



ESCUELA POLITÉCNICA NACIONAL

**APLICACIONES EN AMBIENTES
LIBRES**

SISTEMA DE GESTIÓN DE OBJETOS DE APRENDIZAJE

INTEGRANTES

CRIOLLO JOSÉ

TAMAYO EDISON

VASCO ELSA

GRUPO - 05



1. Contenido

2. Tema	2
3. Definición del problema.....	2
4. Objetivos.....	2
Objetivo General	2
Objetivos Específicos	2
5. Requerimientos	2
6. Diseño Arquitectónico	3
Arquitectura orientada a servicio (SOA).....	3
Tecnologías que soportan a los Servicios Web.....	4
XML (Extensible Markup Language)	4
WSDL (Web Services Description Language)	4
SOAP (Simple Object Access Protocol)	4
UDDI (Universal Description, Discovery and Integration)	5
7. Diseño Detallado	5
Arquitectura del repositorio de objetos de aprendizaje con servicio web.....	5
Descripción de capas	6
Funcionamiento de la Arquitectura del repositorio.....	7
8. Sistema Final.....	7
9. Descripción del proceso de construcción del software	7
Herramientas.....	8
10. Conclusiones	8
11. Recomendaciones.....	8
12. Observaciones.....	8
13. Bibliografía.....	9
14. Anexos	10
Anexo 1	10
Anexo 2	10
Anexo 3	11

2. Tema

Sistema de Gestión de Objetos de Aprendizaje

3. Definición del problema

Las TIC'S (Tecnologías de la Información y la comunicación), están revolucionando el proceso de la enseñanza - aprendizaje con el desarrollo de nuevas metodologías y herramientas que, a través de internet, ponen el conocimiento al alcance de los estudiantes. Entre ellas se puede citar a las plataformas LMS (Sistemas de Gestión de Aprendizaje) como: Moodle, dotLRN, Claroline, Dokeos, y otras, que tienen la capacidad de integrar personas, procesos y equipos en aulas virtuales, apoyando enormemente al proceso de enseñanza al permitir a los docentes incorporar a sus materias, recursos de aprendizaje digitales que son accesibles desde internet en cualquier momento y desde cualquier lugar. La construcción de este tipo de recursos es un reto para el docente, ya que se debe cumplir con normas que aseguren su calidad y efectividad en el aprendizaje.

Muchos docentes hacen grandes esfuerzos para crear recursos de aprendizaje digitales, que incorporan a sus aulas virtuales como apoyo a su trabajo, pero este valioso material queda invisible para otros docentes que están impartiendo los mismos contenidos y que deben crear sus propios recursos educativos, generando una innecesaria duplicación de esfuerzos y de recursos educativos en el entorno de enseñanza virtual.

Los Objetos de Aprendizaje son recursos digitales debidamente estructurados utilizados para la educación, que deben cumplir con estándares de calidad para que sean un aporte efectivo en el proceso de la enseñanza. Sin embargo, los docentes no disponen de una herramienta que facilite y le guíe en la construcción de objetos de aprendizaje. A través de internet se puede conseguir muchos recursos de aprendizaje digitales de los más variados temas, pero no tienen un control de calidad ni la adecuada rigurosidad científica en su elaboración, por lo que no es recomendable utilizarlos como herramientas de apoyo a la enseñanza.

Para mejorar y facilitar la creación de OA se plantea el desarrollo de una herramienta en la cual lo podrá realizar de una manera fácil y rápida los OA, además poderlos compartir tanto con estudiantes y profesores para de esta manera mejorar el aprendizaje diariamente.

4. Objetivos

Objetivo General

Desarrollar una aplicación informática que apoye a la construcción, almacenamiento e intercambio de objetos de aprendizaje.

Objetivos Específicos

- Estudiar los Objetos de Aprendizaje, para determinar sus partes componentes y las técnicas de construcción.
- Establecer la forma de catalogar, empaquetar y almacenar Objetos de aprendizaje.
- Crear una aplicación informática de software libre para la creación de objetos de aprendizaje.
- Crear una aplicación web utilizando herramientas y metodología de desarrollo de software libre, que permita el almacenamiento, catalogación, difusión y utilización de objetos de aprendizaje.

5. Requerimientos

Los requerimientos solicitados para el sistema son los siguientes.

- Crear un sistema con 3 tipos de usuarios.
- ❖ **Usuario Administrador:** tiene acceso a todo el sistema, es el que instala la aplicación y hace el mantenimiento de la aplicación y de sus datos, por lo tanto, tiene permiso para cargar y borrar OA en el repositorio, para editar la catalogación del OA, pero No tiene permisos para editar su contenido. El administrador debe crear a los usuarios profesores y puede borrarlos. Los usuarios Estudiantes también pueden ser eliminados por el administrador.
- ❖ **Usuario Profesor:** para usar el sistema, debe llenar un formulario y enviar al sistema. En dicho formulario debe registrar: cédula, nombres y apellidos, departamento y facultad a la que pertenece, y su correo electrónico. Posteriormente, el administrador le creará como usuario profesor y le asignará un nombre y una clave para que pueda trabajar en el sistema.
El usuario profesor puede cargar y catalogar OA al repositorio; puede editar o borrar solo OA que han sido cargados por él al repositorio.
El usuario profesor también puede buscar, consultar y descargar OA de otros autores desde el repositorio. Puede también realizar observaciones y sugerencias sobre un OA de otros autores, que se registrarán en una bitácora.
- ❖ **Usuario Estudiante:** para utilizar el sistema debe registrarse con su cédula, nombres y apellidos, carrera y facultad a la que pertenece, correo electrónico, nombre de usuario y contraseña. El sistema valida sus datos y permite su acceso. Luego, el estudiante puede acceder a la aplicación y hacer búsquedas de OA por autor, tema, palabras claves y año de construcción del OA. El estudiante puede descargarse un OA.
- **Importar y Catalogación de los OA**
Dentro de esta opción el sistema presentará una pantalla en la que se tenga una opción para importar un objeto de aprendizaje desde un dispositivo de almacenamiento. En esta misma pantalla se deberá capturar toda la información de este objeto de aprendizaje: nombre del OA, pequeña descripción de su contenido, autor, institución a la que pertenece el autor, fecha de creación, palabras clave para ayudar a su localización. Esta pantalla tendrá dos botones, uno para guardar y otro para cancelar y volver a la pantalla anterior.
- **Buscar OA**
En esta opción el sistema presentará una pantalla para buscar OA por autor, por año, por palabras clave, Como resultado de la búsqueda se presentarán los objetos de aprendizaje que cumplan las condiciones de búsqueda, con los siguientes datos: nombre del OA, descripción de su contenido, autor, institución a la que pertenece el autor, fecha de creación, palabras clave, tamaño del archivo, tipo de archivo, fecha de ingreso al repositorio y botones para desempaquetarlo, editarlo o borrarlo. Todas estas opciones deben estar operativas.
- **Repositorio en la nube:** colocar el repositorio en la nube, de tal manera que se pueda acceder al repositorio a través de un navegador.

6. Diseño Arquitectónico

Arquitectura orientada a servicio (SOA)

En la actualidad el concepto de SOA ha resurgido, gracias al auge y madurez de una nueva tecnología: los Servicios Web. Los Servicios Web se han convertido en el estandarte de SOA, ya que esta tecnología posee un conjunto de características que permiten cubrir todos los principios de la orientación a servicios.

Una arquitectura orientada a servicios (SOA) es una forma de compartir funciones de manera generalizada y flexible.

En general se puede decir que SOA es un modelo de arquitectura que establece un marco de diseño para la integración de aplicaciones independientes de manera que desde la red pueda accederse a sus funcionalidades, las cuales se ofrecen como servicios. La forma más habitual de implementarla es mediante Servicios Web, una tecnología basada en estándares e independiente de la plataforma.

Básicamente una arquitectura orientada al servicio es una colección de servicios. Estos servicios se comunican entre sí. La comunicación puede involucrar simplemente el paso de datos o la coordinación de alguna actividad entre varios servicios.

La característica principal de SOA es que es una Arquitectura con acoplamiento débil, esto significa que el cliente de un servicio es independiente de la construcción de ese servicio.

Una arquitectura SOA está formada por tres agentes: un proveedor, un intermediario y un cliente que no presentan ningún acoplamiento entre ellos.

El proveedor ofrece un servicio determinado y que el cliente no tiene por qué conocer directamente. El cliente aprende como utilizar el servicio a partir de la información que le ofrece el intermediario que normalmente simplifica el uso de dicho servicio. El cliente sólo sabe cómo utilizar el servicio, es decir, como enviar y recibir datos pero no conoce ningún detalle de su implementación interna.

Tecnologías que soportan a los Servicios Web.

Las tecnologías más importantes que definen la arquitectura de un Servicio Web se pueden observar en la Ilustración 1.

XML (Extensible Markup Language)

En español Lenguaje extensible de etiquetas. Es un estándar para describir datos y crear etiquetas. Las características especiales son la independencia de datos, o la separación de los contenidos de su presentación.

El lenguaje XML es una forma, estándar industrial e independiente del sistema, de representar datos. Los datos que se representan usando XML se pueden publicar en múltiples medios porque XML describe la estructura de los datos, no su formato, al contrario que el HTML, los datos de XML se pueden pasar entre aplicaciones porque la estructura de los datos se puede especificar en un esquema, lo que permite que un analizador de sintaxis valide y procese los datos que siguen el esquema.

WSDL (Web Services Description Language)

En español Lenguaje de descripción de servicios web. Es un formato en XML estandarizado para describir servicios web, en este archivo se describe el nombre, la ubicación y la forma de comunicarse con el servicio, así como los métodos y parámetros que utiliza y la forma en que devuelve la respuesta. Así, un usuario puede crear una aplicación cliente que comunica con el servicio web.

Un documento WSDL está dividido en dos partes claramente diferenciadas:

- Par te concreta: Es la parte que define el “como” y “donde”.
- Par te abstracta: Es la parte que define qué hace el servicio a través de los mensajes que envía y recibe.

SOAP (Simple Object Access Protocol)

En español Protocolo de Acceso simple a objetos. Es un protocolo de comunicación, por paso de mensajes XML, se utiliza para la formación de los mensajes intercambiados entre los sistemas distribuidos y la red. Los mensajes

SOAP son independientes de los sistemas operativos y de los protocolos, y pueden ser transportados usando una variedad de protocolos Internet, incluyendo SMTP, y HTTP.

El mensaje está compuesto de tres partes: un sobre, encabezado y el cuerpo. ¿El sobre envuelve al mensaje y contiene el encabezado y el cuerpo; el encabezado es un elemento opcional que provee información para el enrutamiento del mensaje; el cuerpo contiene datos etiquetados como XML.

UDDI (Universal Description, Discovery and Integration)

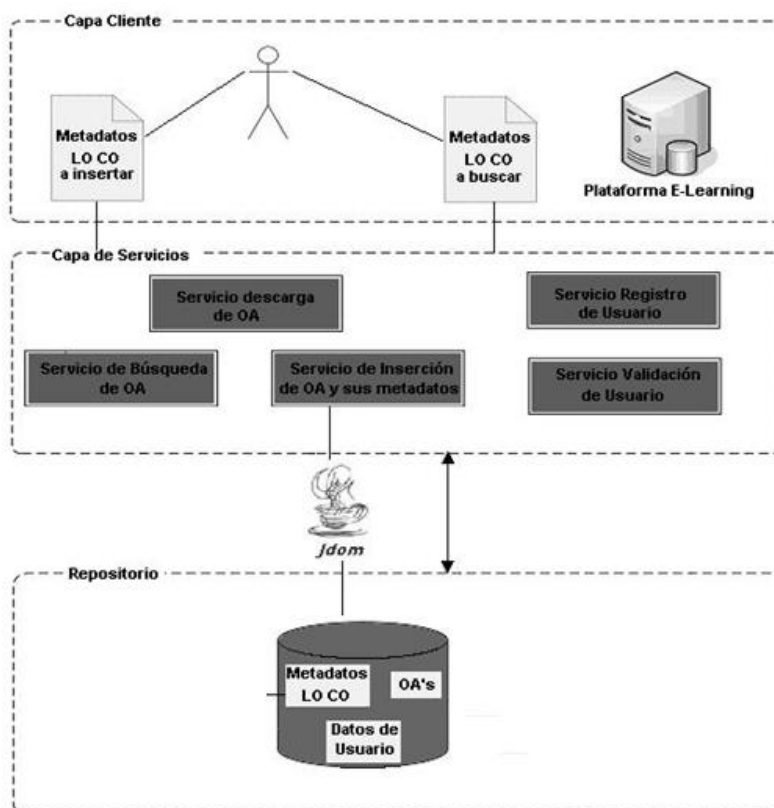
Universal Descripción de Descubrimiento e Integración, es un elemento básico sobre el que se asientan los Servicios web, hace posible que empresas pueden tanto publicar como encontrar Servicios web.

UDDI está construido sobre los estándares de Internet del W3C y de la IETF (Internet Engineering Task Force), como XML, HTTP. Para describir las interfaces hacia los servicios web, utiliza el lenguaje WSDL, las aplicaciones interesadas pueden consultar utilizando mensajes SOAP, lo cual posibilita una interoperabilidad total. UDDI provee un mecanismo para que los negocios se “describan” a si mismos y los tipos de servicios que proporcionan y luego se pueden registrar y publicarse en un Registro UDDI. Tales negocios publicados pueden ser buscados, consultados o “descubiertos” por otros negocios utilizando mensajes con SOAP.

7. Diseño Detallado

Arquitectura del repositorio de objetos de aprendizaje con servicio web

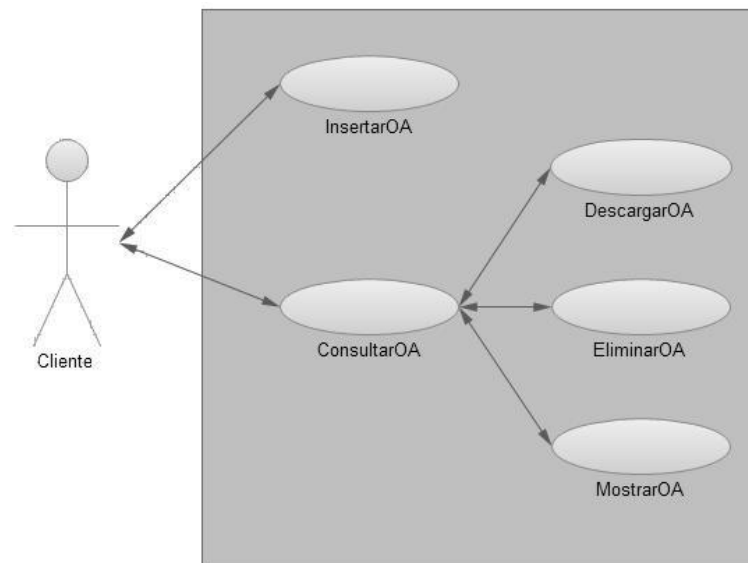
La propuesta de diseño de repositorio de objetos de aprendizaje implementada con servicios Web está estructurada en tres capas: capa cliente, capa de servicios y capa de repositorio como se muestra en la Ilustración 2.



Descripción de capas

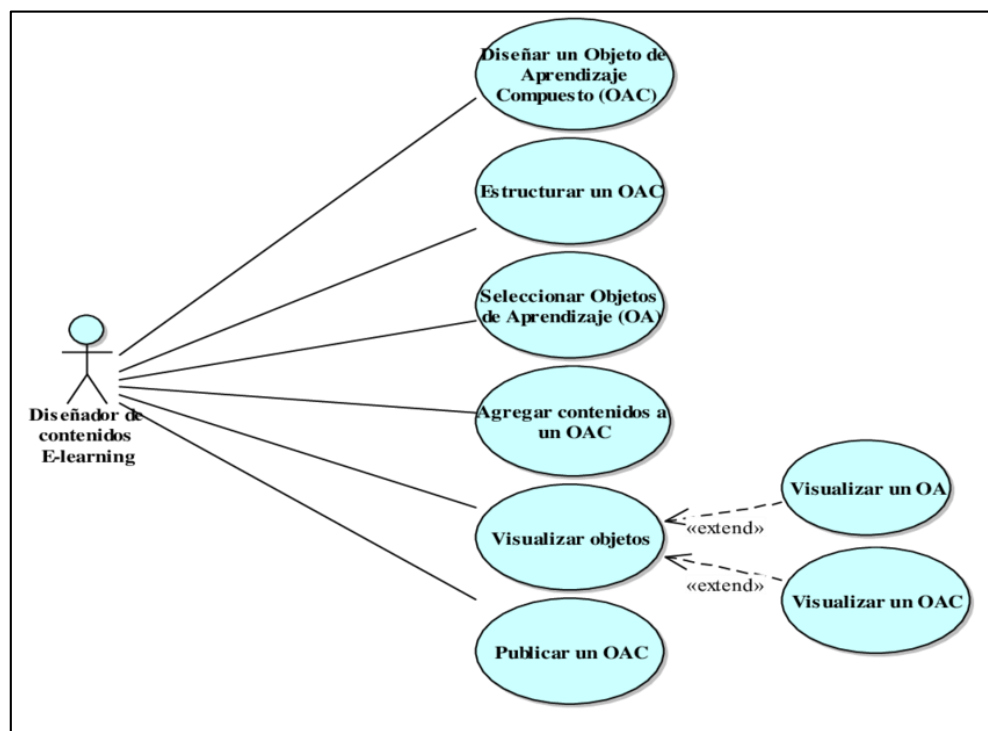
La descripción de cada una de las capas y su funcionalidad se describe a continuación.

La capa del cliente: En esta capa se encuentran las interfaces a través de las cuales el cliente va a solicitar los servicios web dependiendo de su necesidad ilustración a continuación.



La capa de servicios: En esta capa se encuentran los servicios web los cuales producirán una serie de llamadas al repositorio dependiendo de la naturaleza del servicio, los servicios se pueden dividir en dos grupos los servicios con funcionalidades respecto a los Objetos de Aprendizaje y el otro grupo pertenece a los servicios de funcionalidades de gestión de usuarios.

En la ilustración a continuación se presenta el diagrama de casos de uso de las funcionalidades relacionadas con los objetos de aprendizaje.

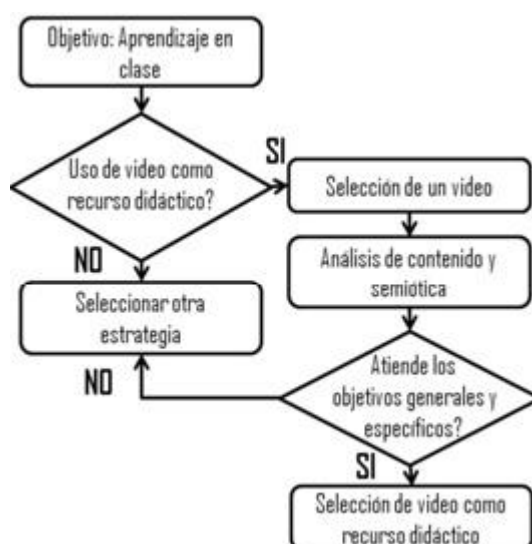


La capa Repositorio: En esta capa se presenta la arquitectura de datos del repositorio en la cual se alojan tanto los Objetos de Aprendizajes (OA) como los metadatos que lo describen. Estos metadatos serán guardados en una base de datos y los OA's se guardan en un directorio dentro del servidor y la URI que referencia la ruta del recurso, la cual será almacenada en la base de datos junto con sus metadatos para poder devolver el recurso cuando algún usuario lo solicite.

Funcionamiento de la Arquitectura del repositorio

La propuesta de repositorio de servicios web funciona de la siguiente manera: el cliente a través de la interfaz soportada en servicios web realiza una solicitud a cualquier servicio ofrecido por la aplicación. Entre los servicios ofrecidos por el sistema se pueden enumerar los siguientes:

- Servicio de registro de objetos de aprendizaje.
- Servicio de carga de información a los objetos de aprendizaje.



- Servicio de descarga de objetos de aprendizaje.
- Servicio de búsqueda de objetos de aprendizaje.

Para el funcionamiento de cada uno de estos servicios los objetos de aprendizaje deben estar contruidos bajo el estándar SCORM 1.2 (2004).

Para la inserción de los metadatos se utilizará el API JDom con el fin de hacer transparente la inserción de estos al usuario, el mismo sistema se encarga de leer el manifiesto y sacar los metadatos y almacenarlo en la base de datos sin que el cliente perciba que inserto los metadatos.

8. Sistema Final

En el siguiente [link](https://github.com/Dayarenth/SistemaDeGestionDeObjetosDeAprendizaje) podrá adquirir el proyecto para realizar las pruebas correspondientes.

9. Descripción del proceso de construcción del software

El proyecto se desarrollará bajo la metodología SCRUM. Es una de las metodologías ágiles más populares.

Es una metodología de adaptación, iterativa, rápida, flexible y eficaz, diseñada para ofrecer un valor significativo de forma rápida en todo el proyecto.

De esta manera se podrá presentar y evaluar progresivamente el avance del proyecto en cada fase de implementación.

Herramientas

- IDE de Eclipse lo puedes descargar del siguiente link <https://www.eclipse.org/downloads/packages/release/Neon/3>
- Git hub, crear una cuenta de forma gratuita en el siguiente link <https://github.com/>
- eXeLearning lo puede adquirir en el siguiente link <http://exelearning.net/>
- Mantis lo puede adquirir en el siguiente link <https://www.mantisbt.org/download.php>

La metodología de comunicación que se utilizó entre los integrantes fue mediante redes sociales, documentos compartidos en la nube de GoogleDrive y reuniones presenciales en las horas de clase.

10. Conclusiones

El presente proyecto ayuda a desarrollar métodos de aprendizaje para diferentes áreas y poder obtener un gran número de documentos relacionados con el mismo tema en una sola dirección de tal manera nos permite realizar mejor la selección.

El trabajar con un IDE libre permite tener más información sobre el mismo es decir existe mayor documentación cuando se presenta un problema.

El trabajar con el método SCRUM favorece y ayuda a colaborar en todo momento con los integrantes del grupo para de esta manera obtener un mejor producto.

11. Recomendaciones

Tener en cuenta todas las librerías que se pueden utilizar en Java para de esta manera utilizar las que sean necesarias para el desarrollo del proyecto.

Realizar el objeto de aprendizaje en eXlearning para facilitar la importación del mismo al software que se está desarrollando.

Estar más al pendiente de los requerimientos que solicita el usuario y no pasarlos por alto.

12. Observaciones

Estar al pendiente de los avances de cada uno de los IDEs que se está utilizando en el desarrollo del proyecto.

Mejor la comunicación con los demás grupos y compartir el conocimiento de cada uno.

Tener en cuenta los requerimientos con los que se está trabajando para el desarrollo del sistema y estar al pendiente de los cambios que desee el usuario.



13. Bibliografía

- [1]. LOM., 2002. Draft Standard for Learning Object Metadata. IEEE 1484.12.1-2002, 15 July 2002. Versión en línea. Consultado el 5 de Abril del 2010 desde: http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf
- [2]. Mills, S., 2002. Learning about learning objects with learning objects, en Proceedings of Society for Information Technology and Teacher Education International Conference, Vol. 1, AACE, págs. 1.158- 1.160.
- [3]. Wiley, D. A., 2002. Connecting learning objects to instructional design theory: A definition, a metaphor and a taxonomy, en D. A. Wiley (ed.), The instructional use of learning objects , Agency for Instructional Technology and Association for Educational Communications and Technology, Bloomington, Indiana, págs. 3-24. Disponible en: <http://www.reusability.org/read/>
- [4]. García Aretio, L. (2005). Objetos de aprendizaje: características y repositorios. [España]: BENED, 2005. (Online). Disponible en:
http://www.tecnoeducativos.com/descargas/objetos_virtuales_deaprendizaje.pdf
- [5]. Los objetos de aprendizaje como recurso para la docencia universitaria: Criterios para su elaboración. Universidad Politécnica de Valencia.(Online) Disponible en: http://www.aqu.cat/doc/doc_22391979_1.pdf
- [6]. López, F. A. (2013). Hacer visible nuestro repositorio digital. En: Infotecarios. Disponible en: <http://www.infotecarios.com/hacer-visible-nuestro-repositorio-digital/>

14. Anexos

Anexo 1

CRONOGRAMA DE TRABAJO

<div>  ESCUELA POLITECNICA NACIONAL CRONOGRAMA DE IMPLEMENTACIÓN DEL SGOA <i>Quito-Ecuador</i>  </div>							
FASE	ACTIVIDADES / ESTRATEGIAS	RESPONSABLE	PRODUCTO	% Cumpl.	Totales		
					Realizadas	Pendientes	Aplazadas
PLANEACIÓN	Plan del proyecto de asignatura	Criollo José Tamayo Edison Vasco Elsa	Informe	100,0	✓		
	Determinación del software libre a utilizarse como línea base	Criollo José Tamayo Edison Vasco Elsa	Software	100,0	✓		
	Levantamiento de requerimientos	Criollo José Tamayo Edison Vasco Elsa	Informe-Software	100,0	✓		
	Creación del repositorio (gestión de objetos de aprendizaje)	Criollo José Tamayo Edison Vasco Elsa	Software	75,0		✓	
	Manual de usuario	Criollo José Tamayo Edison Vasco Elsa	Manual	80,0		✓	
	Informe final del proyecto de asignatura	Criollo José Tamayo Edison Vasco Elsa	Informe	90,0		✓	

Anexo 2

RECURSOS

ESCUELA PÓLITECNICA NACIONAL CRONOGRAMA DE TRABAJO CONSTRUCCIÓN DE OBJETOS DE APRENDIZAJE			
	HUMANOS	MATERIALES	ECONOMICOS
RECURSOS	Ing. Orquera Luis (Product owner) Criollo José (Desarrollador) Tamayo Edison (Desarrollador) Vasco Elsa (Desarrollador y DBA)	Computadoras (Laptops) Papel bond A4	Internet

**Anexo 3
RIESGOS**

RIESGO	IMPACTO	PROBABILIDAD	DESCRIPCIÓN	PLANES DE CONTINGENCIA
Falta de equidad en cuanto al conocimiento de los lenguajes y herramientas utilizadas	ALTO	ALTA	Es posible que todos los miembros del grupo no cuenten con la misma experiencia y conocimiento de los lenguajes y herramientas utilizadas. Si esto sucede, el tiempo de desarrollo en general puede incrementarse significativamente	Se le preguntará a cada miembro que tanta experiencia y conocimiento tiene en los diferentes lenguajes utilizados, con el fin de asignar tareas de acuerdo a los conocimientos, destrezas y habilidades de cada persona. Si alguno de los integrantes tiene un alto conocimiento de alguna de las herramientas, el líder evaluará la conveniencia de programar sesiones de entrenamiento de las herramientas.
Problemas técnicos	MEDIO	BAJA	Se debe considerar la posibilidad de que se presenten problemas técnicos como fallas en los equipos de cómputo, pérdida de información, cortes de energía, etc	Instalar un repositorio de código, datos y documentos, con los niveles de seguridad adecuados, política de permisos y backup definidas.

Problemas de Comunicación	MEDIO	ALTA	Los problemas de comunicación dentro del grupo, pueden provocar una baja en la productividad e incumplimiento de las tareas del proyecto y por ende de los objetivos del grupo.	El líder del grupo debe dar participación a los miembros del grupo en cada reunión, con el fin de conocer los diferentes puntos de vista de cada persona y que se establezca un diálogo sano y amigable. Los acuerdos y responsabilidades deben ser comunicadas a todos los integrantes del grupo, con el fin de evitar malos entendidos. Si el líder detecta que un integrante del grupo tiene alguna dificultad o existe alguna rivalidad, se debe hablar con cada persona y tratar de resolver los conflictos en el menor tiempo posible.
IncompatibilidAd de herramientas y frameworks	ALTO	ALTO	Una vez seleccionada las herramientas que se va a usar es posible que se hallen marcos de trabajo para implementar alguna funcionalidad que no tengan facilidad de integración con lo todo el software del que hasta el momento se disponga	El líder del proyecto debe solicitar más tiempo en caso ocasional para la evaluación de la iteración respectiva. Ver la posibilidad de redefinir la funcionalidad para que se ajuste a un framework existente. Instar reuniones más frecuentes para desarrollar la funcionalidad necesaria desde cero.