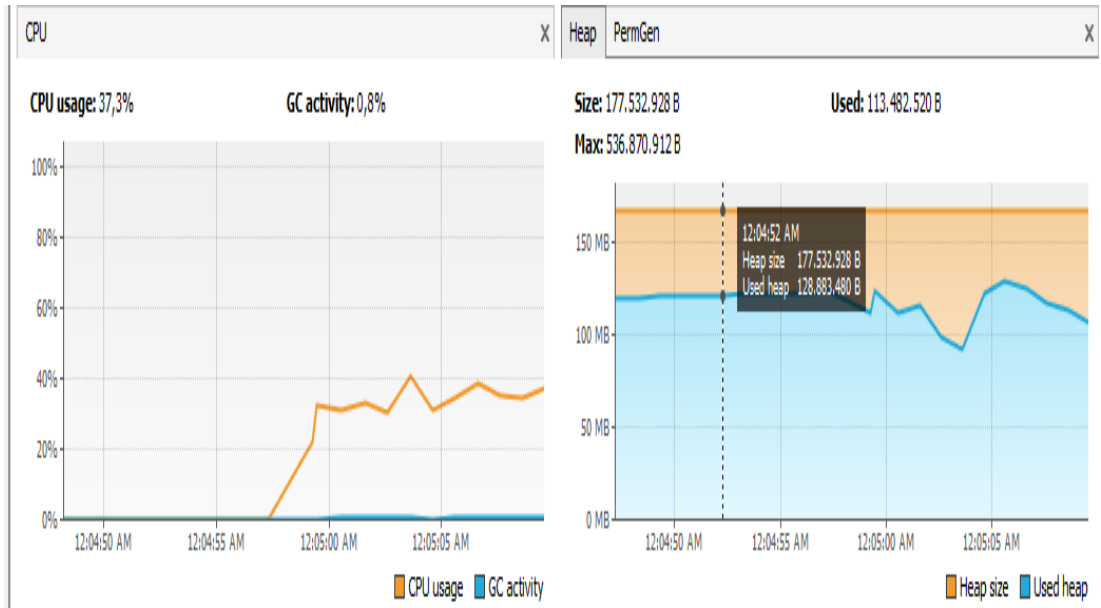


AXIS2

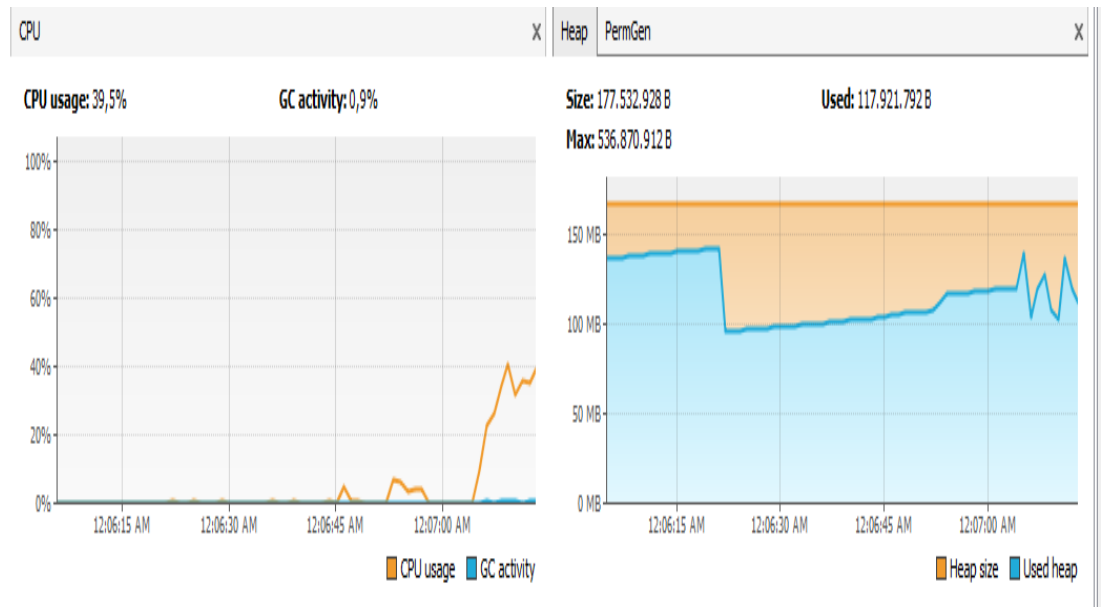
Prueba1



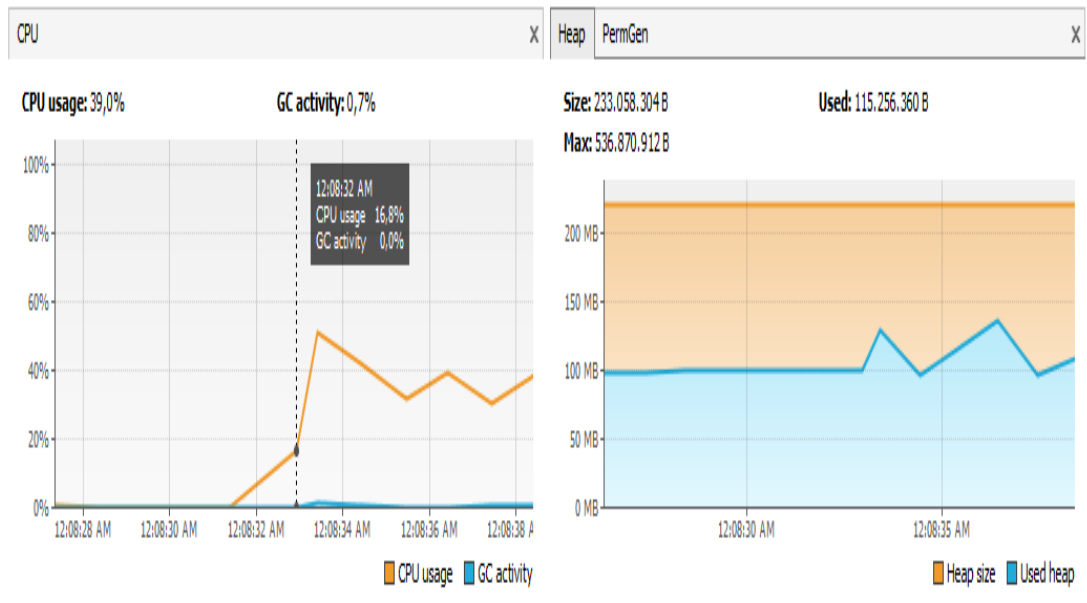
Prueba2



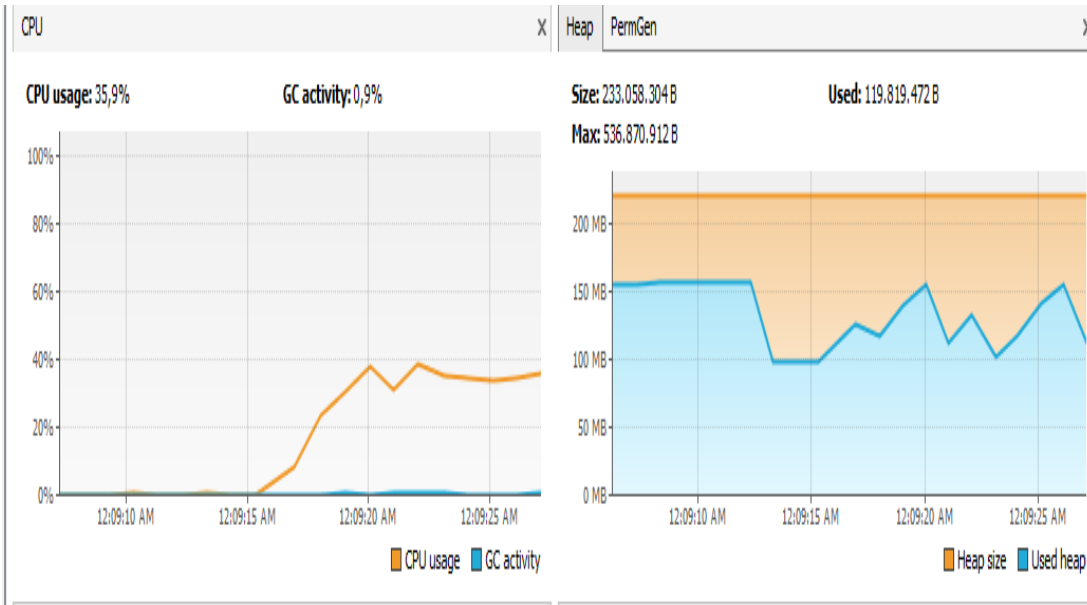
Prueba3



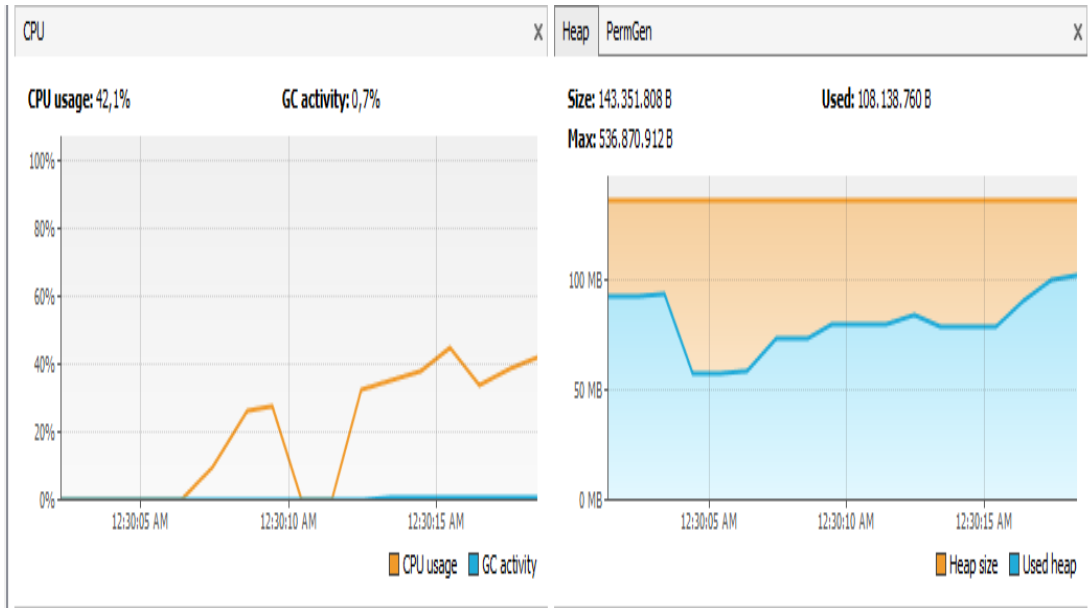
Prueba4



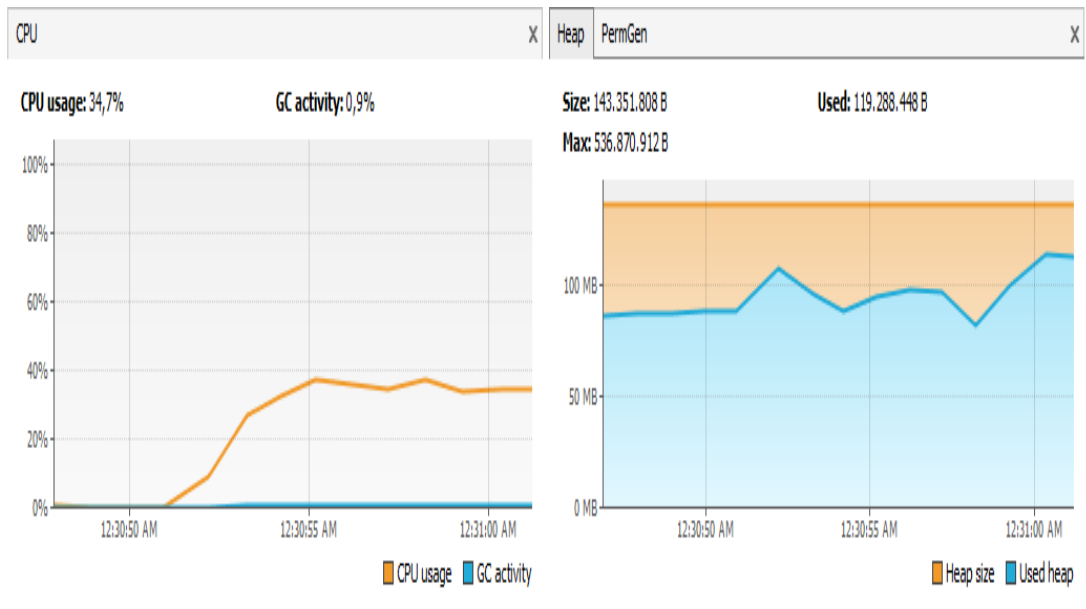
Prueba5



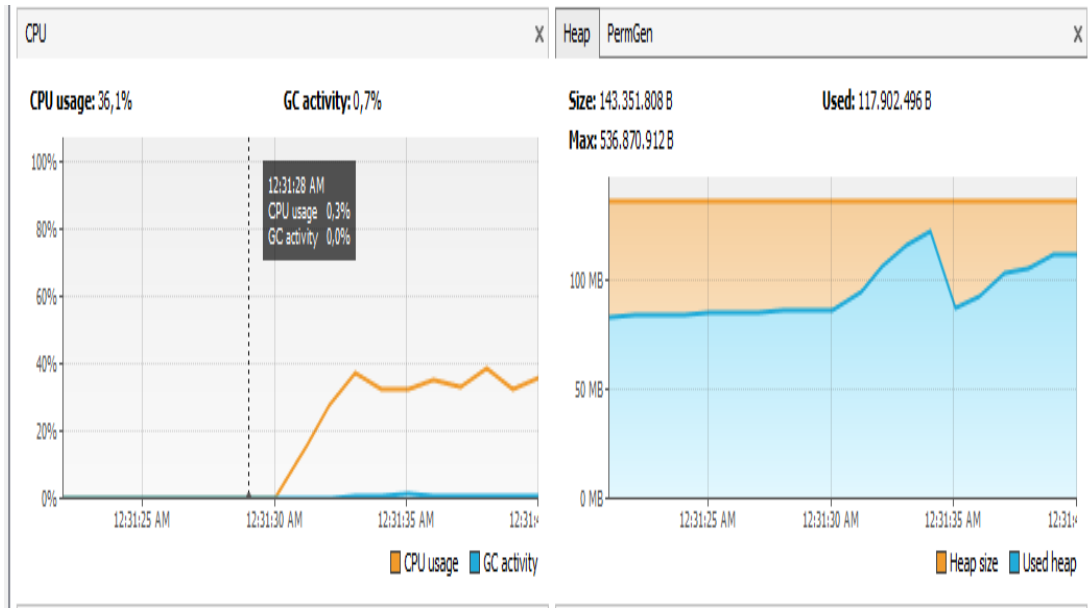
Prueba6



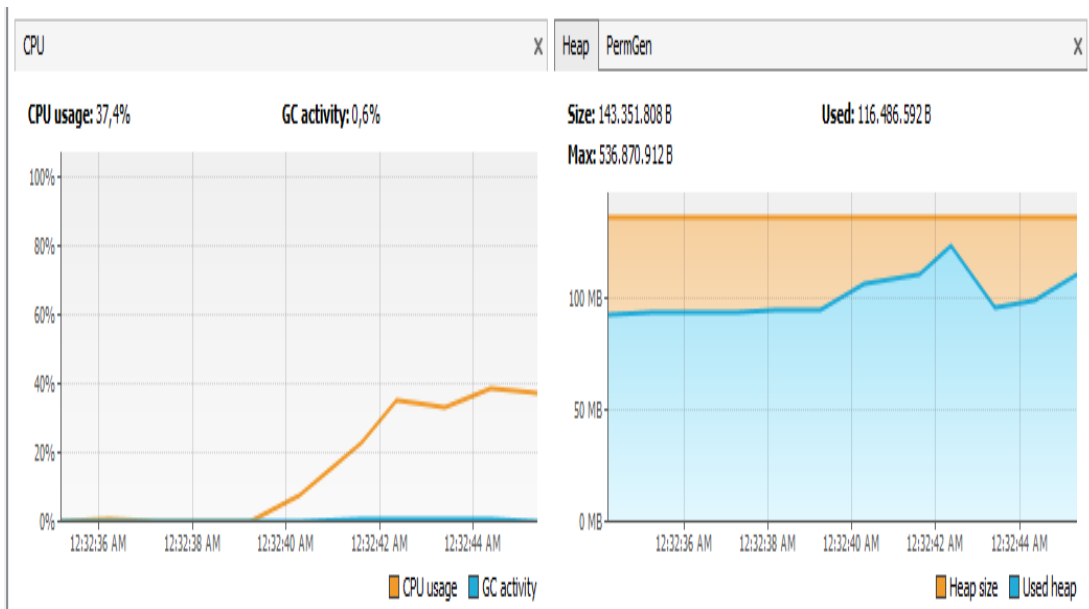
Prueba7



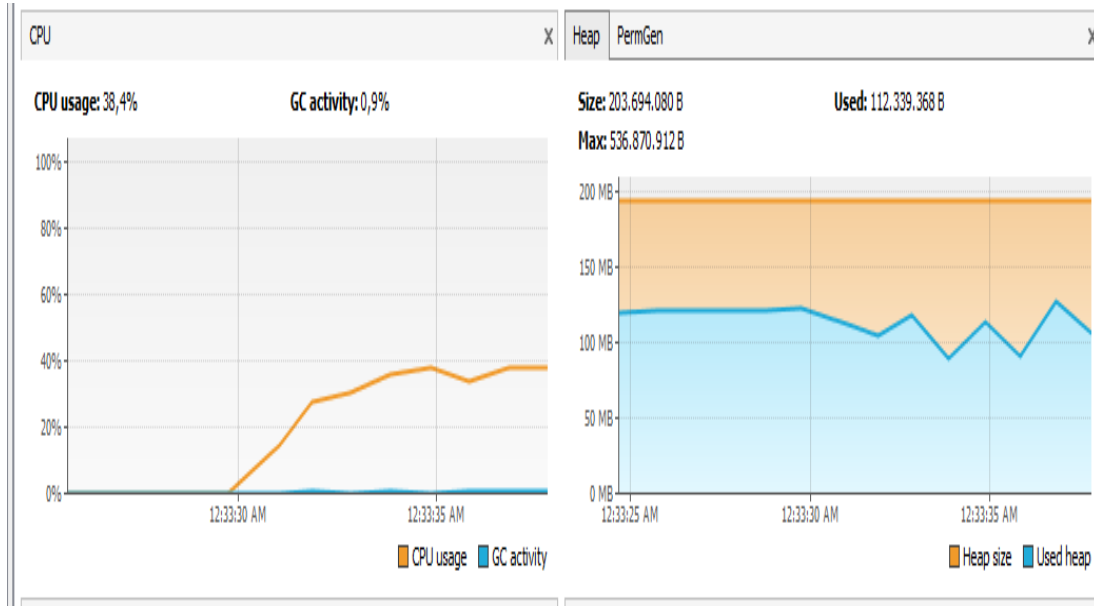
Prueba8



Prueba9

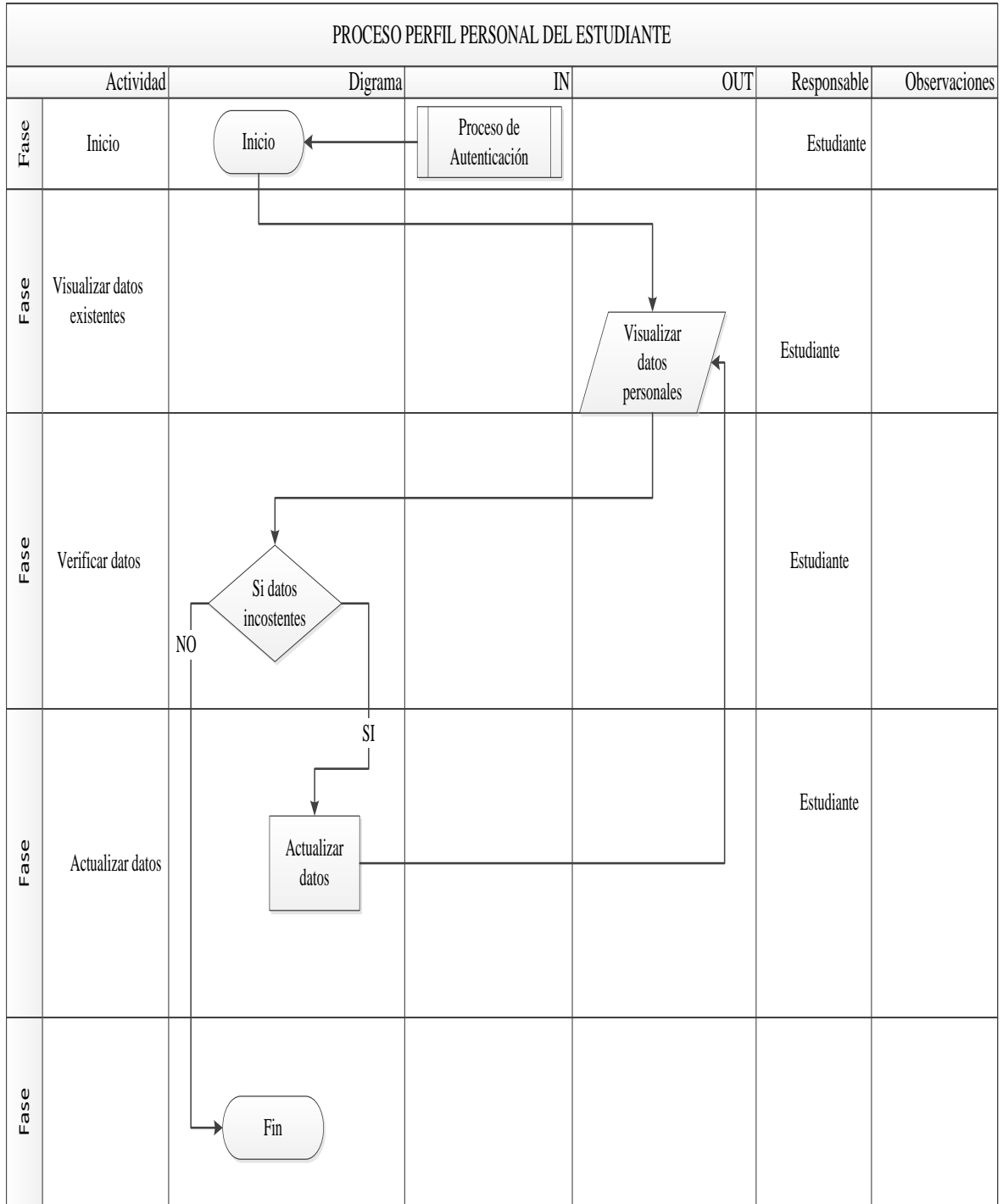


Prueba10









Anexo3.- Flujo de Procesos

Proceso perfil personal



Proceso perfil de discapacidad

PROCESO PERFIL DE DISCAPACIDAD DEL ESTUDIANTE						
	Actividad	Digrama	IN	OUT	Responsable	Observaciones
Fase	Inicio				Estudiante	
Fase	Visualizar datos existentes				Estudiante	
Fase	Verificar datos				Estudiante	
Fase	Actualizar datos				Estudiante	
Fase						

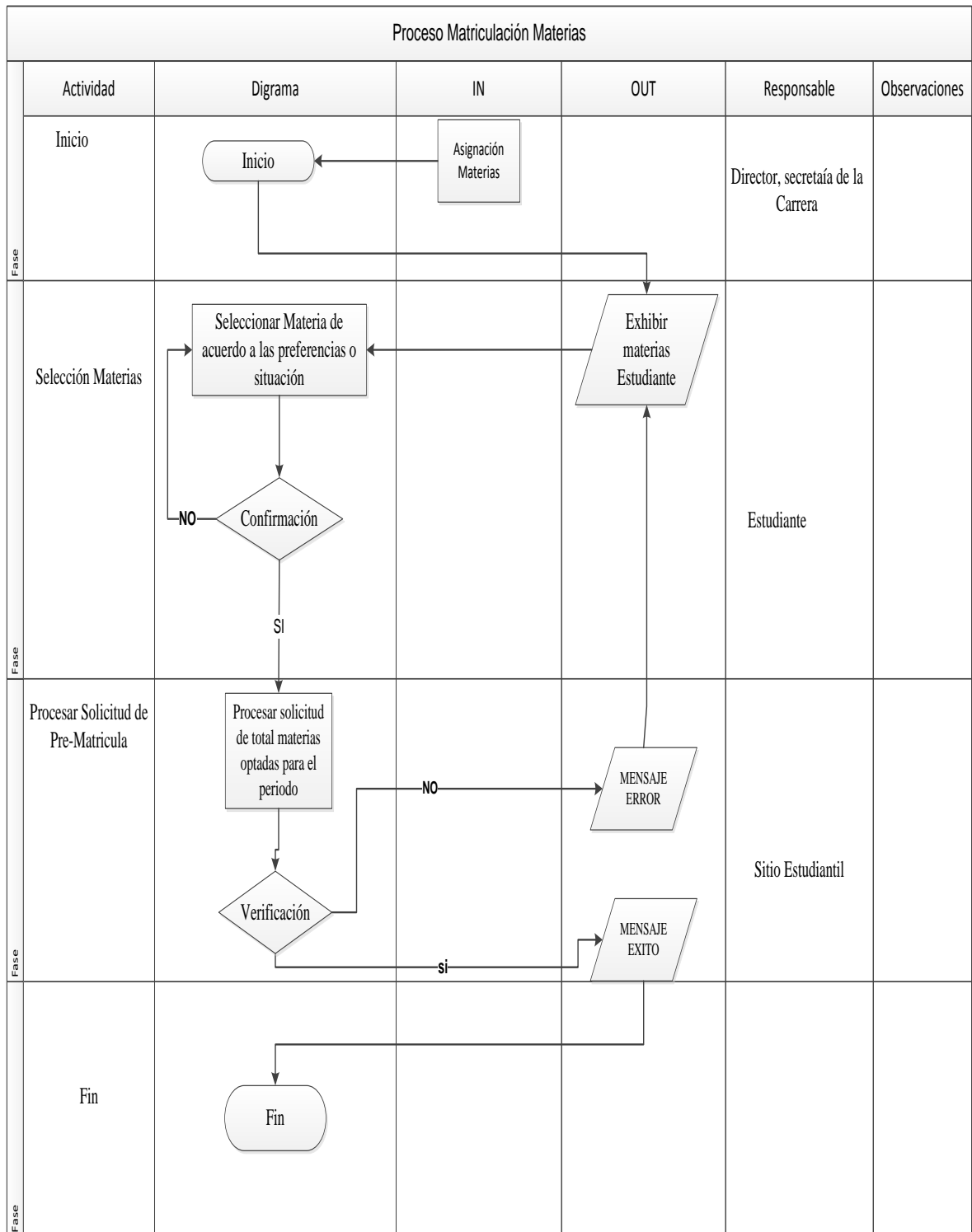
Proceso horario clase estudiante

PROCESO HORARIO CLASE ESTUDIANTE						
Actividades	Diagrama	IN	OUT	Responsable	Observaciones	
Fase Inicio	<pre> graph TD Inicio([Inicio]) --> Establecer[Establecer horarios de clase] Establecer --> Matriculacion[Matriculación] Matriculacion --> Autenticacion[autenticación] Autenticacion --> Inicio </pre>			Director de escuela y estudiante		
Fase Visualización horario de clase			<pre> graph TD Visualizar[/Visualizar horario de clase/] </pre>	Estudiante		
Fase Visualización de horario en formato detalle	<pre> graph TD Imprimir{Imprimir horario clase} Imprimir -- NO --> Fin([Fin]) Imprimir -- SI --> ImprimirClase[Imprimir horario de clase] </pre>			Estudiante		
Fase Impresión horario	<pre> graph TD ImprimirClase[Imprimir horario de clase] </pre>		<pre> graph TD HorarioImpreso[Horario de clase impreso] </pre>	Estudiante		
Fase Fin	<pre> graph TD Fin([Fin]) </pre>					

Proceso horario examen estudiante

PROCESO HORARIO DE EXAMENES DEL ESTUDIANTE						
Actividades	Diagrama	IN	OUT	Responsable	Observaciones	
Fase Inicio	<pre> graph TD Inicio([Inicio]) --> Establecer[Establecer horarios de clase] Establecer --> Matriculacion[Matriculación] Matriculacion --> Autenticacion[Autenticación] Autenticacion --> Inicio </pre>			Director de escuela y estudiante		
Fase Visualización horario de exámenes			<pre> graph TD Inicio([Inicio]) --> Horario[Horario de exámenes] </pre>	Estudiante		
Fase Visualización de horario de exámenes en formato detalle	<pre> graph TD Inicio([Inicio]) --> Imprimir{Imprimir horario exámenes} </pre>			Estudiante		
Fase Impresión horario de exámenes	<pre> graph TD Imprimir{Imprimir horario exámenes} -- SI --> Imprimir[Imprimir horario de exámenes] </pre>		<pre> graph TD Imprimir[Imprimir horario de exámenes] --> Impreso[Horario de exámenes impreso] </pre>	Estudiante		
Fase Fin	<pre> graph TD Impreso[Horario de exámenes impreso] --> Fin([Fin]) </pre>					

Proceso matriculación materias



Proceso notas estudiante

PROCESO NOTAS ESTUDIANTE						
Actividades	Diagrama	IN	OUT	Responsable	Observaciones	
Fase Inicio	<pre> graph TD Inicio([Inicio]) --> Autenticación[Autenticación] Autenticación --> Proceso[Proceso de pasado de notas] Proceso --> Matriculación[Matriculación] </pre>			Director de escuela estudiante y docente		
Fase Visualización notas				Estudiante		
Fase Visualización notas formato detalle	<pre> graph TD VerDetalle{Ver detalle de notas} -- SI --> NotasDetalle[/Notas en detalle/] VerDetalle -- NO --> Imprimir{Imprimir notas} Imprimir -- SI --> NotasAcademicas[Notas Académicas impresas] Imprimir -- NO --> Fin([Fin]) </pre>			Estudiante		
Fase Impresión notas	<pre> graph TD Imprimir[Imprimir notas] </pre>			Estudiante		
Fase Fin	<pre> graph TD Fin([Fin]) </pre>					

Anexo4.- Historias de Usuario

HISTORIA DE USUARIO 2	
Numero: 2	Usuario: Estudiante
Nombre Historia: Perfil personal del estudiante	Iteración Asignada: 1
Prioridad en Negocio: Medio	Riesgo en Desarrollo: Alto
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante procede visualizar sus datos personales.	
Observaciones: Datos personales comunes a la gran mayoría de personas.	

HISTORIA DE USUARIO 3	
Numero: 3	Usuario: Estudiante
Nombre Historia: Actualizar datos contacto perfil.	Iteración Asignada: 2
Prioridad en Negocio: Medio	Riesgo en Desarrollo: Medio
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante modificar los datos de contacto en el perfil personal del estudiante.	
Observaciones: Adicional a la información de contacto se modifica la ciudad y la parroquia.	

HISTORIA DE USUARIO 4	
Numero: 4	Usuario: Estudiante
Nombre Historia: Perfil de discapacidad del estudiante	Iteración Asignada: 2
Prioridad en Negocio: Medio	Riesgo en Desarrollo: Medio
Programador Responsable: Luis Patiño	

Descripción: El usuario estudiante procede visualizar sus datos de discapacidad.
Observaciones: Datos de discapacidad, lo poseen las personas con este tipo de capacidades

HISTORIA DE USUARIO 5	
Numero: 5	Usuario: Estudiante
Nombre Historia: Inscripciones del estudiante.	Iteración Asignada: 2
Prioridad en Negocio: Medio	Riesgo en Desarrollo: Medio
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante procede visualizar sus inscripciones realizadas en las unidades académicas de las ESPOCH.	
Observaciones: Los datos de inscripción hacen referencia las fechas en las cuales se inscribió el estudiante en las unidades académicas.	

HISTORIA DE USUARIO 6	
Numero: 6	Usuario: Estudiante
Nombre Historia: Periodos optados en una unidad académica.	Iteración Asignada: 3
Prioridad en Negocio: Medio	Riesgo en Desarrollo: Medio
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante procede visualizar sus periodos optados en una determinada unidad académica.	
Observaciones: Los periodos se visualizan según la unidad académica.	

HISTORIA DE USUARIO 7	
Numero: 7	Usuario: Estudiante

Nombre Historia: Asignaturas optadas en un periodo optado.	Iteración Asignada: 3
Prioridad en Negocio: Medio	Riesgo en Desarrollo: Alto
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante procede seleccionar un periodo y se visualiza las asignaturas optadas en este periodo.	
Observaciones: Las asignaturas dependen tanto del periodo como de la unidad academia.	

HISTORIA DE USUARIO 8	
Numero 8	Usuario: Estudiante
Nombre Historia: Títulos pre-grado	Iteración Asignada: 3
Prioridad en Negocio: bajo	Riesgo en Desarrollo: bajo
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante procede visualizar sus títulos de pre-grado	
Observaciones:	

HISTORIA DE USUARIO 9	
Numero:9	Usuario: Estudiante
Nombre Historia: Notas de cada asignatura.	Iteración Asignada: 4
Prioridad en Negocio: Alto	Riesgo en Desarrollo: Alto
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante procede visualizar sus notas en las asignaturas seleccionado un determinado periodo académico.	
Observaciones: El desglose de notas incluye valores cuantitativos y cualitativos.	

HISTORIA DE USUARIO 10	
Numero: 10	Usuario: Estudiante
Nombre Historia: Notas parciales.	Iteración Asignada: 4
Prioridad en Negocio: Alto	Riesgo en Desarrollo: medio
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante procede a seleccionar la nota acumulada para visualizar sus notas parciales	
Observaciones: La nota acumulado es la suma de las tres notas parciales.	

HISTORIA DE USUARIO 11	
Numero: 11	Usuario: Estudiante
Nombre Historia: Horario de clase del estudiante	Iteración Asignada: 5
Prioridad en Negocio: Alto	Riesgo en Desarrollo: Alto
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante procede a visualizar su horario de clase definido según las asignaturas optadas.	
Observaciones: El horario de clase del estudiante pertenece al periodo activo y unidad académica en cuestión.	

HISTORIA DE USUARIO 12	
Numero: 12	Usuario: Estudiante
Nombre Historia: Horarios de clase por paralelo	Iteración Asignada: 5
Prioridad en Negocio: Alto	Riesgo en Desarrollo: Alto
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante procede a visualizar la lista de horarios de clase	

de la unidad académica agrupados por nivel y paralelo.
Observaciones: Los horarios de clase por paralelo son vitales a la hora de matricularse.

HISTORIA DE USUARIO 13	
Numero: 13	Usuario: Estudiante
Nombre Historia: Horarios de clase por materia	Iteración Asignada: 5
Prioridad en Negocio: Alto	Riesgo en Desarrollo: Alto
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante procede a visualizar la lista de horarios de clase de la unidad académica agrupados por materia.	
Observaciones: Estos horarios ayudaran a seleccionar un paralelo en un centro de apoyo.	

HISTORIA DE USUARIO 14	
Numero: 14	Usuario: Estudiante
Nombre Historia: Horarios de exámenes	Iteración Asignada: 5
Prioridad en Negocio: Medio	Riesgo en Desarrollo: Alto
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante procede a visualizar la lista de horarios de exámenes en el periodo de los mismos	
Observaciones: Únicamente se visualizaran en periodo de exámenes.	

HISTORIA DE USUARIO 15	
Numero: 15	Usuario: Estudiante
Nombre Historia: Inscribirse centros de	Iteración Asignada: 5

apoyo.	
Prioridad en Negocio: Bajo	Riesgo en Desarrollo: Bajo
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante procede seleccionar cada uno de los centros de apoyo para obtener su código respectivo.	
Observaciones: La inscripción en el centro de apoyo se lo realizara únicamente una vez en cada uno.	

HISTORIA DE USUARIO 16	
Numero: 16	Usuario: Estudiante
Nombre Historia: Noticias estudiantiles	Iteración Asignada: 6
Prioridad en Negocio: media	Riesgo en Desarrollo: alto
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante procede visualizar las noticias estudiantiles que se presentan al inicio del sistema.	
Observaciones:	

HISTORIA DE USUARIO 17	
Numero: 17	Usuario: Estudiante
Nombre Historia: Pre-matricula en materias	Iteración Asignada: 6
Prioridad en Negocio: Alto	Riesgo en Desarrollo: Alto
Programador Responsable: Luis Patiño	
Descripción: El usuario estudiante procede cargar su propuesta de matrícula, selecciona las materias obligatorias y/u optativas, y por ultimo solicita la matricula. Todo esto en el periodo de matrículas.	
Observaciones: Inicialmente el sistema carga una propuesta de materias que el estudiante puede optar.	

Anexo5. Diccionario de datos

Base de datos OASIS-Sitio Estudiantil

Tabla Dias

Atributo	Descripción	Tipo	PK	NULL	FK
strCodigo	Identifica a Dias	varchar(10)	Si	NOT NULL	
strDescripcion	Descripción de Dias	varchar(10)		NULL	
bytOrden	Orden de los Dias	binary(10)		NULL	

Tabla Horas

Atributo	Descripción	Tipo	PK	NULL	FK
strCodigo	Identifica a Horas	varchar(10)	Si	NOT NULL	
strDescripcion	Descripción de Horas	varchar(10)		NULL	
dtHoraInic	Fecha de inicio de la Hora	datetime		NULL	
dtHoraFin	Fecha de fin de la Hora	datetime		NULL	

Tabla Horas

Atributo	Descripción	Tipo	PK	NULL	FK
strCodDia	Identifica a Horarios	varchar(10)	Si	NOT NULL	Si
strCodHora	Identifica a Horarios	varchar(10)	Si	NOT NULL	Si

Tabla Niveles

Atributo	Descripción	Tipo	PK	NULL	FK
strCodigo	Identifica a Nivel	varchar(10)	Si	NOT NULL	
strDescripcion	Nombre de Nivel	varchar(10)		NULL	

Tabla Curso

Atributo	Descripción	Tipo	PK	NULL	FK
strCodParalelo	Identifica a Curso	varchar(10)	Si	NOT NULL	
strCodNivel	Identificador de Nivel	varchar(10)	SI	NOT NULL	SI
strCodPeriodo	Código del Periodo Asignado	varchar(10)		NOT NULL	SI
strAula	Nombre del Aula	varchar(100)		NULL	
strUbicacion	Descripción del lugar del aula	varchar(100)		NULL	

Tabla Materias

Atributo	Descripción	Tipo	PK	NULL	FK
strCodigo	Identifica a Mterias	varchar(10)	Si	NOT NULL	
strNombre	Descripción de Materias	varchar(10)	SI	NOT NULL	SI
dtFechaCreada	Fecha de la creación de la Materia	varchar(10)		NOT NULL	SI
detFechaElim	Nombre del Aula	varchar(100)		NULL	
blnActiva	Descripción del lugar del aula	varchar(100)		NULL	

Tabla Materias_Asignadas

Atributo	Descripción	Tipo	PK	NULL	FK
sintCodMatricula	Identifica a Materias_Asignadas	varchar(10)	Si	NOT NULL	SI
strCodPeriodo	Código de Periodo	varchar(10)	SI	NOT NULL	SI
strCodMateria	Codigo de Materia	varchar(10)	SI	NOT NULL	SI
byNumMat	Numero Matricula en la Materia	varchar(100)		NULL	
byAsistencia	Asistencia a la Materia	varchar(100)		NULL	

strObservaciones	Notas adicionales	varchar(100)		NULL	
strCodParalelo	Código del Paralelo	varchar(10)		NOT NULL	Si
strCodNivel	Código del Nivel	varchar(10)		NOT NULL	Si
strCodEstud	Codigo del Estudiante	varchar(10)		NOT NULL	Si

Tabla Horario_clases

Atributo	Descripción	Tipo	PK	NULL	FK
strCodigoParalelo	Identifica a Horario_clases	varchar(10)	Si	NOT NULL	SI
strCodoPeriodo	Identifica a Horario_clases	varchar(10)	SI	NOT NULL	SI
strCodNivel	Identifica a Horario_clases	varchar(10)	SI	NOT NULL	SI
strCodMateria	Identifica a Horario_clases	varchar(10)		NULL	
byAsistencia	Identifica a Horario_clases	varchar(10)		NULL	
strCodDia	Identifica a Horario_clases	varchar(10)		NULL	
strCodHora	Identifica a Horario_clases	varchar(10)		NOT NULL	Si

Tabla Dictado_Materias

Atributo	Descripción	Tipo	PK	NULL	FK
strCodParalelo	Identifica a dictado_Materias	varchar(10)	SI	NOT NULL	SI
strCodPeriodo	Identifica a dictado_Materias	varchar(10)	SI	NOT NULL	SI
strCodNivel	Identifica a dictado_Materias	varchar(10)	SI	NOT NULL	SI
strCodMateria	Identifica a dictado_Materias	varchar(10)	SI	NOT NULL	SI
strCodDocente	Identifica a dictado_Materias	varchar(10)	SI	NOT NULL	SI
dtFechaExPrinc	Fecha Examen Principal	datetime		NULL	
dtFechaExSusp	Fecha Examen Suspensión	datetime		NULL	
dtFechaEntPrinc	Entrega notas Principal	datetime		NULL	

dtFechaEntSusp	Entrega notas Suspensión	datetime		NULL	
dtFechaAsentExPrinc	Fecha asentar notas Principal	datetime		NULL	
dtFechaAsentExSusp	Fecha asentar notas Suspensión	datetime		NULL	
dtFechaEntEvaluacion	Fecha de evaluaciones	datetime		NULL	

Tabla Notas_Examenes

Atributo	Descripción	Tipo	PK	NULL	FK
sintCodMatricula	Identifica a Notas_Examenes	varchar(10)	Si	NOT NULL	SI
strCodPeriodo	Identifica a Notas_Examenes	varchar(10)	SI	NOT NULL	SI
strCodMateria	Identifica a Notas_Examenes	varchar(10)	SI	NOT NULL	SI
strCodTipoExamen	Identifica a Notas_Examenes	varchar(10)	SI	NOT NULL	
bytAcumulado	Nota del acumulado	binary(10)		NULL	
stbytNota	Nota del examen principal	binary(10)		NULL	
strCodEquiv	Código de la Equivalencia	varchar(10)		NOT NULL	Si
strObservaciones	Notas Adicionales	varchar(100)		NULL	

Tabla Matriculas

Atributo	Descripción	Tipo	PK	NULL	FK
sintCodigo	Identifica a Matriculas	binary(10)	SI	NOT NULL	
strCodPeriodo	Identifica a Matriculas	varchar(10)	SI	NOT NULL	SI
strCodEstud	Identifica a Matriculas	binary(10)	SI	NOT NULL	SI
strCodNivel	Código del Nivel	varchar(10)		NULL	SI
strAutorizadaPor	Agente que autoriza la	varchar(10)		NULL	

	matricula				
dtFechaAutorizada	Fecha de autorización	datetime		NULL	
strCreadaPor	Agente que crea la matricula	datetime		NULL	
dtFechaCreada	Fecha de la creación de la matricula	datetime		NULL	
strCodEstado	Estado actual de la matricula	datetime		NULL	
strObservaciones	Notas Adicionales	datetime		NULL	

Tabla Estudiante

Atributo	Descripción	Tipo	PK	NULL	FK
strCodigo	Identifica a Matriculas	binary(10)	SI	NOT NULL	I
strCedula	Identifica a Matriculas	varchar(15)		NOT NULL	
strNombres	Identifica a Matriculas	varchar(100)		NOT NULL	
strApellidos	Código del Nivel	varchar(100)		NULL	
strCedulaMil	Agente que autoriza la matricula	varchar(100)		NULL	
dtFechaNac	Fecha de autorización	datetime		NULL	
dtFechaIng	Agente que crea la matricula	datetime		NULL	
strEmail	Fecha de la creación de la matricula	varchar(10)		NULL	
strNacionalidad	Estado actual de la matricula	varchar(10)		NULL	
strDocumentacion	Notas Adicionales	varchar(10)		NULL	
strCodSexo		varchar(10)		NOT NULL	
strCodTit		varchar(10)		NOT NULL	
strCodInt		varchar(10)		NOT NULL	
strFormaIns		varchar(10)		NULL	

Base de datos Central-Sitio Estudiantil

Tabla persona

Atributo	Descripción	Tipo	PK	NULL	FK
per_id	Identifica a Persona	int	Si	NOT NULL	
per_nombres	Nombre de la Persona	varchar(100)		NULL	
per_primerApellido	Primer apellido de la Persona	varchar(100)		NULL	
per_segundoApellido	Segundo Apellido de la Apersona	varchar(100)		NULL	
per_email	Correo electrónico	varchar(100)		NULL	
per_emailAlternativo	Correo electrónico alternativo	varchar(100)		NULL	
per_telefonoOficina	Número de teléfono de la Oficina	varchar(10)		NULL	
per_telefonoCelular	Número de teléfono del Móvil	varchar(10)		NULL	
per_fechaNacimiento	Fecha de nacimiento	datetime		NULL	
per_afiliacionIESS	Afiliación al IESS	varchar(100)		NULL	
eci_id		int		NOT NULL	SI
gen_id	Identificador de Genero	int		NOT NULL	SI
prq_id	Identificador de Parroquia	int		NOT NULL	SI
per_tieneCapacidadesDiferentes	Posee capacidad especial	varchar(100)			
per_creadoPor	Agente registrador de la Persona	varchar(100)			
per_fechaCreacion	Fecha de creación	datetime			
per_modificadoPor	Agente modificador de	varchar(100)			

Atributo	Descripción	Tipo	PK	NULL	FK
	datos				
per_fechaModificacion	Fecha de modificación	datetime			
per_telefonoCasa	Número de teléfono de casa.	varchar(10)			
etn_id	Identificador de la Etnia	int		NOT NULL	SI
tsa_id	Identificador del tipo de sangre	int		NOT NULL	SI

Tabla discapacidad

Atributo	Descripción	Tipo	PK	NULL	FK
dis_id	Identifica a Discapacidad	int	Si	NOT NULL	
dis_valor	Valor de discapacidad	int		NOT NULL	SI
cad_id	Identificador de Causa de Discapacidad	int		NOT NULL	SI
cdi_id	Identificador de Carnet de Discapacidad	int		NOT NULL	SI
mdi_id	Identificador de Medida de Discapacidad	int		NULL	
dis_observacion	Notas adicionales	varchar(100)			

Tabla carnetDiscapacidad

Atributo	Descripción	Tipo	PK	NULL	FK
cdi_id	Identifica a carnetDiscapacidad	int	Si	NOT NULL	
cdi_numero	Número del carnetDiscapacidad	int		NOT NULL	
org_id	Identificador de la Organización	int		NOT NULL	SI
per_id	Identificador de la Persona	int		NOT NULL	SI

Tabla documentoPersonal

Atributo	Descripción	Tipo	PK	NULL	FK
pid_id	Identifica a documentoPersonal	int	Si	NOT NULL	
pid_valor	Valor del documentoPersonal	varchar(20)		NOT NULL	
tdi_id	Identificador del TipoDocumento	int		NOT NULL	SI
per_id	Identificador de la Persona	int		NOT NULL	SI
pid_activo	Estado del documentoPersonal	binary(10)			

Tabla tipoDocumentoID

Atributo	Descripción	Tipo	PK	NULL	FK
tdi_id	Identifica a tipoDocumentoID	int	Si	NOT NULL	
tdi_nombre	Nombre del tipoDocumentoID	varchar(100)		NOT NULL	

Tabla tipoSangre

Atributo	Descripción	Tipo	PK	NULL	FK
tsa_id	Identifica a tipoSangre	int	Si	NOT NULL	
tsa_nombre	Nombre del tipoSangre	varchar(100)		NOT NULL	

Tabla etnia

Atributo	Descripción	Tipo	PK	NULL	FK
etn_id	Identifica a etnia	int	Si	NOT NULL	
etn_codigo	Código del etnia	varchar(10)		NOT NULL	
etn_nombre	Nombre del etnia	varchar(100)			

Tabla plurinacionalidad

Atributo	Descripción	Tipo	PK	NULL	FK
plu_id	Identifica a etnia	int	Si	NOT NULL	
plu_nombre	Nombre de la plurinacionalidad	varchar(10)		NOT NUL	
tpl_id	Identifica al tipoPlurinacionalidad	int		NOT NULL	SI
etn_id	Identifica a la etnia	int		NOT NULL	SI
plu_ubicacion	Ubicación de la plurinacionalidad	varchar(100)			

Tabla cuenta

Atributo	Descripción	Tipo	PK	NULL	FK
cta_id	Identifica a cuenta	Int	Si	NOT NULL	
cta_login	Nombre de usuario	varchar(10)		NOT NUL	
cta_password	Clave del usuario	varchar(100)		NULL	
cta_CodigoSalt	Código de Salto	varchar(100)		NULL	
cta_Token	Token de la cuenta	varchar(100)		NULL	
cta_activo	Estado de cuenta	binary(10)		NULL	
per_id	Identificador de la Persona	int		NOT NUL	SI

Tabla direccion

Atributo	Descripción	Tipo	PK	NULL	FK
dir_id	Identifica a dirección	Int	Si	NOT NULL	
dir_callePrincipal	Nombre de la calle principal	varchar(10)		NOT NUL	
dir_numero	Numero de casa	varchar(100)		NULL	
dir_referencia	Referencia de algo muy conocido	varchar(100)		NULL	
dir_activa	Estado de dirección.	binary(10)		NULL	
prq_id	Identificador de Parroquia	int		NOT NUL	SI
per_id	Identificador de la Persona	int		NOT NUL	SI

Tabla parroquia

Atributo	Descripción	Tipo	PK	NULL	FK
prq_id	Identifica a parroquia	Int	Si	NOT NULL	
prq_nombre	Nombre de la parroquia	varchar(100)		NOT NUL	
cui_id	Identificador de ciudad	int		NOT NUL	SI
prq_esUrbana	Estado urbanístico	binary(10)		NULL	

Tabla ciudad

Atributo	Descripción	Tipo	PK	NULL	FK
cui_id	Identifica a ciudad	Int	Si	NOT NULL	
cui_nombre	Nombre de la ciudad	varchar(100)		NOT NUL	
pro_id	Identificador de provincia	int		NOT NUL	SI

Tabla provincia

Atributo	Descripción	Tipo	PK	NULL	FK
pro_id	Identifica a provincia	Int	Si	NOT NULL	
por_nombre	Nombre de la provincia	varchar(100)		NOT NUL	
pai_id	Identificador de país	int		NOT NUL	SI

Tabla pais

Atributo	Descripción	Tipo	PK	NULL	FK
pai_id	Identifica a pais	Int	Si	NOT NULL	
pai_nombre	Nombre del pais	varchar(100)		NOT NUL	

Anexo6.- Manual de usuario

1. INTRODUCCIÓN

1.1 Propósito del documento

El presente manual tiene como objeto mostrar la utilización del Sitio de Información para Estudiantes de la Epoch (Sitio Estudiantil), con la finalidad de que los estudiantes puedan tener una completa comprensión de todo su contenido y funcionalidad, clasificando la información de acuerdo a cada una de los módulos que posee el mismo.

Por lo antes expuesto presentamos el siguiente Manual de Usuario como una guía indispensable para el correcto funcionamiento.

1.2 Funcionamiento General

El sitio estudiantil busca actualizar el sitio de información de estudiantes actual, en reducción de tiempos e incremento de funcionalidades. Mediante el cual los estudiantes dispondrán de toda su información, personal, discapacidad en caso de tenerla y académica, además serán capaces de realizar su pre-matricula en la diferentes unidades académicas de la institución.

1.3 Objetivos

- Dar a conocer detalladamente el manejo de la información y funcionalidad contenida en el sitio estudiantil
- Orientar a los estudiantes de la ESPOCH en el uso del sitio estudiantil.
- Brindar un soporte para capacitaciones futuras sobre el uso del sitio.

2 Generalidades del Sitio Estudiantil

El sitio estudiantil es un sistema web que se ejecuta en un explorador web o browser, dentro de un computador personal. Los exploradores recomendados son el Internet Explorer superiores al 8, Mozilla Firefox 16, Google Chrome.

2.1 Acceso al Sitio Estudiantil

Para ingresar en el sitio de información para estudiantes se debe seguir los siguientes pasos:

1. Abrir el navegador de internet,(Mozilla Firefox).
2. Digitar en la barra de direcciones estudiantil.esPOCH.edu.ec o puede acceder desde el portal institucional principal, ubicando Servicios Web, Académico-OASIS, Sitio Estudiantil.

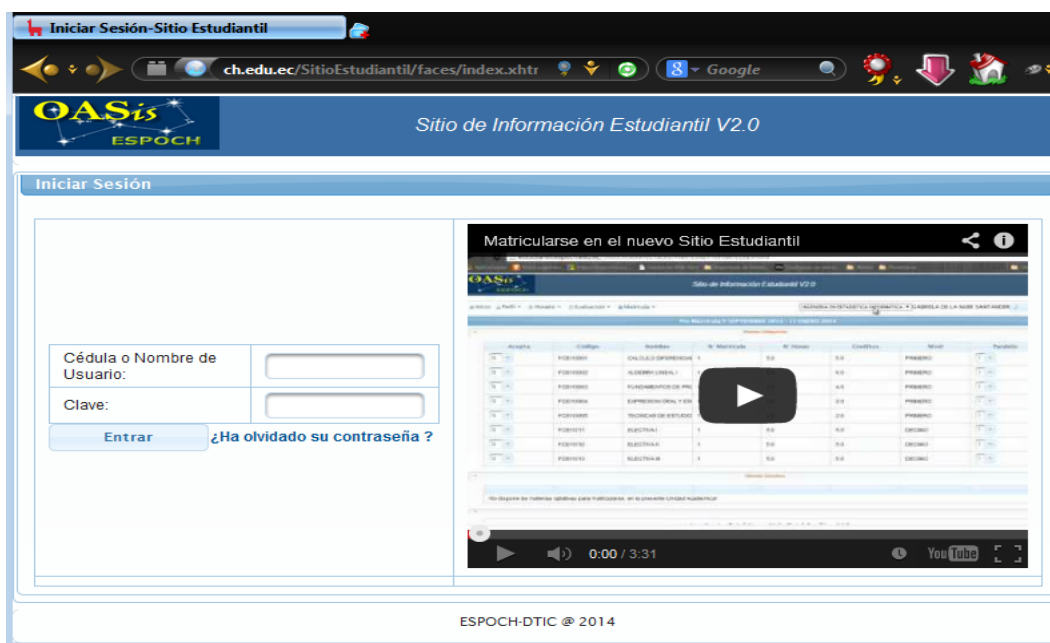



Figura1 Acceso al Sitio Estudiantil

Una vez accedido al sitio, este presentara un aspecto similar al ilustrado en la imagen anterior.

3. USO DE MÓDULOS

Dentro del sitio estudiantil se encuentran los siguientes módulos:



The screenshot shows a horizontal navigation bar with five items: 'Inicio' (Home), 'Perfil' (Profile), 'Horario' (Schedule), 'Evaluacion' (Evaluation), and 'Matricula' (Registration). Each item has a small icon and a dropdown arrow. Below the navigation bar, a list of modules is displayed, each preceded by a diamond icon. The 'Perfil' and 'Evaluación' modules have sub-items listed with circle and square icons respectively.

- ❖ Inicio
- ❖ Perfil
 - Personal
 - Discapacidad
- ❖ Horario
 - Clase
 - Examen
 - +Horarios
 - Horario Materia
 - Horario Paralelo
- ❖ Evaluación
 - Notas
 - + Notas
 - Notas Anteriores
- ❖ Matrícula

- Matrículas Realizadas
- Pre-Matrícula
- Inscripción Centros Apoyo

3.1 Página Principal

Una vez ingresado en el browser la URL se presentara la página principal del sitio con el nombre y logo del mismo. Además de visualizar los campos de ingreso de credenciales para la autenticación del estudiante (Ver Figura 2)



Figura2 Página principal del Sitio Estudiantil

3.2 Iniciar Sesión

Para iniciar sesión en el sitio estudiantil como estudiante se debe ingresar en los campos de credenciales el nombre de usuario o cédula y la clave que puede ser el mismo número de cédula pero sin el guion, una vez ingresado los credenciales se procede a presionar el botón “entrar”. Posterior al a autenticación se visualizara la página de inicio del sitio estudiantil, la cual contiene un menú superior y un lateral izquierdo con las opciones que se pueden realizar, además del centro de noticias estudiantiles (Ver Figura3).



Figura3 Página inicio

3.3 Módulo Perfil

3.3.1 Perfil Personal

Visualiza tres opciones o sub módulos: datos, inscripciones, títulos. Los datos hacen referencia a la información personal del estudiante actual, clasificada en información elemental, extendida y de contacto (Ver Figura 4).

Información Elemental	
Nombre:	LUIS GUSTAVO
Primer Apellido:	PATIÑO
Segundo Apellido:	CASTILLO
Genero:	<input type="radio"/> Femenino <input checked="" type="radio"/> Masculino
Fecha Nacimiento:	1989-10-20
Estado Civil:	SOLTERO/A
País:	ECUADOR
Provincia:	CAÑAR
Ciudad:	CAÑAR
Parroquia:	CAÑAR
Nacionalidad:	Documentos Personales

Información de Contacto	
Dirección	Teléfono
Correo Electrónico:	gus_cas@hotmail.com
Correo Electrónico Alternativo:	

Actualizar

Figura4 Página perfil personal datos

Los datos durante la migración se modifican únicamente en el lugar de nacimiento siendo los campos afectados la ciudad y parroquia de procedencia. Para contrarrestar este posible problema se ha dejado la posibilidad de actualizar los datos personales, para lo cual se debe dar clic en el botón actualizar, y en seguida se visualizará la página con los datos expuestos a ser modificados (Ver Figura 5).



Figura5 Página perfil personal datos actualizar

Cabe destacar que en esta página se puede cambiar la clave del sitio estudiantil. Una vez realizados los cambios se procede a dar clic en el botón actualizar y se presentara un mensaje de confirmación al cual se dará clic en si listo, con lo cual se completará el proceso de actualización de los datos personales dentro del perfil personal del estudiante.

Inscripciones

Al dar clic en el botón izquierdo inscripciones se visualizará una lista de carreras optadas en diferentes periodos en la institución (Ver Figura6).

The screenshot shows the 'OASIS ESPOCH' website interface. At the top, it says 'Sitio de Información Estudiantil V2.0'. Below the navigation bar, the user's name 'LUIS GUSTAVO PATIÑO' is displayed. The main content area is titled 'Perfil Personal' and contains a table of 'Datos de Inscripción'. On the left, there are buttons for 'Datos', 'Inscripciones', and 'Títulos'. The table has four columns: Carrera, Fecha, Periodo, and Estado.

Datos de Inscripción			
Carrera	Fecha	Periodo	Estado
CENTRO DE EDUCACION FISICA	22/05/2008	INSCRIPCIONES CENTROS DE AP	Confirmado
CENTRO DE IDIOMAS - FACULTADE	25/03/2008	INSCRIPCIONES CENTROS DE AP	Confirmado
COMISION DE VINCULACION	03/06/2009	INSCRIPCIONES CENTROS DE AP	Confirmado
INGENIERIA EN SISTEMAS	15/08/2007	1 OCTUBRE 2007 - 26 FEBRERO 2	Confirmado

Figura6 Lista de inscripciones

Títulos

Al dar clic en el botón izquierdo Títulos se visualizará una lista de títulos obtenidos en pregrado como en postgrado en la institución (Ver Figura7).

The screenshot shows the 'OASIS ESPOCH' website interface. At the top, it says 'Sitio de Información Estudiantil V2.0'. Below the navigation bar, the user's name 'LUIS GUSTAVO PATIÑO' is displayed. The main content area is titled 'Perfil Personal' and contains a table of 'Títulos Académicos'. On the left, there are buttons for 'Datos', 'Inscripciones', and 'Títulos'. The table has four columns: Título, Institucion, Fecha, and No Refrendación.

Títulos Académicos			
Título	Institucion	Fecha	No Refrendación
INFORMATICA	COLEGIO NACIONAL MIXTO AGRO	15/08/2007 0:00:00	XXX

Figura7 Lista de Títulos

3.3.2 Perfil de Discapacidad

Dentro de este sub modulo se visualiza la lista de canets en la cuales el estudiante se ha registrado (Ver Figura8).

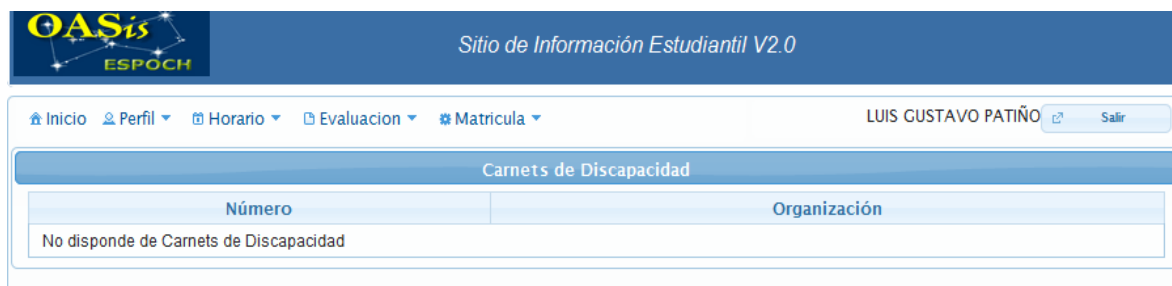


Figura8 Página perfil discapacidad

3.4 Horario

Al presionar la opción horario se visualizara las sub opciones Clase, Examen y +Horarios (Ver Figura9)

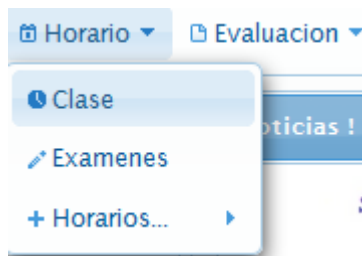


Figura9 Sub Opciones de Horario

3.4.1 Horario Clase

Al dar clic en horario clase se visualiza una tabla con el horario asignado para el presente periodo según la materias en las cuales se haya matriculado, todo esto en una unidad académica seleccionada (Ver Figura10)



The screenshot shows the OASIS ESPQCH website interface. The header includes the logo and the text 'Sitio de Información Estudiantil V2.0'. Below the header, there are navigation links: Inicio, Perfil, Horario, Evaluacion, and Matricula. The user is logged in as 'INGENIERIA EN SISTEMAS' and 'DEYSI CAROLINA LATORRE'. The main content area is titled 'Horario de Clase Asignado' and contains a table with the following data:

Hora	Lunes	Martes	Miercoles	Jueves	Viernes	Sabado	Domingo
PRIMERA HORA: 7h: 0min - 9h: 0m	REDES COMPUTA (SEXTOA) ING. DIEGO FERN	TECNICAS DE SIM (OCTAVOA) ING. IVAN MENES	TECNICAS DE SIM (OCTAVOA) ING. IVAN MENES				
SEGUNDA HORA: 9h: 0min - 11h: 0m	REDES COMPUTA (SEXTOA) ING. DIEGO FERN	DISEÑO DE TESI (OCTAVOB) ING. DIEGO FERN	INTELIGENCIA AF (SEPTIMOA) ING. PATRICIO RE				
TERCERA HORA: 11h: 0min - 13h: 0		REDES COMPUTA (SEXTOA) ING. DIEGO FERN		INTELIGENCIA AF (SEPTIMOA) ING. PATRICIO RE			
CUARTA HORA: 13h: 0min - 15h: 0							
QUINTA HORA EX 14h: 0min - 16h: 0							
SEXTO HORA EXT 16h: 0min - 18h: 0							
SEPTIMA HORA: 18h: 0min - 20h: 0							

Figura10 Horario de clase asignado

A este horario se le suma la posibilidad de convertirlo en PDF e imprimirlo. Para esto se da clic en el icono de PDF y en seguida se visualiza el mismo horario listo a ser impreso (Ver Figura11).



Cédula de Identidad: 060445348-0

Riobamba, lunes 10 marzo 2014

Nombres y Apellidos: DEYSI CAROLINA LATORRE

Horario de Clase: 17 MARZO - 8 AGOSTO 2014

Hora	Lunes	Martes	Miercoles	Jueves	Viernes	Sabado	Domingo
PRIMERA HORA 7h: 0min - 9h: 0min	REDES COMPUTADORES (SEXTOA) ING. DIEGO FERNANDO AVILA PESANTEZ	TECNICAS DE SIMULACION (OCTAVOA) ING. IVAN MENES CAMEJO	TECNICAS DE SIMULACION (OCTAVOA) ING. IVAN MENES CAMEJO				
SEGUNDA HORA 9h: 0min - 11h: 0min	REDES COMPUTADORES (SEXTOA) ING. DIEGO FERNANDO AVILA PESANTEZ	DISEÑO DE TESIS (OCTAVOB) ING. DIEGO FERNANDO AVILA PESANTEZ	INTELIGENCIA ARTIFICIAL (SEPTIMO A) ING. PATRICIO RENE MORENO COSTALES				
TERCERA HORA 11h: 0min - 13h: 0min		REDES COMPUTADORES (SEXTOA) ING. DIEGO FERNANDO AVILA PESANTEZ		INTELIGENCIA ARTIFICIAL (SEPTIMO A) ING. PATRICIO RENE MORENO COSTALES			

Figura11 Horario de clase asignado PDF

3.4.2 Horario Examen

Al dar clic en horario Examen se visualiza una tabla con el horario de exámenes para el presente periodo según las materias en las cuales se haya matriculado, todo esto en una unidad académica seleccionada (Ver Figura12)

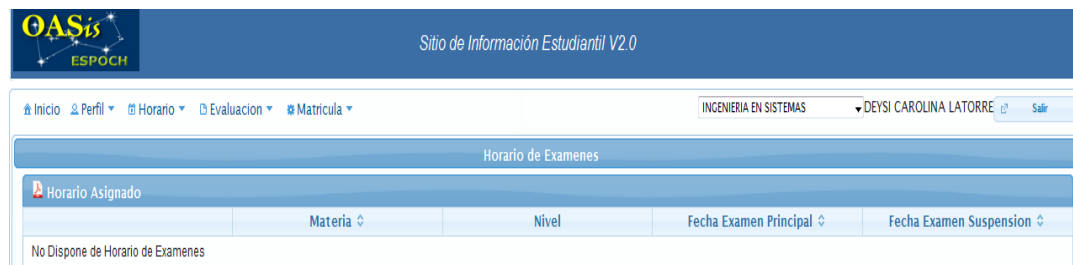


Figura12 Horario de examen asignado

De igual forma este horario puede ser convertido en PDF para imprimirlo.

3.4.3 + Horarios

Al dar clic en +Horarios se visualiza las sub opciones Horario Materia y Horario Paralelo (Ver Figura 13)

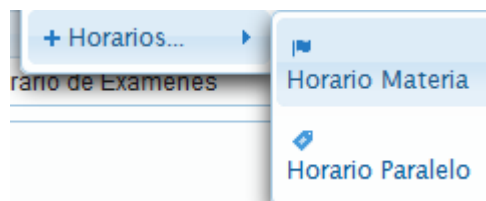


Figura 13 Sub Opciones +Horarios

3.4.3.1 Horario Materia

La dar clic en la sub opción Horario Materia se visualizara una lista de niveles, que al dar clic en uno de ellos se desplegara una lista de materias, cada una de estas con una tabla representando a un horario (Ver Figura 14).

Horario de Clase por Materias								
PRIMERO	▶ ECOLOGIA Y MEDIO AMBIENTE							
SEGUNDO	▶ METODOLOGIA DE LA INVESTIGACION							
TERCERO	▼ MATEMATICA II							
CUARTO	Horario de Clase: MATEMATICA II							
QUINTO	Hora_	Lunes	Martes	Miercoles	Jueves	Vierenes	Sabado	Domingo
SEXTO	PRIMERA HORA 7h: 0min - 9h: 0min	◦ SEGUNDO B				Selecione un Nivel		
SEPTIMO	SEGUNDA HORA 9h: 0min - 11h: 0min	◦ SEGUNDO A	◦ SEGUNDO B	◦ SEGUNDO A		◦ SEGUNDO A		
OCTAVO	TERCERA HORA 11h: 0min - 13h: 0min					◦ SEGUNDO B		
NOVENO	CUARTA HORA 13h: 0min - 15h: 0min							
DECIMO								

Figura 14., Horario por materia

De igual forma el horario por materia puede ser exportado a PDF para en lo posterior se impreso.

3.4.3.2 Horario Paralelo

La dar clic en la sub opción Horario Paralelo se visualizara una lista de niveles, que al dar clic en uno de ellos se desplegara una tabla con el horario referente a este (Ver Figura 15).

Horario Niveles								
Horario de Clase: PRIMERO A								
PRIMEROA	Hora	Lunes	Martes	Miercoles	Jueves	Viernes	Sabado	Domingo
PRIMEROB	PRIMERA HORA: 7h: 0min - 9h: 0min	FUNDAMENTOS DE PR	ALGEBRA LINEAL	FUNDAMENTOS DE PR	ALGEBRA LINEAL	FUNDAMENTOS DE PR		
SEGUNDOB		ING. EDUARDO ROLAN	DR. JUAN MARIO VARG	ING. EDUARDO ROLAN	DR. JUAN MARIO VARG	ING. EDUARDO ROLAN		
SEGUNDOB	SEGUNDA HORA: 9h: 0min - 11h: 0min	MATEMATICA I	FISICA I	MATEMATICA I	FISICA I	MATEMATICA I		
TERCEROA		DRA. NARCISA DE JESI	ING. BERNARDO EZEQ	DRA. NARCISA DE JESI	ING. BERNARDO EZEQ	DRA. NARCISA DE JESI		
TERCEROB	TERCERA HORA: 11h: 0min - 13h: 0min	TECNICAS DE ESTUDI			COMUNICACION ORAL			
CUARTOA		DR. JULIO ROBERTO S			ING. MIGUEL ANGEL CA			
CUARTOB	CUARTA HORA: 13h: 0min - 15h: 0min							
CUARTOC	QUINTA HORA EXT: 14h: 0min - 16h: 0min							
QUINTOA								
QUINTOB	SEXTA HORA EXT: 16h: 0min - 18h: 0min							
SEXTOA								
SEXTOB	SEPTIMA HORA: 18h: 0min - 20h: 0min							

Figura 15. Horario por paralelo

De la misma forma que en los horarios anteriores el presente puede exportarse a un archivo PDF.

3.5 Evaluación

Al dar clic en la opción Evaluación del menú principal, se despliega la lista de sub opciones: Notas y + Notas (Ver Figura 16).

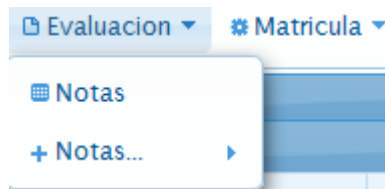


Figura 16. Sub Opciones de Evaluación

3.5.1 Notas

Al dar clic en la sub opción Notas se visualiza una tabla con las notas en las materias seleccionadas para el presente periodo académico en una unidad académica (Ver Figura 17).

Notas Academicas

▼ Materias Asignadas

Periodo : 17 MARZO - 8 AGOSTO 2014

Código	Materia	Eval Acum	Ex.Princip	Prom.Ciclo	Ex.Susp	Total	Equv
IS14136	INTELIGENCIA ARTIFICIAL	0	0	0	0	0	No Aprueba
IS14146	TECNICAS DE SIMULACION	0	0	0	0	0	No Aprueba
IS14148	DISEÑO DE TESIS	0	0	0	0	0	No Aprueba
IS16309	REDES COMPUTADORES	0	0	0	0	0	No Aprueba

#Materias: 4

► Materias Convalidadas

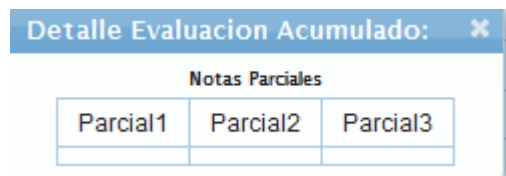
► Materias Retiradas

▼ Promedio Notas :

Promedio General: 0.0/40
Promedio Ciclo: 0.0/20

Figura 17. Notas de materias en el presente periodo.

Adicionalmente a la vista general de las notas en su despliegue acostumbrado, se puede visualizar el detalle del acumulado en sus tres parciales dando clic en el mismo (Ver Figura 18).



Notas Parciales		
Parcial1	Parcial2	Parcial3

Figura 18. Mensaje con las notas parciales del acumulado

De la misma forma que en los horarios, las notas también pueden ser exportadas a PDF para ser impresas.

3.5.2 +Notas

Al dar clic en +Notas se visualiza la sub opción Notas Anteriores (Ver Figura 19).



Figura 19. Sub Opción +Notas

3.5.2.1 Notas Anteriores

Al dar clic la sub opción Notas Anteriores se desplegará una lista de periodos optados por el estudiante, al seleccionar uno de estos se visualizará una tabla con las notas obtenidas en las materias optadas en ese periodo en la unidad académica optada (Ver Figura 20).

Notas Académicas								
9 SEPTIEMBRE 2013	Materias Asignadas							
3 OCTUBRE 2013	Período :9 SEPTIEMBRE 2013 - 17 ENERO 2014							
21 MARZO - 3 ABRIL 2014	Código	Materia	Eval Acur	Ex.Princip	Prom.Cicl	Ex.Susp	Total	Equiv
13 SEPTIEMBRE 2013	IS14115	INGENIERIA DE SOFTWARE I	22	4	14	15	28	APROBADO
15 MARZO - 23 JUNIO 2014	IS14138	APLICACIONES WEB	22	11	17	0	33	APROBADO
15 SEPTIEMBRE 2013	IS14157	ARQUITECTURA DE SOFTWARE	19	9	14	0	28	APROBADO
17 MARZO 2009	IS16309	REDES COMPUTADORES	20	5	13	12	25	REPROBADO
15 SEPTIEMBRE 2013	#Materias: 4							
17 MARZO - 30 JUNIO 2014	Materias Convalidadas							
1 OCTUBRE 2007	Materias Retiradas							
19 MARZO 2007	Promedio Notas :9 SEPTIEMBRE 2013 - 17 ENERO 2014							
25 SEPTIEMBRE 2013	Promedio General: 28.5/40							
	Promedio Ciclo: 14.25/20							

Figura 20. Notas Anteriores

De igual forma que notas, notas anteriores también puede exportarse a PDF y ser impresa.

3.5 Matrícula

Al dar clic en la opción Matrícula del menú principal se despliega la lista de sub opciones:

Matrículas Realizadas, Pre-Matrícula e Inscripción Centros Apoyo (Ver Figura 21).

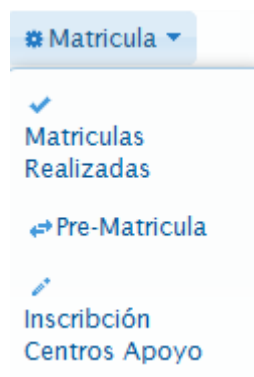


Figura 21. Sub Opciones Matrícula

3.5.1 Matrículas Realizadas

Al dar clic en la sub opción Matrículas Realizadas se desplegará una tabla con los periodos optadas o matriculados en la unidad académica (Ver Figura 22).

Matrículas Realizadas		
Periodo	Fecha de Matricula	No Materias
25 SEPTIEMBRE 2006 - 27 FEBRERO 2007	Septiembre - 18 - 2006	7
19 MARZO 2007 - 30 JULIO 2007	Marzo - 7 - 2007	6
1 OCTUBRE 2007 - 26 FEBRERO 2008	Septiembre - 26 - 2007	3
17 MARZO - 30 JULIO 2008	Marzo - 6 - 2008	5
15 SEPTIEMBRE 2008 - 26 FEBRERO 2009	Septiembre - 8 - 2008	5
17 MARZO 2009 - 30 JULIO 2009	Marzo - 2 - 2009	5
15 SEPTIEMBRE 2009 - 5 FEBRERO 2010	Septiembre - 7 - 2009	5
15 MARZO - 23 JULIO 2010	Mayo - 14 - 2010	5
13 SEPTIEMBRE 2010 - 4 FEBRERO 2011	Septiembre - 3 - 2010	3
21 MARZO - 3 AGOSTO 2011	Marzo - 8 - 2011	5
3 OCTUBRE 2011 - 28 FEBRERO 2012	Noviembre - 10 - 2011	4
9 SEPTIEMBRE 2013 - 17 ENERO 2014	Agosto - 26 - 2013	4

Figura 22. Matriculas Realizadas

Al dar clic uno de los periodos se visualizar un mensaje con las materias optadas en el presente (Ver Figura 23).

Materias Optadas: 15 SEPTIEMBRE 2008 - 26 FEBRERO 2009				
Código	Nombre	No Creditos	No Matrícula	Curso
IS11113	ETICA PROFESIONAL	2.0	1	CUARTO2
IS11123	REALIDAD SOCIO ECONOMICA	2.0	1	CUARTO1
IS13112	MATEMATICA II	6.0	2	TERCERO1
IS13143	CONTABILIDAD	4.0	1	CUARTO1
Total Creditos Periodo: 14.0				

Figura 23. Materias matriculadas en el periodo seleccionado.

3.5.2 Pre-Matrícula

Al presionar la sub opción Pre –Matrícula se desplegara la propuesta de materias a matricularse, a aquí el estudiante puede aceptar o negar cada una de las propuestas (Ver Figura 24). Seleccionadas las materias, se procede a seleccionar los diferentes paralelos, se actualiza total y por último se consulta la propuesta.

Acepta	Código	Nombre	N° Matrícula	N° Horas	Creditos	Nivel	Paralelo
Si	FCEI10001	CALCULO DIFEREN	1	5.0	5.0	PRIMERO	1
Si	FCEI10002	ALGEBRA LINEAL I	1	6.0	6.0	PRIMERO	1
Si	FCEI10003	FUNDAMENTOS DE	1	4.0	4.0	PRIMERO	1
Si	FCEI10004	EXPRESION ORAL	1	2.0	2.0	PRIMERO	1
Si	FCEI10005	TECNICAS DE ESTI	1	2.0	2.0	PRIMERO	1
Si	FCEI1011	ELECTIVA I	1	5.0	5.0	DECIMO	1
Si	FCEI1012	ELECTIVA II	1	5.0	5.0	DECIMO	1
Si	FCEI1013	ELECTIVA III	1	5.0	5.0	DECIMO	1

Materias Optativas

[Actualizar Totales](#) Total Horas:34.0 Total Creditos:34.0

[Consultar Propuesta](#)

Figura 24. Pre-matricula selección materias

Posterior a dar clic en consultar propuesta, se visualizará la página confirmación, en la cual se puede aceptar o rechazar la propuesta. (Ver Figura 25)

Inicio Perfil Horario Evaluacion **Matricula** INGENIERIA EN ESTADISTICA INFORMATICA GABRIELA DE LA NUBE SANTANDER Salir

Si desea aceptar la propuesta, de clic en "Aceptar Propuesta", caso contrario clic en "Realizar Nueva Propuesta"

Notas:

[Aceptar Propuesta](#) [Realizar Nueva Propuesta](#)

Recuerde que el arancel tiene que ser cancelado por medio del Banco del Pacifico en cualquiera de sus medios de pago Ej: <http://www.intermatico.com>, bancomático, cajas del banco. Una vez que la orden de pago haya sido cancelada deberá presentarse en la secretaría académica de la Unidad Académica donde se prodrá registrar a su matrícula como definitiva

Figura 25. Página confirmación

Una vez confirmada la pre-matricula únicamente resta esperar que el director acepta la solicitud.

3.5.3 Inscripción Centros Apoyo

Al dar clic en Inscripción Centros Apoyo se visualizara la página de inscripción en los centros de apoyo: Centro de Vinculación, Centro de Idiomas y Centro de Educación Física, en los cuales se pueden hacer clic y se obtendrá un código de inscripción (Ver Figura 26).

The screenshot shows the 'OASIS ESPOCH' logo and 'Sitio de Información Estudiantil V2.0' in the top header. A navigation menu includes 'Inicio', 'Perfil', 'Horario', 'Evaluación', and 'Matricula'. A user profile box on the right displays 'Inscripción Centro de Educación Física' and 'Su código de Educación Física es: GABRIELA 20990 NUBE SANTANDER'. The main content area is titled 'Inscripción en los Centros de Apoyo' and contains the following instructions:

Inscribirse en los centros de apoyo:

1. Agregarse como alumno al centro de apoyo en caso de no serlo:
 - Si no forma parte del Centro de Educación Física [Clic Aquí](#)
 - Si no forma parte de la comisión de Vinculación [Clic Aquí](#)
 - Si no forma parte de Ingles [Clic Aquí](#)
2. Cerrar sesión y volver a logearse
3. Seleccionar en la parte superior derecha la carrera: "Educación Física" y realizar el proceso de Matricula via Web

Figura 26. Inscripción Centros Apoyo

4. Conclusiones

- ❖ Se ha detallado la funcionalidad del sitio estudiantil desglosando todas las opciones del menú principal
- ❖ Se realizó un documento de apoyo al aprendizaje del sitio para los estudiantes.

5. Recomendaciones

- ❖ Se recomienda hacer uso del presente documento como material de apoyo para capacitaciones futuras.
- ❖ Perfeccionar el presente manual con una detallada especificación de las funcionalidades

Anexo7.- Pruebas para las historias de usuario.

Prueba para la historia de usuario2: Perfil personal del estudiante

Descripción

El usuario estudiante procede visualizar sus datos personales.

Entrada

1.- El usuario estudiante una vez ingresado en el sistema procede a seleccionar perfil.

2.- Selecciona personal.

3.- Visualiza sus datos personales.

Resultado deseado

Al seleccionar el perfil personal se visualizara los datos personales del usuario estudiante.

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Perfil personal del estudiante	Al seleccionar el perfil personal se visualizara los datos personales del usuario estudiante.	Si
Modificaciones	Luis Patiño			

Prueba para la historia de usuario3: Actualizar datos contacto perfil.

Descripción

El usuario estudiante modificar los datos de contacto en el perfil personal del estudiante.

Entrada

1.- El usuario estudiante una vez ingresado en el sistema procede a seleccionar perfil y personal

2.- Procede a editar sus datos de contacto.

3.- Modifica los datos de contacto

4.- Actualiza los datos de contacto modificados.

5.- Visualiza sus datos de contacto actualizados.

Resultado deseado

Al editar los datos de contacto, el usuario estudiante actualizara estos, visualizando los cambios.

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Actualizar datos contacto perfil	Al editar los datos de contacto, el usuario estudiante actualizara estos, visualizando los cambios.	Si
Modificaciones	Luis Patiño			

Prueba para la historia de usuario4: Perfil personal del estudiante

Descripción

El usuario estudiante procede visualizar sus datos de discapacidad.

Entrada

- 1.- El usuario estudiante una vez ingresado en el sistema procede a seleccionar perfil.
- 2.- Selecciona discapacidad.
- 3.- Visualiza sus datos de discapacidad si los posee.

Resultado deseado

Al seleccionar el perfil de discapacidad se visualizara los datos de discapacidad del usuario estudiante en caso de poseerlos.

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Perfil personal del estudiante	Al seleccionar el perfil personal se visualizara los datos personales del usuario estudiante.	Si
Modificaciones	Luis Patiño			

Prueba para la historia de usuario5: Inscripciones del estudiante

Descripción

El usuario estudiante procede visualizar sus inscripciones realizadas en las unidades académicas de las ESPOCH.

Entrada

- 1.- El usuario estudiante una vez ingresado en el sistema selecciona perfil y personal.
- 2.- Selecciona inscripciones.
- 3.- Visualiza su lista de inscripciones en las unidades académicas.

Resultado deseado

Al seleccionar inscripciones se visualizara la lista de inscripciones realizadas.

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Inscripciones del estudiante	Al seleccionar inscripciones se visualizara la lista de inscripciones realizadas.	Si
Modificaciones	Luis Patiño			

Prueba para la historia de usuario6: Periodos optados en una unidad académica

Descripción

El usuario estudiante procede visualizar sus periodos optados en una determinada unidad académica.

Entrada

- 1.- El usuario estudiante una vez ingresado en el sistema selecciona matricula.
- 2.- Selecciona matriculas realizadas
- 3.- Visualiza su lista de periodos optados o matriculados en la unidad académica.

Resultado deseado

Al seleccionar matriculas realizadas, se visualizara una lista de periodos optados.

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Periodos optados en una unidad académica	Al seleccionar matriculas realizadas, se visualizara una lista de periodos optados	Si
Modificaciones	Luis Patiño			

Prueba para la historia de usuario7: Asignaturas optadas en un periodo optado

Descripción

El usuario estudiante procede seleccionar un periodo y se visualiza las asignaturas optadas en este periodo.

Entrada

- 1.- El usuario estudiante una vez ingresado en el sistema selecciona matricula.
- 2.- Selecciona matriculas realizadas
- 3.- Selecciona un periodo optado
- 4.- Visualiza en un cuadro de mensaje su lista de asignaturas optadas.

Resultado deseado

Al seleccionar un periodo determinado, se visualizara una lista de asignaturas optadas en ese periodo.

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Asignaturas optadas en un periodo optado	Al seleccionar un periodo determinado, se visualizara una lista de asignaturas optadas en ese periodo.	Si
Modificaciones	Luis Patiño			

Prueba para la historia de usuario8: Títulos pre-grado

Descripción

El usuario estudiante procede visualizar sus títulos de pre-grado.

Entrada

- 1.- El usuario estudiante una vez ingresado en el sistema selecciona perfil y personal
- 2.- Selecciona títulos.
- 3.- Visualiza una lista de títulos que posee actualmente.

Resultado deseado

Al seleccionar títulos, se visualizara su lista de títulos de pregrado y posgrado.

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Títulos pre-grado	Al seleccionar títulos, se visualizara su lista de títulos de pregrado y posgrado.	Si
Modificaciones	Luis Patiño			

Prueba para la historia de usuario9: Notas de cada asignatura

Descripción

El usuario estudiante procede visualizar sus notas en las asignaturas seleccionado un determinado periodo académico.

Entrada

- 1.- El usuario estudiante una vez ingresado en el sistema selecciona evaluación
- 2.- Selecciona notas.
- 3.- Visualiza las notas de las asignaturas, que actualmente opta.

Resultado deseado

Al seleccionar notas, se visualizara la lista de notas por asignaturas.

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Notas de cada asignatura	Al seleccionar notas, se visualizara la lista de notas por asignaturas	Si
Modificaciones	Luis Patiño			

Prueba para la historia de usuario10: Notas parciales

Descripción

El usuario estudiante procede a seleccionar la nota acumulada para visualizar sus notas parciales.

Entrada

- 1.- El usuario estudiante una vez ingresado en el sistema selecciona evaluación
- 2.- Selecciona notas.
- 3.- Selecciona la nota acumulada de la asignatura.
- 3.- Se visualiza en un cuadro de mensaje las notas parciales de la nota seleccionada.

Resultado deseado

Al seleccionar una nota acumulada, se visualizara las notas parciales de esta.

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Notas parciales	Al seleccionar una nota acumulada, se visualizara las notas parciales de esta	Si
Modificaciones	Luis Patiño			

Prueba para la historia de usuario11: Horario de clase del estudiante

Descripción

El usuario estudiante procede a visualizar su horario de clase definido según las asignaturas optadas.

Entrada

- 1.- El usuario estudiante una vez ingresado en el sistema selecciona horario
- 2.- Selecciona clase.
- 3.- Se visualiza su horario de clase.

Resultado deseado

Al seleccionar horario de clase, se visualizara una tabla con su horario de clase actual.

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Horario de clase del estudiante	Al seleccionar horario de clase, se visualizara una tabla con su horario de clase actual	Si
Modificaciones	Luis Patiño			

Prueba para la historia de usuario12: Horarios de clase por paralelo

Descripción

El usuario estudiante procede a visualizar la lista de horarios de clase de la unidad académica agrupados por nivel y paralelo.

Entrada

- 1.- El usuario estudiante una vez ingresado en el sistema selecciona horario
- 2.- Selecciona más horarios.
- 3.- Selecciona horario paralelo
4. Visualiza una lista de nivel con paralelos a seleccionar según la unidad académica.
5. Selecciona un nivel paralelo
6. Visualiza el horario del nivel paralelo.

Resultado deseado

Al seleccionar un nivel paralelo, se visualizará el horario establecido.

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Horarios de clase por paralelo	Al seleccionar un nivel paralelo, se visualizará el horario establecido	Si
Modificaciones	Luis Patiño			

Prueba para la historia de usuario13: Horarios de clase por materia

Descripción

El usuario estudiante procede a visualizar la lista de horarios de clase de la unidad académica agrupados por materia.

Entrada

- 1.-** El usuario estudiante una vez ingresado en el sistema selecciona horario
- 2.-** Selecciona más horarios.
- 3.-** Selecciona horario materia
- 4.** Visualiza una lista de nivel a seleccionar según la unidad académica.

5. Selecciona un nivel y materia

6. Visualiza el horario de la materia.

Resultado deseado

Al seleccionar una materia, se visualizará el horario establecido

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Horarios de clase por materia	Al seleccionar una materia, se visualizará el horario establecido	Si
Modificaciones	Luis Patiño			

Prueba para la historia de usuario14: Horarios de exámenes

Descripción

El usuario estudiante procede a visualizar la lista de horarios de exámenes en el periodo de los mismos

Entrada

1.- El usuario estudiante una vez ingresado en el sistema selecciona horario

2.- Selecciona Exámenes.

3. Visualiza el horario de exámenes establecido para el periodo activo.

Resultado deseado

Al seleccionar horario examen, se visualizará el horario establecido.

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Horarios de exámenes	Al seleccionar horario examen, se visualizará el horario establecido	Si
Modificaciones	Luis Patiño			

Prueba para la historia de usuario15: Inscribirse centros de apoyo

Descripción

El usuario estudiante procede seleccionar cada uno de los centros de apoyo para obtener su código respectivo.

Entrada

- 1.- El usuario estudiante una vez ingresado en el sistema selecciona matricula e inscribirse centros de apoyo.
- 2.- Selecciona un determinado centro de apoyo y se inscribe.

3.- Visualiza un mensaje con el código en el centro de apoyo.

Resultado deseado

Al seleccionar un centro de apoyo, se visualizara un mensaje confirmación de inscripción.

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Inscribirse centros de apoyo	Al seleccionar un centro de apoyo, se visualizara un mensaje confirmación de inscripción	Si
Modificaciones	Luis Patiño			

Prueba para la historia de usuario16: Noticias estudiantiles

Descripción

El usuario estudiante procede visualizar las noticias estudiantiles que se presentan al inicio del sistema.

Entrada

1.- El usuario estudiante ingresa al sistema.

2.-Visualiza las noticias estudiantiles.

Resultado deseado

Al iniciar el sistema, visualizara un show slider de noticias estudiantiles.

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Noticias estudiantiles	Al iniciar el sistema, visualizara un show slider de noticias estudiantiles	Si
Modificaciones	Luis Patiño			

Prueba para la historia de usuario17: Pre-matricula en materias

Descripción

El usuario estudiante procede cargar su propuesta de matrícula, selecciona las materias obligatorias y/u optativas, y por ultimo solicita la matricula. Todo esto en el periodo de matrículas.

Entrada

- 1.- El usuario estudiante una vez ingresado en el sistema selecciona matricula y pre-matricula
- 2.- Visualiza una lista de materias en las que puede matricularse

3. -Selecciona las materias en las que se va a matricular.

4.-Consulta propuesta.

5. Visualiza página de confirmación

6. Confirma propuesta y se matricula.

Resultado deseado

Matricularse en materias en una unidad académica específica.

Cumplimiento de la Prueba

Prueba superada con éxito.

Actividad	Autor	Historia de Usuario	Resultado deseado	Cumplimiento
Prueba	Estudiante de la escuela de Fisca y Matemática de la ESPOCH	Pre-matricula en materias	Matricularse en materias en una unidad académica específica.	Si
Modificaciones	Luis Patiño			

BIBLIOGRAFÍA

1. APACHE AXIS2 ARCHITECTURE GUIDE

<http://axis.apache.org/axis2/java/core/docs/Axis2ArchitectureGuide.html#thearchie.html#thearchie>

2012-01-01

2. AXIS2 CREATING CLIENTS

<http://axis.apache.org/axis2/java/core/docs/userguide-creatingclients.html#createclients>

2012-01-01

3. APACHE SOFTWARE FOUNDATION

<http://axis.apache.org/axis2/java/core/docs/userguide.html> >

2012-04-17

4. CODEJBOS

<<http://www.codejobs.biz/es/blog/2013/06/05/programacion-extrema-xp#sthash.N4UmJ7bY.dpbs>>

2013 06-01

5. CONSORCIO WORLD WIDE WEB ESPAÑA

<<http://www.w3c.es/Divulgacion/GuiasBreves>>

2009-01-01

6. GLASSFISH METRO

<<https://metro.java.net/>>

2013-01-01

7. IBM DESARROLLO

<<http://publib.boulder.ibm.com/infocenter/radhelp/v7r5/index.jsp?topic=%2Forg.eclipse.jst.ws.consumption.ui.doc.user%2Fconcepts%2Fcwsdlud.html>>

2010-01-1

8. INSTITUTO POLITÉCNICO NACIONAL

<<http://148.204.210.201/tesis/1313442753812TesisEmilioAn.pdf>>

2011-01-01

9. INTRODUCCIÓN A LOS SERVICIOS WEB

<<http://www.di.uniovi.es/~labra/cursos/XMLAvanzado/ServiciosWeb.html#%286%29>>

2006-10-01

10. JAVA NET

<<https://www.java.net/forums/glassfish/metro-and-jaxb>>

2014-01-01

11. KULANDAI JAXB TUTORIAL

<<http://javapapers.com/jee/jaxb-tutorial/>>

2007-07-09

12. METRO

<<https://metro.java.net/guide/ch01.html>>

25 10 2013

<<https://metro.java.net/guide/ch02.html>>

2013-10-25

13. MICROSOFT DEVELOPER NETWORK WCF

<<http://social.msdn.microsoft.com/Forums/en-US/435f43a9-ee17-4700-8c9d-d9c3ba57b5ef/advantages-disadvantages-of-webservices?forum=asmxandxml>>

2007-11-19

<<http://msdn.microsoft.com/es-es/library/t745kdsh%28v=vs.100%29.aspx> >

2012-02-10

<<http://msdn.microsoft.com/es-ec/library/ms731082.aspx>>

2014-01-01

<<http://msdn.microsoft.com/es-ec/library/ms731082.aspx>>

2014-01-01

<<http://msdn.microsoft.com/es-es/library/bb412196.aspx?cs-save-lang=1&cs-lang=csharp#code-snippet-1>>

2011-10-14

<<http://msdn.microsoft.com/es-es/library/bb924407%28v=s.110%29.aspx>>

2014-01-01

<[http://msdn.microsoft.com/es-es/library/ms731082\(v=VS.90\).aspx](http://msdn.microsoft.com/es-es/library/ms731082(v=VS.90).aspx)>

2011-01-01

14. MICROSOFT WINDOWS SERVER

<<http://technet.microsoft.com/en-us/library/ccc10047-f34e-4862-84cd-23dad191a06c> >

2008-01-1

15. UNDERSTANDING WEB SERVICES SPECIFICATIONS

<<http://www.ibm.com/developerworks/ssa/webservices/tutorials/ws-understand-web-services4/>>

2011-08-08

16. UNIVERSIDAD TÉCNICA DEL NORTE

<<http://repositorio.utn.edu.ec/bitstream/123456789/624/3/CAPITULO%20IV.pdf>>

2012-10-01

17. UNIVERSITY OF SEVILLA PRUEBAS DE PROGRAMACION EXTREMA

<http://www.lsi.us.es/~javierj/investigacion_ficheros/PSISEXTREMA.pdf>

2009-04-23

18. W3C WEB SERVICES DESCRIPTION LANGUAGE

[<http://www.w3.org/TR/2007/REC-wsdl20-primer-20070626/ >](http://www.w3.org/TR/2007/REC-wsdl20-primer-20070626/)

2007-06-26

19. WEB SERVICES DESCRIPTION LANGUAGE

[<http://www.w3.org/TR/wsdl >](http://www.w3.org/TR/wsdl)

2001-03-15